

The author(s) shown below used Federal funds provided by the U.S. Department of Justice and prepared the following final report:

Document Title: RACKETS: Case Tracking and Mapping System

Author(s): Colin Reilly ; Victor Goldsmith

Document No.: 182918

Date Received: June 14, 2000

Award Number: 98-LB-VX-0004

This report has not been published by the U.S. Department of Justice. To provide better customer service, NCJRS has made this Federally-funded grant final report available electronically in addition to traditional paper copies.

Opinions or points of view expressed are those of the author(s) and do not necessarily reflect the official position or policies of the U.S. Department of Justice.

182918

RACKETS

Case Tracking and Mapping System

Developed for:
United States Attorney's Office,
Southern District of New York

Developed by:
Colin Reilly
Center for Applied Studies of the Environment,
Hunter College, City University of New York
Victor Goldsmith,
Dean of Research,
Hunter College, City University of New York

PROPERTY OF
National Criminal Justice Reference Service (NCJRS)
Box 6009
Rockville MD 20849-6000

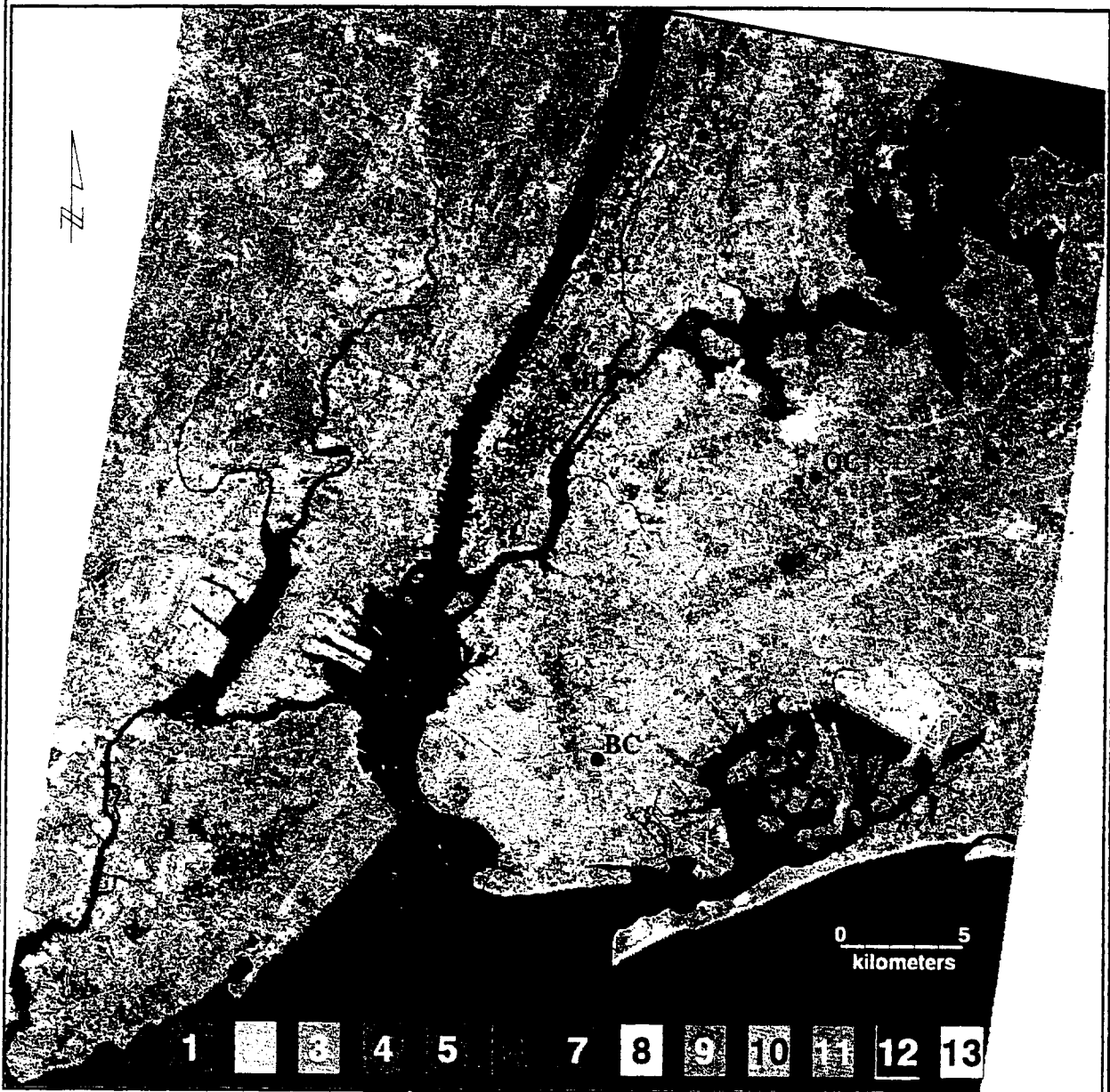
FINAL REPORT TO:
National Institute of Justice
United States Department of Justice
Grant No. 98-LB-VX-0004
August 1999

CENTER FOR APPLIED STUDIES OF THE ENVIRONMENT CITY UNIVERSITY OF NEW YORK

Professor Victor Goldsmith (Hunter College), Director
Professor Arthur Langer (Brooklyn College), Deputy Director
Professor Robert Graff (City College), Deputy Director

Senior Colleges of CUNY:

BC - Brooklyn College, CC - City College, GSUC - Graduate School and University Center,
HC - Hunter College, LC - Herbert H. Lehman College and QC - Queens College.



A SPOT Image (June 25, 1988) is shown merged with an EDSAT IMAGE (August 30, 1990), resulting in a multispectral image with a 10 m resolution (after a technique by Ahearn, 1993). In this classification, the lower numbers represent more urbanized areas. This merged image is from a report prepared for the NYC Department of Environmental Protection (Goldsmith et al., 1993).

Introduction

Rackets is a case tracking and mapping application developed by the Center for Applied Studies of the Environment (CAPSE) for the U.S. Attorney's Office (USAO), Southern District of New York (SDNY), hereto referred to as SDNY. The SDNY and the National Institute of Justice (NIJ) provided the funding for this project. Work on the project began in July 1997 and the bulk of which ended in October of 1998.

The impetus for Rackets developed out of a desire by the SDNY to move away from the traditionally episodic way of handling cases to a comprehensive and strategic method. The SDNY is the federal prosecuting office for Manhattan and Bronx counties in New York City, as well as Westchester and several other northern counties. As such, the Office's primary mission is to investigate and prosecute federal crimes in these counties. In the course of this work, the Office interacts with many different federal, state and local agencies. Oftentimes, information gathered by one agency for a certain case is related to other cases under investigation by other agencies without either agency aware of the other's investigation. For example, the Department of Agriculture may be investigating Food Stamp fraud in a grocery store where the Bureau of Alcohol Tobacco and Firearms is investigating a gun trafficking ring and there may be a critical link that could strengthen both cases.

Additionally, criminals are linked by geography and pedigree which cross investigative boundaries. Therefore, with the development of system that could collect case specific information and link it to specific geographic locations the Office would have at its disposal the ability to cut across agency lines and establish geographic connections between criminals and federal cooperators. Rackets offers, at a glance, an in depth analytic capacity of the entire federal landscape for the Southern District of New York.

Rackets was initially designed to collect information on the cases, defendants, investigations, wiretaps and warrants for the violet gangs, narcotics, organized crime and general crimes units. The system was developed to collect information either not handled,

or not handled at sufficient enough detail by the Office's existing case management system (PROMIS at the time of development). The current case management system, LIONS, is charged with handling the administrative details of the Office's cases and as such does not attempt to collect the detailed information that exists in the charging instruments or through proffers or cooperation agreements. Information is collected on a case by case basis, filed, and utilized further if an office-wide request is made. Rackets attempts to fill this information void by utilizing the information for future investigations and cases. Further, all geographic information is entered and mapped thereby linking the data to discrete locations. Then, the data can be searched by traditional methods or by spatial queries.

The System

Rackets is an end-user application designed to be run largely by the Office's non-technical staff. To accomplish this, a simple user-interface was developed to allow the staff to intuitively enter and query the data. Numerous interactive canned queries and reports are included in the application. The mapping component of the system has also been customized to allow the staff to perform spatial queries without having to be a Geographic Information Systems (GIS) expert. Nonetheless, the system is not maintenance free and requires a degree of technical expertise to maintain and undertake the more complicated tasks.

Information input into the database is collected off of the Office's charging instruments. These documents are digested for all pertinent information and then entered into the system by USAO staff. To clothe this data, pedigree information for all arrested defendants is obtained via electronic transfer from the U.S. Marshals Prisoner Tracking System (PTS) database and entered into Rackets.

As part of the implementation of the system, existing case data from PROMIS were imported into Rackets. This was a cumbersome procedure that required the data to be cleaned (removal of duplicates and closed cases) and converted into relational format. Additionally, prisoner data from PTS was also imported into Rackets. Thus, upon Rackets inception over 2,00 cases and 4,000 defendants resided in the database.

Due to the sensitive nature of some of the data as well as organizational constraints, the system was developed as a stand-alone application. In the future, a link could be developed to LIONS, but until then, the system functions as its own entity. To further maintain security, the system runs on a secure Windows NT network with restricted access.

The Network

The network contains a server (Bronx) and two workstations (New York, and Richmond). Bronx contains all server applications and the data. New York is the GIS workstation and Richmond is for data entry. Both workstations have a mapped-out network drive (Z:) which links them to the network drive on Bronx.

Rackets

Rackets is a client-server application composed of database and GIS components. A client-server application separates the data and interface components into two separate applications, a client and a server. The server database manages the data while the client applications manage the user interface (i.e., all forms, queries, reports and underlying code). A single copy of the server database exists and is installed on the network server.¹ Operationally, the client applications send requests to the server and in turn the server returns the data or pointers to that data. The database server is the component that carries out the actual updating of the data. This allows multi-users to view and edit the same data

¹ A network server is a physical computer that functions as the network's brain and allows the sharing of resources between computers, while a database server is an application that manages and stores the data of a client-server database.

concurrently, although not the same record. The database was developed using Microsoft Access.

In actuality, Rackets is composed of three database servers that are all linked to the client applications installed on each computer. In total, there are five separate components making up the entire application, which are as follows:

1. Main database server
2. PTS database server
3. Archive database server
4. Database client applications, installed on each workstation.
5. GIS mapping component

The following section discusses the database structure and provides a detailed description of each component, all the objects contained in them and their interconnectivity.

System Components

Database Structure

Since the office carries out its operations on a case basis, the database mimics this organizational structure and uses a case table as its central table. The case table collects information specific to the case, such as the investigating agency and agent, Assistant U.S. Attorney (AUSA) handling the case and pertinent dates. From this table, defendant, wiretap, warrant and target (individuals or entities being investigated) tables are linked and contain from zero to many entries for each case. The next major table stores defendant information and also has other tables linked to it. For instance, alias, statutes, and events tables are linked in a one-to-many relationship with the defendant table. Although there is frequently a defendant linked to more than one case, warranting a many-to-many relationship between the case and defendant table, the Office preferred to maintain a one-to-many relationship. This is because many times defendants give different information each time they are arrested. Therefore, all details from each arrest would be recorded arrest those details would be picked up.

Although new arrests are electronically transferred on a daily basis to the office from PTS (detailed in Appendix B), these records are not immediately transferred into the main

database. This is due in part because each time there is some activity for a prisoner (e.g., sentencing), update information is electronically transferred to the SDNY. As such, the electronic transfers represent not only new arrestees, but existing prisoners who could predate the system. Rackets is setup to enforce referential integrity (e.g., a defendant record can not be entered without an existing case record to which to be associated) and prevent the accumulation of orphan records (i.e., unrelated records).

Additionally, by maintaining these data in a separate location it enables the Office's staff to monitor all new entries into the system. For this reason, the PTS database functions as a sort of holding cell that contains all of the PTS transfers. These records are maintained in this database until they are transferred to the main database. Even then, only some of the data is transferred to the main database, including the identification number for each prisoner record (*pr_key*), which allows linkages to all the data remaining in the PTS database for each prisoner. Finally, an archive database server was developed which is an abbreviated version of the main database and stores records of closed cases and sentenced defendants.

Database Servers

As the data manager component, the server database acts as the warehouse of the application and as such stores all the data, and contains the necessary tables to efficiently manage these data. Appendix A is a data dictionary that details every table and field included in the database. All tables in the server application are linked to the client applications from the client end. The file locations of the server applications on Bronx are:

Main - z:\network\rackets\data\Rackets_be.mdb
PTS - z:\network\Pts\snpts.mdb

The client applications on the workstations must be directed to the location of the server via their network drives to the following location:

z:\Rackets\data\Rackets_be.mdb

If changes are made to the location of the server application, the links from the client application need to be updated. To update the table links go to the Linked Table Manager from the Tools menu, select Add-Ins. Once there, separately select all tables from **Rackets_be.mdb** and **snpts.mdb**, and direct the application to their respective file locations.

If a table is accidentally deleted or missing, it can be re-linked from the client application as follows: under the File menu go to Get External Data and select Link Tables. Then direct the application to the location of the table (either Rackets_be.mdb or snpts.mdb) that should be re-linked.

Client

As stated previously, the client application contains the interface objects of the database application. This is what the user sees when running RACKETS from the desktop by double clicking the icon in figure 1.1. The client application contains all the forms, queries, reports, macros, table links and code necessary to enter, navigate through and query the data (the Visual Basic Scripts behind the forms are in Appendix D). A copy of the client application is installed locally on each computer on the network. The local client applications access the server application on the network server via their network drive (Z:\).



Figure 1.1

Mapping Application

The mapping application was developed using Environmental Systems Research Institute's (ESRI) ArcView 3.0a GIS. ArcView was customized to link to the data

maintained in the server databases, to automate the Attorney's Offices most common queries and the ability to run simple analyses. The customization includes the following:

- a new menu item Rackets, (see figure 1.2) containing the customized operations;
- a button that labels streets and RACKETS themes (themes based on tables in the RACKETS database);
- a button that saves a map layout as a graphic image file;
- automatic connection to, and visualization of the data in the databases;
- setup of all basemap themes.

The customization is contained in an ArcView project (rackets.apr) and is installed locally on each computer. This project includes all the Avenue code (the scripting language included with ArcView), dialogs, and formatting required to carry out all operations (Appendix C). The mapping application is accessed from the main menu of Rackets (see figure 1.)

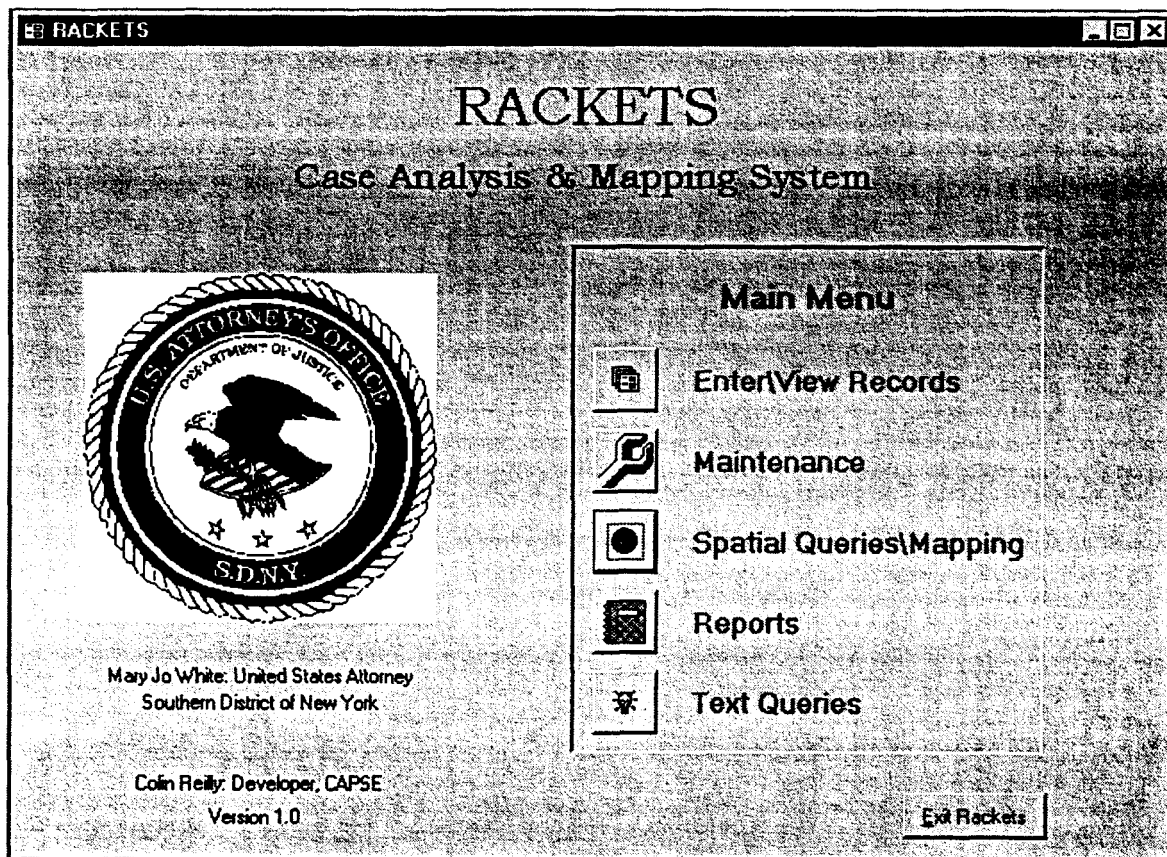


Figure 1.2

The mapping application is connected to the database via Open Database Connectivity (ODBC). A DSN system data source was set up on every computer. The DSN is referred to by the same name as the application, RACKETS. The connection is made each time the mapping application is run via the "StartUp" script associated with the project, and the connection is subsequently closed each time the project is closed by the "ShutDown" script.

New York functions as the GIS workstation and the GIS functionality has been optimized for this computer, therefore all mapping work should be done on the New York. However the mapping application functions on all computers except for the reporting capability.

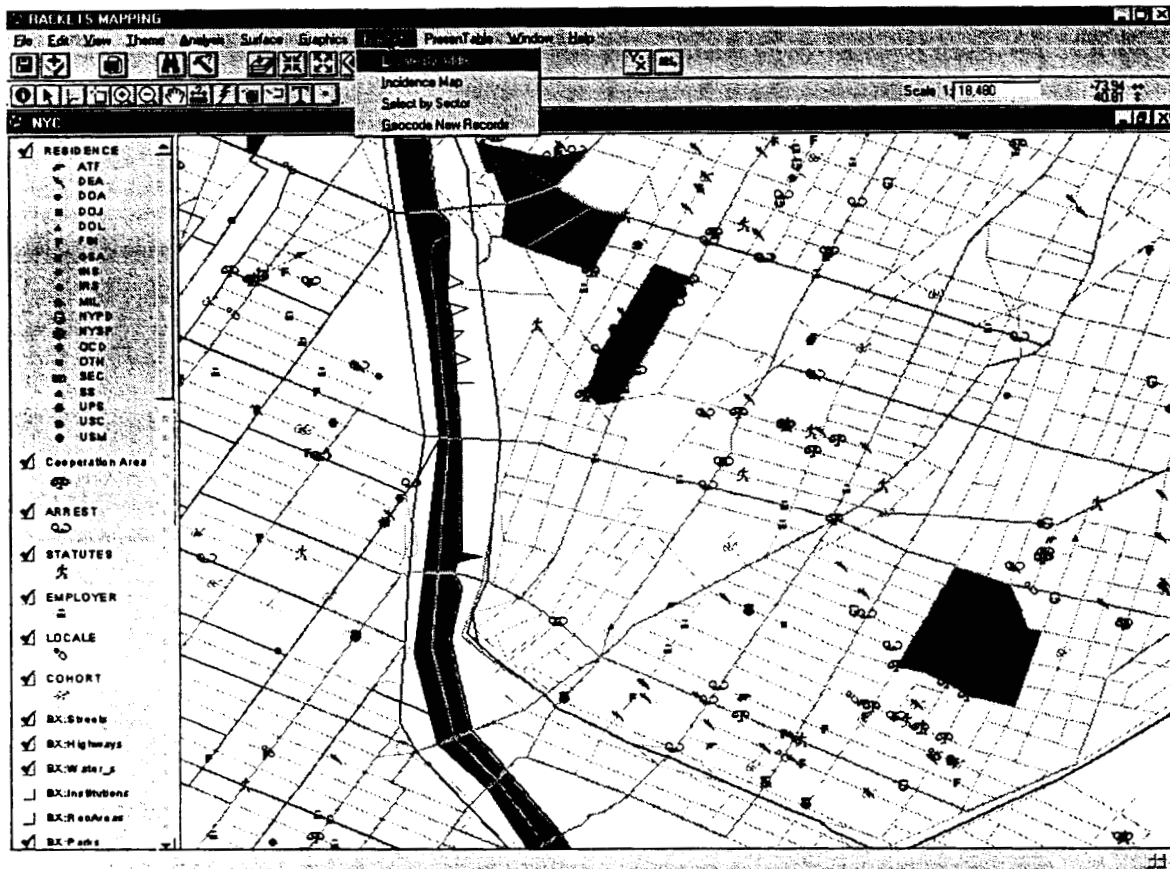


Figure 1.3

RACKETS Themes

These are the themes that are based on tables in the Rackets database. These themes are not ArcView shape files; they are event themes based on virtual tables resident in the Rackets database. These virtual tables are established by sending queries to the database via the ODBC connection and represent a view of the data, not an actual copy of the data. This way, the database manages and stores all the data in one single location and the GIS can map, query and update the data via the ODBC connection. Therefore, one does not need to worry about having to create a shape file from the tabular data each time a new query is run. This prevents the possibility of having numerous time-dependant copies of the data floating around. With the virtual tables, changes made to the data are reflected in the virtual tables immediately after the tables are refreshed. This occurs each time the application is run or by selecting refresh from the table menu for each table.

For each table in the Rackets database that is mapped there are two virtual tables in the mapping application. One is the actual table to be mapped, the map table, and consists of records already geocoded and containing their x and y coordinates (e.g., EMPLOYER). These tables are built from map queries residing on the client application (e.g., mEMPLOYER) and have the prefix "m". A second table contains records that have not yet been geocoded and are based on geocoding queries residing on the client application. These tables have the same name as their mapped table counterpart with the addition of the prefix "gc" (e.g., gcEMPLOYER), for geocoding. This is the table to geocode with, and the mapped table is the table that is added as an event theme and visible in the view (see figure 1.3).

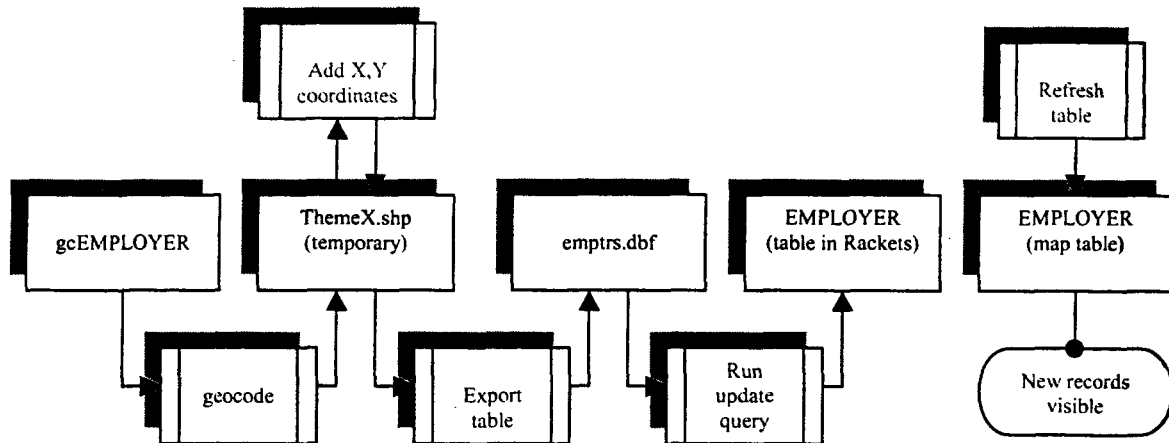


Figure 1.4

The process necessary to go from the geocode table to the virtual table that is added as an event theme is outlined in figure 1.4 for the Employer table. The steps have been automated by the script “Rackets.Geocode” which is run from the Rackets menu item “Geocode new records.” First, the records are geocoded and a temporary shape file is created, then the x and y coordinates are generated from the shape field and the file is exported as a dbase file (emptrs.dbf). This file is linked to the Rackets database and an existing update query in Rackets is run via a macro using Dynamic Data Exchange (DDE) to insert the x and y coordinates into the corresponding records. Once the records are updated, the map table (e.g., EMPLOYER) can be refreshed to view the newly geocoded records.

There are eight such mapping tables in the application and others can be added following the same process. The above example outlines the procedure for the Employer table in Rackets. Table 1.1 outlines all the Rackets themes, tables and queries used in the geocoding and data visualization process.

Event theme	Map table	Geocode table	Update query	Macro
Residences	DEFENDANTS	gcDEFENDANTS	uqDefXY	mcrDefXY
Cooperation Area	COOP_AREA	gcCOOPAREA	uqCoopAreaXY	mcrCoopAreaXY
ARREST	ARREST	gcARREST	uqArrXY	mcrArrXY
STATUTES	STATUTES	gcSTATUTES	uqStatuesXY	mcrStatutesXY
EMPLOYER	EMPLOYER	gcEMPLOYER	uqEmployXY	mcrEmployXY
LOCALE	LOCALE	gcLOCALE	uqLocaleXY	mcrLocaleXY
COHORT	COHORT	gcCOHORT	uqCohortXY	mcrCohortXY
HOMICIDES	HOMICIDES	gcHOMICIDES	uqHomicideXY	mcrHomicideXY

Table 1.1

Base Maps

The customization of the mapping application also included the selection, purchase and setup of base data layers. All of the base maps were added to the project and their legends and properties were set (see figure 1.3). The base maps are Geographic Data Technologies' Dynamap 2000 for the five boroughs of New York City. The base maps include the following layers:

- Major streets
- Local streets
- Water bodies
- Parks
- Landmarks
- Institutions
- County polygons
- U.S. Census tracts
- U.S. Census Block groups

In addition, New York Police Department (NYPD) precinct and precinct census boundaries were created for the GDT base maps from NYPD maps and overlay directly on the streets.

To assist in the geocoding process a customized table of place name aliases (e.g., WTC for World Trade Center) was developed. ArcView uses this alias table to locate places

with which it is unfamiliar. The table was developed over the course of the year as common aliases that were not being geocoded became evident.

Queries and Procedures

Avenue scripts were written to automate the most frequent queries run by the SDNY. The first such procedure, selects and zooms to specified records (e.g., Cooperation Area; an area given by a federal cooperator for which he/she has specific knowledge) within a certain radius of a specified address or intersection. Depending on the user input, the procedure produces a map layout and/or report for printing based on the search results. Information for the query is entered in the dialog below (figure 1.5) which is run from the Rackets main menu (see figure 1.3). Prior to showing this query dialog, a dialog prompts the user for an address. If the address is found the Locate By Address dialog appears, otherwise a message appears indicating that the address could not be located and notifies the user to check the address and re-enter it or try another address.

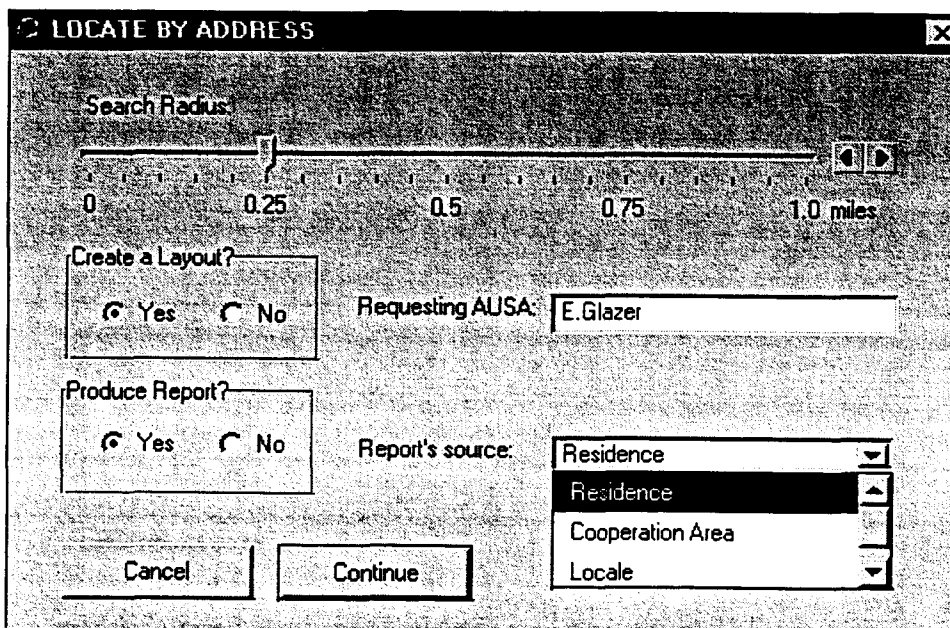


Figure 1.5

If the user chooses to generate a report, the report gives detailed information contained in the database for the records selected within the search radius. For example, if an AUSA

has a new case on Maple Ave & Smith St and she would like to know if there are any cooperators with knowledge of the area, she could run the above query selecting Cooperation Area as the Report's source and the procedure would produce a report listing all federal cooperators entered in Rackets and within the specified radius.

The next procedure Select By Sector selects all records from a specified table ("Select Records from:") contained within a selected NYPD precinct sector. This query uses the dialog below (figure 1.6) to gather the query information specified above from the user. A selection table is created containing the selected records for this query.

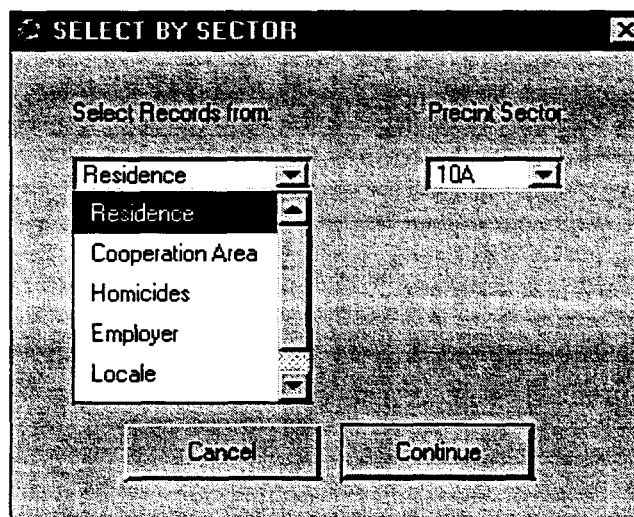


Figure 1.6

The final procedure from the Rackets menu, Incidence Map counts the number of events (e.g., Cooperators, Homicides) within each precinct sector and classifies the resulting theme by the number of events per sector. Figure 1.7 is an example of the output from this operation. This offers the Attorney's Office a simplified view of the spatial distribution of their docket at any given point in time.

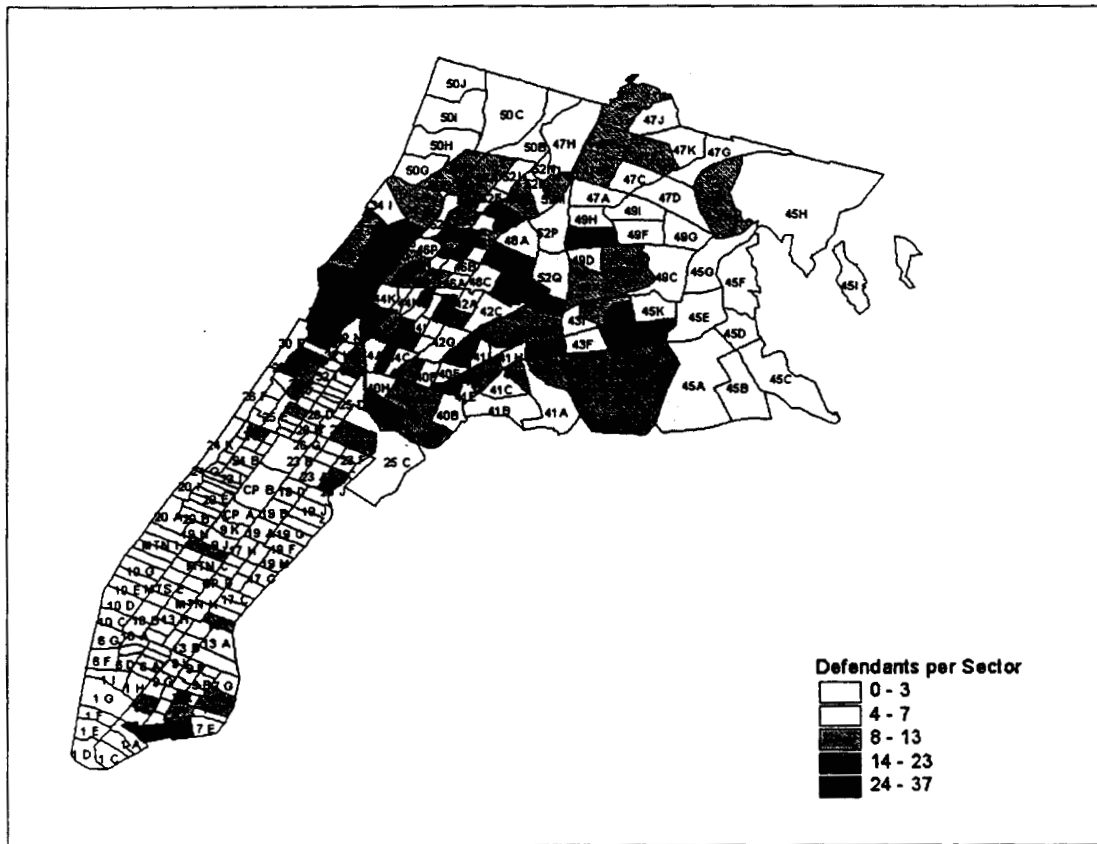


Figure 1.7

Conclusion

This manual provides a brief overview and description of Rackets. The development of Rackets has been an ongoing and evolving process over the last year. Many changes and updates were incorporated into the system over the course of the year. The evolution of the system was evident, as it went from running on a stand-alone PC, to a local area network with three workstations. Ideas were hatched and visions clarified as we moved along with the work. From a concept came a system and potentially a powerful tool as envisioned by its creator, Elizabeth Glazer. Hopefully the work will prove beneficial and further work will be done to improve and expand the system.

Appendix A- Data Dictionary

The data dictionary details every table and field included in the database. All queries and reports derived from the system are based on the information found within these tables. Users can update and navigate through this information through the client application / user interface.

Appendix A

Data Dictionary

Tables:

ALIAS

Field Name	Data Type	Field Size	Special	Description
AliasKey	Number (Long)	4	Primary key	Auto Number
DefKey	Number (Long)	4	Foreign key	Link to Defendant table (1:M)
Alias	Text	50		Alias
AliasLN	Text	30		Alias Last name
Alias_rm	Text	30		Remarks

ARREST

Field Name	Data Type	Field Size	Special	Description
ArrKey	Number (Long)	4	Primary key	Auto Number
DefKey	Number (Long)	4	Foreign key	Link to Defendant table (1:M)
ArrDate	Date/Time	8		Arrest date
Location	Text	50		Location of arrest (e.g., Restaurant, Home)
Address	Text	50		Address of arrest
City	Text	20		City of arrest
State	Text	2		State of arrest
Zip	Text	10		Zip code of arrest
Remarks	Text	70		Comments
Xcoord	Number (Single)	4		X coordinate – latitude
Ycoord	Number (Single)	4		Y coordinate – longitude
Av_status	Text	9		Geocoding status (M=matched, U=unmatched)

CASE

Field Name	Data Type	Field Size	Special	Description
CaseKey	Number (Long)	4	Primary key	Auto Number
USAONum	Text	20		USAO number
IndictmentNum	Text	30		Instrument number
CaseName	Text	80		Title of case
AUSA	Text	70	LUT	Assistant US Attorney
Unit	Text	4	LUT	Prosecuting unit
agency_code	Text	7	LUT	Abbreviation of arresting agency
Agent	Text	40		Investigating agent
StatuteNum	Text	25	LUT	Statute number of lead charge
OpenDate	Date/Time	8		Date case opened
CloseDate	Date/Time	8		Date case closed
EnterDate	Date/Time	8		Date case entered into RACKETS
Comments	Memo	-		Comments

COOP_AREA

Field Name	Data Type	Field Size	Special	Description
CoopAreaKey	Number (Long)	4	Primary key	Auto Number
CoopKey	Number (Long)	4	Foreign key	
CoopArea	Text	70		Cooperation area
NypdSec	Text	8		NYPD sector
Topic	Text	20	LUT	Cooperation topic (e.g., narcotics, homicide)
City	Text	25		City of cooperation area
State	Text	2		State of cooperation area
Zip	Text	10		Zip code of cooperation area
Xcoord	Number (Single)	4		X coordinate – latitude
Ycoord	Number (Single)	4		Y coordinate – longitude
Av_status	Text	9		Geocoding status (M=matched, U=unmatched)

COOP_PEOPLE

Field Name	Data Type	Field Size	Special	Description
CPKey	Number (Long)	4	Primary key	Auto Number
CoopKey	Number (Long)	4	Foreign key	
CoopAreaKey	Number (Long)	4	Foreign key	
LName	Text	40		Last name
Activity	Text	50	LUT	Activity at location
FName	Text	35		First name
Remarks	Text	100		Comments

COOPERATOR

Field Name	Data Type	Field Size	Special	Description
DefKey	Number (Long)	4	Foreign key	Link to Defendant table (1:M)
CoopKey	Number (Long)	4	Primary key	
CoopDate	Date/Time	8		
Status	Text	6	COOP or PROF	Status (Cooperator or Proffer)
Topic	Text	20	LUT	Cooperation topic
Comment	Memo	-		Comments

DEFENDANT

Field Name	Data Type	Field Size	Special	Description
DefKey	Number (Long)		4 Primary key	Auto number
CaseKey	Number (Long)		4 Foreign key	Link to Case table (1:M)
pr_key	Number (Long)		4 Foreign key	Link to PTS data
DefName	Text	35		Last name
DefFName	Text	30		First name
DefMName	Text	25		Middle name
Address	Text	60		Address of residence
Apartment	Text	12		Apartment of residence
City	Text	40		City of residence
State	Text	5		State of residence
Zipcode	Text	10		Zip code of residence
USMSNum	Text	50		US Marshals Service number
FBI Num	Text	20		FBI number
SSNum	Text	20		Social security number
NYSID	Text	20		New York State ID
LeadDef	Yes/No	1		Lead defendant?
Coop	Yes/No	1		Cooperator?
Weapon	Yes/No	1		Weapon seized?
CrimeType	Text	20	LUT	Type of Crime
CrimeRole	Text	70		Role in crime
Sealed	Yes/No	1		Sealed?
Fugitive	Yes/No	1		Fugitive?
LicNum	Text	30		License number
LicSt	Text	3		License state
DOB	Text	8		Date of birth
POBcity	Text	40		Place of birth city
POBstate	Text	50		Place of birth state
race_code	Text	50	LUT	Abbreviation for race
Citizenship	Text	12		Country of citizenship
AlienNum	Text	20		Alien number
CounselLN	Text	30		Defense Counsel's last name
CounselFN	Text	30		Defense Counsel's first name
TelNum	Text	25		Telephone number
Beeper	Text	15		Beeper number
Xcoord	Number (Double)	8		X coordinate – latitude
Ycoord	Number (Double)	8		Y coordinate – longitude
Av_status	Text	9		Geocoding status (M=matched, U=unmatched)
Updated	Date/Time	8		Date record entered or transferred from PTS
MugshotA	OLE Object	-		Mug shot photo – front
MugshotB	OLE Object	-		Mug shot photo - side

EMPLOYER

Field Name	Data Type	Field Size	Special	Description
DefKey	Number (Long)	4		Link to Defendant table (M:1)
EmplName	Text	50		Name of employer
Occupation	Text	25		Occupation of employee
Address	Text	50		Address of employer
City	Text	20		City of employer
State	Text	2		State of employer
Zip	Text	10		Zip code of employer
Xcoord	Number (Single)	4		X coordinate – latitude
Ycoord	Number (Single)	4		Y coordinate – longitude

FIREARMS

Field Name	Data Type	Field Size	Special	Description
FireKey	Number (Long)	4	Primary key	Auto Number
DefKey	Number (Long)	4	Foreign key	Link to Defendant table (1:M)
CoopKey	Number (Long)	4		
FirearmCode	Text	2		Type of firearm
Make	Text	3		Make of firearm
Model	Text	30		Model of firearm
Caliber	Text	4		Caliber of firearm
SerialNum	Text	20		Serial number
PurchaseDate	Date/Time	8		Purchase date
Trace	Yes/No	1		Was the firearm traced?
Source	Text	35		Source of firearm (e.g., store, individual)
Address	Text	50		Address of source
City	Text	25		City of source
State	Text	2		State of source
Zip	Text	10		Zip code of source
Remarks	Text	70		Comments
Xcoord	Number (Single)	4		X coordinate – latitude
Ycoord	Number (Single)	4		Y coordinate – longitude

GROUPS

Field Name	Data Type	Field Size	Special	Description
DefKey	Number (Long)	4	Foreign key	Link to Defendant table (M:1)
GroupName	Text	50		Name of group/organization/enterprise
Location	Text	30		General Location of group
Xcoord	Number (Single)	4		X coordinate – latitude (NOT USED)
Ycoord	Number (Single)	4		Y coordinate – longitude (NOT USED)

HOMICIDE

Field Name	Data Type	Field Size	Special	Description
HomKey	Number (Long)	4	Primary key	Auto number
DefKey	Number (Long)	4	Foreign key	Link to Defendant table (1:M)
CoopKey	Number (Long)	4		
Status	Text	15		Status of homicide
VicLName	Text	30		Victim's last name
VicFName	Text	25		Victim's first name
TgtLName	Text	35		Target's last name
TgtFName	Text	30		Target's first name
HomDate	Date/Time	8		Date of Homicide
Address	Text	50		Address of homicide
City	Text	20		City of homicide
State	Text	2		State of homicide
Zipcode	Text	10		Zip code of homicide
Xcoord	Number (Single)	4		X coordinate – latitude
Ycoord	Number (Single)	4		Y coordinate – longitude

LOCALE

Field Name	Data Type	Field Size	Special	Description
LocKey	Number (Long)	4	Primary key	Auto number
DefKey	Number (Long)	4	Foreign key	Link to Defendant table (1:M)
LocName	Text	30		Name of locale
Address	Text	50		Address of locale
Apartment	Text	12		Apartment of locale
City	Text	40		City of locale
State	Text	5		State of locale
Zipcode	Text	10		Zip code of locale
Comment	Memo	-		Comments
Xcoord	Number (Single)	4		X coordinate – latitude
Ycoord	Number (Single)	4		Y coordinate – longitude
Av_status	Text	9		Geocoding status (M=matched, U=unmatched)

MISC

Field Name	Data Type	Field Size	Special	Description
MiscKey	Number (Long)	4	Primary key	Auto number
DefKey	Number (Long)	4	Foreign key	Link to Defendant table (1:M)
Item	Text	70		Miscellaneous item
Desc	Text	150		Description

NARCOTIC

Field Name	Data Type	Field Size	Special	Description
NarcKey	Number (Long)	4	Primary key	Auto number
DefKey	Number (Long)	4	Foreign key	Link to Defendant table (1:M)
NarcType	Text	15	LUT	Type of narcotic
Brand	Text	30		Brand of Narcotic (i.e., street name)

Origin	Text	30		Origin of narcotic
--------	------	----	--	--------------------

REL_CASES

Field Name	Data Type	Field Size	Special	Description
CaseKey	Number (Long)	4	Foreign key	Link to Case table (M:1)
RelCase	Text	20		Related case's USAO number
Comments	Text	100		Comments

RELATIONS

Field Name	Data Type	Field Size	Special	Description
RelKey	Number (Long)	4	Primary key	Auto number
DefKey	Number (Long)	4	Foreign key	Link to Defendant table (1:M)
rel_desc	Text	15	LUT	Type of relation (e.g., Brother, Mother)
RelLName	Text	30		Relation's last name
RelFName	Text	25		Relation's first name
Address	Text	50		Address of relation
City	Text	20		Address of relation
State	Text	2		Address of relation
Zip	Text	10		Address of relation
Xcoord	Number (Single)	4		X coordinate – latitude
Ycoord	Number (Single)	4		Y coordinate – longitude

STATUTES

Field Name	Data Type	Field Size	Special	Description
StatKey	Number (Long)	4	Primary key	Auto number
DefKey	Number (Long)	4	Foreign key	Link to Defendant table (1:M)
Count	Text	15		Count
Supercede	Text	2		Superceding indictment
StatuteNum	Text	25	LUT	Statute number
Victim	Text	40		Victim of crime
Location	Text	40		Location of crime (e.g., restaurant)
Address	Text	50		Address of crime
City	Text	20		City of crime
State	Text	2		State of crime
Zip	Text	10		Zip code of crime
Xcoord	Number (Single)	4		X coordinate – latitude
Ycoord	Number (Single)	4		Y coordinate – longitude
Av_status	Text	9		Geocoding status (M=matched, U=unmatched)

TARGET_ALIAS

Field Name	Data Type	Field Size	Special	Description
TgtAliasKey	Number (Long)	4	Primary key	Auto number
TargetKey	Number (Long)	4	Foreign key	Link to target table
TgtAlias	Text	35		Alias of target
TgtAliasLast	Text	40		Alias last name
TgtAliasRemark	Text	100		Comments

TARGETS

Field Name	Data Type	Field Size	Special	Description
TargetKey	Number (Long)	4	Primary key	Auto number
CaseKey	Number (Long)	4	Foreign key	Link to Case table (1:M)
TargetLast	Text	35		Last name of target
TargetFirst	Text	30		First name of target
TargetMiddle	Text	30		Middle name of target
Address	Text	60		Address of target
Apartment	Text	12		Apartment
City	Text	40		City
State	Text	5		State
Zipcode	Text	10		Zip code
SSNum	Text	20		Social security number
NYSID	Text	20		New York State Identification number
FBI	Text	15		FBI number
Photo	OLE Object	-		Photo
Remarks	Memo	-		Comments
Xcoord	Number (Double)	8		X coordinate - latitude
Ycoord	Number (Double)	8		Y coordinate - longitude
Av_status	Text	9		Geocoding status (M=matched, U=unmatched)

WARRANT

Field Name	Data Type	Field Size	Special	Description
WarrantKey	Number (Long)	4	Primary key	Auto number
CaseKey	Number (Long)	4	Foreign key	Link to Case table (1:M)
SearchDate	Date/Time	8		Date of search
Address	Text	50		Address searched
Apartment	Text	12		Apartment
City	Text	20		City
State	Text	2		State
Zipcode	Text	10		Zip code
Xcoord	Number (Single)	4		X coordinate - latitude
Ycoord	Number (Single)	4		Y coordinate - longitude
Av_status	Text	9		Geocoding status (M=matched, U=unmatched)

WARRANT_INVT

Field Name	Data Type	Field Size	Special	Description
WarrantKey	Number (Long)	4	Foreign key	Link to Warrant table (M:1)
Item	Text	50		Item confiscated

WARRANT_NAMES

Field Name	Data Type	Field Size	Special	Description
WarrantKey	Number (Long)	4	Foreign key	Link to Warrant table (M:1)
LName	Text	30		Last name on warrant
FName	Text	25		First name on warrant

WIRE_NAMES

Field Name	Data Type	Field Size	Special	Description
WireKey	Number (Long)	4	Foreign key	Link to Wiretap table (M:1)
LName	Text	30		Last name on wiretap
FName	Text	25		First name on wiretap

WIRETAP

Field Name	Data Type	Field Size	Special	Description
WireKey	Number (Long)	4	Primary key	Auto number
CaseKey	Number (Long)	4	Foreign key	Link to Case table (1:M)
Comments	Memo	-		Comments
TelNum	Text	50		Telephone number
Address	Text	50		Address of wiretap
Apartment	Text	12		Apartment
City	Text	20		City
State	Text	2		State
Zipcode	Text	10		Zip code
Xcoord	Number (Single)	4		X coordinate -- latitude
Ycoord	Number (Single)	4		Y coordinate -- longitude
Av_status	Text	9		Geocoding status (M=matched, U=unmatched)

WIRETAP_DATES

Field Name	Data Type	Field Size	Special	Description
WireKey	Number (Long)	4	Foreign key	Link to Wiretap table (M:1)
BeginDate	Date/Time	8		Begin date
EndDate	Date/Time	8		End date

Lookup Tables**agency**

Field Name	Data Type	Field Size	Special	Description
agency_code	Text	4		Agency code
agency_name	Text	35		Agency name

ausa

Field Name	Data Type	Field Size	Special	Description
AUSA	Text	50		Assistant US Attorney

coopTopic

Field Name	Data Type	Field Size	Special	Description
Topic	Text	50		Cooperation topic

crime_type

Field Name	Data Type	Field Size	Special	Description
CrimeType	Text	25		Crime type

coop_activity

Field Name	Data Type	Field Size	Special	Description
Activity	Text	50		Linked to COOP_AREA Topic

event_type

Field Name	Data Type	Field Size	Special	Description
event_type	Text	20		Type of event (e.g., Proffer, Indictment)

crimecode

Field Name	Data Type	Field Size	Special	Description
StatuteNum	Text	20		Statute number
StatDesc	Text	70		Description

narc_type

Field Name	Data Type	Field Size	Special	Description
NarcType	Text	20		Type of narcotic (e.g., heroin, cocaine)

unit

Field Name	Data Type	Field Size	Special	Description
Unit	Text	5		Unit code
UnitDesc	Text	30		Unit

***LUT: See Lookup table**

Appendix B - Downloading PTS Data from USMS

Everyday, all new arrest information received from the United States Marshall's Service Prisoner Tracking System is transferred to the RACKETS system. Once downloaded these records are separated from the main database until they are properly transferred to the main database.

Appendix B details the process by which data are initially brought into the RACKETS system from the United States Marshall's Service.

Instructions for downloading PTS data from USMS

This must be done from RICHMOND.

1. Double-click USMS Dial-up Networking connection from desktop.



2. Enter password and hit Connect. When connection is established the dialog box will minimize to lower right corner of computer. *If there is a problem first check Dial-up Networking connection and then if the problem persists contact USMS.*
3. Double-click PTS Downloads.

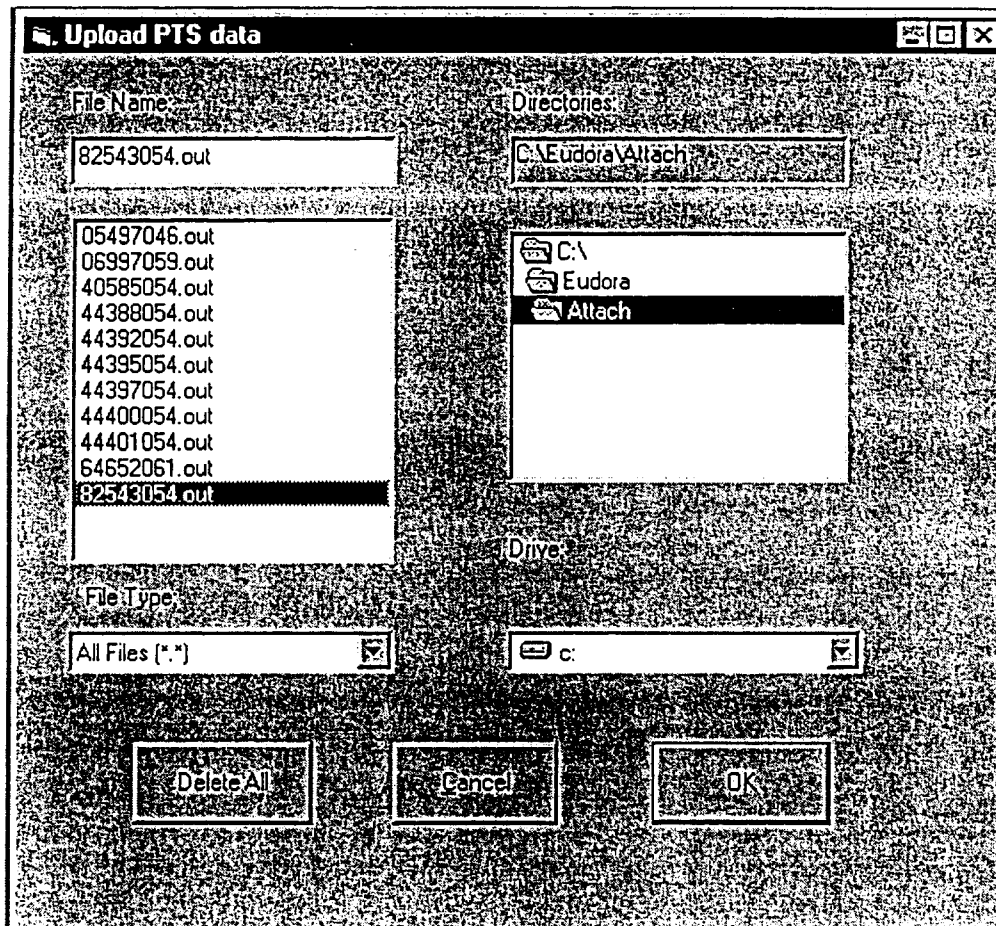


4. Check for messages by clicking on the check mail icon. This should take a while depending on the number of downloads.



5. When completed, go to the Edit menu and choose Select All and hit delete.
6. Close Eudora (the email application) and disconnect from USMS by double clicking on the Dial-up Networking icon in the lower right of the computer and hitting Disconnect.
7. Open Windows Explorer from Start, Programs.
8. Navigate to C:\Eudora\Attach and sort to files by type by clicking on Type in the upper right portion of the screen.
9. Select all the image files (.jpg extension) by selecting the first, holding down Shift and selecting the last image file. Then select Cut from the Edit menu and navigate to 'Network on 'Bronx' Z:\mugs_t' and select Paste from Edit. Overwrite any old mug shots if prompted.
10. Open Rackets and go to the Maintenance screen.

11. Select Upload PTS emails transfers and navigate to C:\Eudora\Attach. Highlight the first message and hit OK, move to the next file and repeat, continuing until all messages have been transferred to the PTS database. Hit Cancel to Exit.



12. Navigate back to C:\Eudora\Attach and delete all files.
13. Back in Rackets, from the main menu select Reports to go to the main Report screen and click the report PTS Arrests (Date Range). From the next screen enter the preceding day as the begin and end dates and hit Print. These will be all the prisoners arrested the previous day and if the "Entered in Rackets" is blank the defendant has not yet been entered from the Rackets end and needs to be. Checked to see if they defendant is in Lions and proceed from there.

Appendix C - RACKETS Script

These scripts are written in Avenues, the programming language for Arcview (the RACKETS system GIS). These codes were specifically written to automate routine tasks within the mapping component of the RACKETS system. The first script, Rackets.LabelThemeUpdate, activates the label button when the scale on the map is less than or equal to 20,000. The Rackets.LabelTheme.Open script opens the label theme dialog box when the user presses the 'label theme' button. The label theme dialog box allows the user to determine what components of the map to label (such as streets, defendant names, agencies, cooperator, etc.) and then automatically labels what the user selects. The Rackets.LocateBy Address script selects all active records within a given radius of a specified address and produces a report and layout (when selected). The last script is used to produce a thematic map from a selected theme (Cooperation Area, Residence, Homicides, Arrest, Crimes) by summarizing the number of events per precinct sector.

Rockets.LabelThemeUpdate

'Colin Reilly

'2/98

'Update script for Rockets.LabelTheme

'Activates lable button if scale of view < 20,000

```
theView = av.GetActiveDoc  
self.SetEnabled(FALSE)
```

```
'get scale  
theScale = theView.ReturnScale
```

```
If (theScale.<=(20000)) Then  
  self.SetEnabled(TRUE)  
End
```

Rockets.LabelTheme.Open

'Colin Reilly

'2/98

'open script for the LabelTheme dialog

```
self.FindByName("lbtCancel").SetTag("FALSE")
```

```
chkStr = self.FindByName("chkStreets")  
cpaStr = self.FindByName("cpaStrThm")  
chkStr.SetListeners({cpaStr})
```

```
tList = {"Residence", "Cooperation Area",  
        "Statutes", "Arrest"}  
self.FindByName("cboRThm").DefineFromList(tList)
```

```
chkRkt = self.FindByName("chkRockets")  
cboR = self.FindByName("cboRThm")  
cpaL = self.FindByName("cpaLabel")  
chkRkt.SetListeners({cboR,cpaL})
```

```
chkRkt.SetSelected(FALSE)  
chkStr.SetSelected(TRUE)
```


Rackets.LocateByAdd

'Colin Reilly 1/98

'This script selects all active records w/in a given radius of a specified address.

'Produces a report and layout when selected

'Dialog = LocateByAddress

thePrj = av.GetProject
theView = av.GetActiveDoc

'clear all existing selections and deactivate all themes
for each t in theView.GetActiveThemes

 t.ClearSelection
 t.SetActive(FALSE)

end

'delete all existing graphics
theView.GetGraphics.Empty
tDisplay = theView.GetDisplay
theView.GetGraphics.Invalidate

'get matchtheme - theme is set to geocode using "US Streets"

strTheme = theView.FindTheme("NYC_geocode_s")

du = tDisplay.GetDistanceUnits

if (strTheme = nil) then

 exit

end

strTheme.SetActive(True)

'query for address to locate
pt = LocateDialog.Show(strTheme)

'if user hits cancel end

if (pt = NIL) then

 Return NIL

End

'if address is not found notify user and end

'get search address from dialog

LocAdd = LocateDialog.GetLastAddress

if (pt.IsNull) then

 MsgBox.Info(LocAdd++"was not found."

 +nl+"Make sure to use proper address format and try again."

 , "Cannot Locate Address")

 Return NIL

end

proj = theView.GetProjection
pp = pt.ReturnProjected(proj)
g = GraphicShape.Make(pp)

'query user for all necessary info

tDialog = av.FindDialog("LocateByAddress")

tDialog.Open

```

'If user hits cancel end script
If (tDialog.FindByName("lbtCancel").GetTag = "TRUE") then
    Return NIL
End

'get radius from dialog, if nil end
r = tDialog.FindByName("sldRadius").GetValue
if((r = nil) or (r = 0)) then
    Return NIL
end

'convert miles to decimal degrees
r2 = Units.Convert (r, #UNITS_LINEAR_MILES, #UNITS_LINEAR_DEGREES)

'make the circle
c = Circle.Make(pt,r2)
gc = GraphicShape.Make(c)
cSym = gc.GetSymbol
cSym.SetStyle(#RASTERFILL_STYLE_EMPTY)
cSym.SetOLColor(Color.GetBlack)
cSym.SetOLWidth(2)
tGList = theView.GetGraphics
theView.GetGraphics.Add(gc)
strTheme.SetActive(False)

'select features from active themes
thmList = {theView.FindTheme("RESIDENCE"),theView.FindTheme("STATUTES"),
           theView.FindTheme("ARREST"),theView.FindTheme("Cooperation Area")}

if (thmList.Count = 0) then
    msgBox.Warning("Themes were not found.,"Script will end!")
    Return NIL
end
for each t in thmList
    t.SetActive(True)
    t.SelectByShapes({c},#VTAB_SELTYPE_NEW)
end
rb = gc.GetBounds
if (rb.IsEmpty) then
    return nil
elseif ( rb.ReturnSize = (0@0) ) then
    theView.GetDisplay.PanTo(rb.ReturnOrigin)
else
    theView.GetDisplay.SetExtent(rb.Scale(1.1))
end

'check dialog to produce a layout
go = tDialog.FindbyName("cpaLayout").GetSelected
If (go.AsString = "radLayoutYes") Then
    'creat a new layout
    newLayout = Layout.Make
    tempLst = TemplateMgr.GetTemplates
    'Get the layout template - "LocateAddress"
    '- referred to by its number in the list of templates
    t = tempLst.Get(5)
    if (t <> nil) then

```

```

    today = Date.Now.SetFormat("MMMM dd, yyyy").AsString
    txtLayout =
{LocAdd,r.AsString,tDialog.FindByName("txtRequest").GetText,today}
    objs = newLayout.GetGraphics.GetFrameObjs
    objs.Merge(av.GetProject.GetDocs)
    objs.Merge(txtLayout)
    newLayout.UseTemplate(t)
    newLayout.GetGraphics.FillFrames(objs)
    newLayout.SetName(tDialog.FindByName("txtRequest").GetText
+Date.Now.SetFormat("MMddyy").AsString)
    lwin = newLayout.GetWin
    lwin.Open
    lwin.MoveTo(2,400)
    lwin.Resize(320,275)
Else
End
End

'check dialog to run report
cont = tDialog.FindByName("cpaReport").GetSelected

if (cont.AsString = "radReportYes") then
    theView.GetWin.Maximize
    'Generate reports
    rptTheme = tDialog.FindByName("cboReport").GetSelection
    vt = theView.FindTheme(rptTheme.AsString).GetFTab
    vt.Export("c:\temp\temp_rpt.dbf".AsFileName,dBASE,TRUE)
    ' Build dictionary to pass parameters to report
    rinDict = Dictionary.Make(5)
    rinDict.Add("RI_LIBRARY","d:\rackets_gis\reports\rackets.rp5")

    'run report based on theme selected
    If (rptTheme.AsString = "Residence") Then
        rpt = "residences"
    ElseIf (rptTheme.AsString = "Cooperation Area") Then
        rpt = "cooperators"
    End

    rinDict.Add("RI_REPORT", rpt)
    rinDict.Add("RI_TITLE",LocAdd)
    rinDict.Add("RI_PRINTER","D")
    rinDict.Add("RI_MASTER","c:\temp\temp_rpt.dbf")
    rinDict.Add("RI_STATFILE","c:\temp\rout.txt")
    'Call the main script, passing the dictionary object above
    av.run("PTRuntime.RunRpt",rinDict)
Else
end

```

```

'Colin Reilly 3/98
'generates a thematic map from the selected theme by summarizing the number of
'events per precinct sector

tView = av.GetActiveDoc
thmLst = {"Cooperation Area", "Residence", "Homicides", "Arrest", "Crimes"}
selThm = msgBox.ChoiceAsString(thmLst, "Select the theme to create a thematic map
from",
    "Thematic Map")
If (selThm = nil) then
    RETURN NIL
Else
End

'get selected theme
tTheme = tView.FindTheme(selThm.AsString)
toVTab = tTheme.GetFTab
toField = toVTab.FindField("Shape")

'get sector theme
frmThm = tView.FindTheme("bxmnsec")
fromThm = frmThm.Clone
fromVTab = fromThm.GetFTab
fromField = fromVTab.FindField("Shape")

toVTab.Join(toField, fromVTab, fromField)

secField = toVTab.FindField("Label")

'summarize vtab by sector
tDir = "c:\temp\".AsFileName
fName = tDir.MakeTmp("sum", "dbf")
fVTab = toVTab.Summarize(fName, dbase, secField, {toField},
{#VTAB_SUMMARY_COUNT})

'make a copy of bxmansec to create thematic map from
fName2 = tDir.MakeTmp("thmmap", "shp")
thmFTab = fromVTab.Export(fName2, Shape, FALSE)
sn = SrcName.Make(fName2.AsString)
newTheme = Theme.Make(sn)
newTheme.SetName (tTheme.GetName++"per Sector")
newVTab = newTheme.GetFTab
tFld = newVTab.FindField("Label")

fFld = fVTab.FindField("Label")
newVTab.Join(tFld, fVTab, fFld) - joined summarized vtab to new sector theme

'add a new count field and return zero for those sectors with no data
shpFld = newVTab.FindField("shape")
countFld = newVTab.FindField("Count")
idFld = newVTab.FindField("Id")
newVTab.SetEditable(TRUE)
cFld = Field.Make("Count2", #FIELD_BYTE, 4, 0)
newVTab.AddFields({cFld})

'if count field is null set value to 0
For each s in newVTab

```

```

x = newVTab.ReturnValueNumber(idFld,s)
z = newVTab.ReturnValueNumber(countFld,s)
If (Not(x = 0) and (z.IsNull)) Then
    newVTab.SetValueNumber(cFld,s,0)
ElseIf (Not(x = 0) and (Not(z.IsNull))) Then
    newVTab.SetValueNumber(cFld,s,z)
End
End
'newVTab.SetEditable(FALSE)

'set legend
tLgnd = newTheme.GetLegend
tLgnd.SetLegendType(#LEGEND_TYPE_COLOR)
'use new count field for legend and divide into 5 classes
tLgnd.Natural(newTheme,"Count2",5)
'use - yellow to orange to red color scheme
theColorRamp = SymbolList.GetPreDefined(#SYMLIST_TYPE_COLORRAMP).Get(19)
tLgnd.GetSymbols.RampSavedColors(theColorRamp)
tLgnd.DisplayNoDataClass(FALSE)
'do not display no data class
newTheme.UpdateLegend

'add theme to view
tView.AddTheme ( newTheme )
NewTheme.SetVisible( TRUE )
NewTheme.SetActive(TRUE)
tView.Invalidate
toVTab.UnjoinAll

'zoom to bronx and manhattan
theThemes = tView.GetActiveThemes
r = Rect.MakeEmpty
for each t in theThemes
    r = r.UnionWith(t.ReturnExtent)
end

if (r.IsEmpty) then
    RETURN NIL
elseif ( r.ReturnSize = (0@0) ) then
    tView.GetDisplay.PanTo(r.ReturnOrigin)
else
    tView.GetDisplay.SetExtent(r.Scale(1.1))
end

```

Appendix D - Microsoft Access Visual Basic Code (pg 1-65)

These codes create a user-friendly interface (Access forms) to facilitate data entry and querying in the underlying data tables. In general, the codes are used to search for individual records, to create new records, to close out records and open new ones, to select the appropriate entries under certain categories, to query the data, and to let the data entry person know if a defendant or case is already in the system. The underlying codes provide the basis for a relatively simplistic data entry format.

Code

```

1 Attribute VB_Name = "Form_CASE"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8
9 Private Sub AUSA_NotInList(NewData As String, Response As Integer)
10
11 Dim strMessage As String
12 Dim dbsRkts As Database
13 Dim rstTypes As Recordset
14
15 strMessage = "Add '" & NewData & "to the list of AUSAs?"
16
17 If Confirm(strMessage) Then
18
19     'open table and add record
20     Set dbsRkts = CurrentDb()
21     Set rstTypes = dbsRkts.OpenRecordset("ausa")
22     rstTypes.AddNew
23     rstTypes!AUSA = NewData
24     rstTypes.Update
25     Response = acDataErrAdded 'requery list
26 Else
27     Response = acDataErrDisplay ' display error
28 End If
29
30 End Sub
31
32 Sub Combo15_AfterUpdate()
33     ' Find the record that matches the control.
34     If IsNull(Me![Combo15]) Then
35         Exit Sub
36     End If
37
38     Me.RecordsetClone.FindFirst "[CaseKey] = " & Me![Combo15]
39     Me.Bookmark = Me.RecordsetClone.Bookmark
40 End Sub
41
42
43
44 Private Sub Command50_Click()
45
46 End Sub
47
48 Private Sub Combo15_BeforeUpdate(Cancel As Integer)
49
50 End Sub
51
52 Private Sub Form_Current()
53     Combo15 = CaseKey
54 End Sub
55 Private Sub cmdClose_Click()
56 On Error GoTo Err_cmdClose_Click
57
58
59     DoCmd.Close
60
61 Exit_cmdClose_Click:
62     Exit Sub
63
64 Err_cmdClose_Click:
65     MsgBox Err.Description

```

```

66     Resume Exit_cmdClose_Click
67
68 End Sub
69 Private Sub cmdPrevious_Click()
70 On Error GoTo Err_cmdPrevious_Click
71
72
73     DoCmd.GoToRecord , , acPrevious
74
75 Exit_cmdPrevious_Click:
76     Exit Sub
77
78 Err_cmdPrevious_Click:
79     MsgBox Err.Description
80     Resume Exit_cmdPrevious_Click
81
82 End Sub
83 Private Sub cmdNext_Click()
84 On Error GoTo Err_cmdNext_Click
85
86
87     DoCmd.GoToRecord , , acNext
88
89 Exit_cmdNext_Click:
90     Exit Sub
91
92 Err_cmdNext_Click:
93     MsgBox Err.Description
94     Resume Exit_cmdNext_Click
95
96 End Sub
97 Private Sub cmdAddRecord_Click()

```


Form: CASE

```

 98 On Error GoTo Err_cmdAddRecord_Click
 99
100
101     DoCmd.GoToRecord , , acNewRec
102     Me!USAONum.SetFocus
103
104 Exit_cmdAddRecord_Click:
105     Exit Sub
106
107 Err_cmdAddRecord_Click:
108     MsgBox Err.Description
109     Resume Exit_cmdAddRecord_Click
110
111 End Sub
112 Private Sub cmdArchive_Click()
113 On Error GoTo Err_cmdArchive_Click
114
115     Dim stDocName As String
116     Dim stDocName1 As String
117     Dim stDocName2 As String
118     Dim stDocName3 As String
119     Dim stDocName4 As String
120     Dim stDocName5 As String
121     Dim stDocName6 As String
122
123     stDocName = "arcCASE"
124     stDocName1 = "arcDEFENDANT"
125     stDocName2 = "arcALIAS"
126     stDocName3 = "arcSTATUTES"
127     stDocName4 = "arcEVENTS"
128     stDocName5 = "arcGROUPS"
129     stDocName6 = "delCASE"
130
131     DoCmd.OpenQuery stDocName, acNormal, acEdit
132     DoCmd.OpenQuery stDocName1, acNormal, acEdit
133     DoCmd.OpenQuery stDocName2, acNormal, acEdit
134     DoCmd.OpenQuery stDocName3, acNormal, acEdit
135     DoCmd.OpenQuery stDocName4, acNormal, acEdit
136     DoCmd.OpenQuery stDocName5, acNormal, acEdit
137     DoCmd.OpenQuery stDocName6, acNormal, acEdit
138
139 Exit_cmdArchive_Click:
140     Exit Sub
141
142 Err_cmdArchive_Click:
143     MsgBox Err.Description
144     Resume Exit_cmdArchive_Click
145
146 End Sub
147
148 Private Sub Form_Error(DataErr As Integer, Response As Integer)
```

Form: CASE

```
149
150 Const conErrFieldRequired = 3314
151
152 If DataErr = conErrFieldRequired Then
153     DisplayMessage "You must enter a Case name and USAO number to save the record."

154     Response = acDataErrContinue
155 Else
156     Response = acDataErrDisplay
157 End If
158
159 End Sub
160
161 Private Sub Form_Load()
162 End Sub
```

Code

```
1 Attribute VB_Name = "Form_COOP_AREA"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Topic_NotInList(NewData As String, Response As Integer)
9 Dim strMessage As String
10 Dim dbsRkts As Database
11 Dim rstTypes As Recordset
12
13 strMessage = "Add '" & NewData & "' to the list of Activities?"
14
15 If Confirm(strMessage) Then
16
17     'open table and add record
18     Set dbsRkts = CurrentDb()
19     Set rstTypes = dbsRkts.OpenRecordset("coop_activity")
20     rstTypes.AddNew
21     rstTypes!Activity = NewData
22     rstTypes.Update
23     Response = acDataErrAdded 'requery list
24 Else
25     Response = acDataErrDisplay ' display error
26 End If
27
28 End Sub
```

Form: COOP_PEOPLEsubArea

Code

```
1 Attribute VB_Name = "Form_COOP_PEOPLEsubArea"  
2 Attribute VB_Creatable = True  
3 Attribute VB_PredeclaredId = True  
4 Attribute VB_Exposed = False  
5 Option Compare Database  
6 Option Explicit  
7  
8  
9  
10 Private Sub Form_BeforeUpdate(Cancel As Integer)  
11 Me![CoopKey] = Me.Parent![CoopKey]  
12 End Sub
```

Code

```

1 Attribute VB_Name = "Form_COOPERATORS"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cboFindName_AfterUpdate()
9
10 Me.RecordsetClone.FindFirst "[DefKey] = " & Me![cboFindName]
11 Me.Bookmark = Me.RecordsetClone.Bookmark
12
13
14 End Sub
15
16 Private Sub cmdClose_Click()
17 On Error GoTo Err_cmdClose_Click
18
19
20     DoCmd.Close
21
22 Exit_cmdClose_Click:
23     Exit Sub
24
25 Err_cmdClose_Click:
26     MsgBox Err.Description
27     Resume Exit_cmdClose_Click
28
29 End Sub
30
31 Private Sub LName_AfterUpdate()
32
33 'Fill in Last name from defendant table
34 strFilter = "Me!DefKey = Forms!DEFENDANT!DefKey"
35 Me!LName = DLookup("[DefLName]", "DEFENDANT", strFilter)
36
37 End Sub
38
39 Private Sub LName_BeforeUpdate(Cancel As Integer)
40
41 End Sub
42 Private Sub cmdArchive_Click()
43 On Error GoTo Err_cmdArchive_Click
44
45     Dim stDocName As String
46     Dim stDocName1 As String
47
48     Dim stDocName2 As String
49     Dim stDocName3 As String
50     Dim stDocName4 As String
51     Dim stDocName5 As String
52
53     stDocName = "arcCOOPERATOR"
54     stDocName1 = "arcCOOP_AREA"
55     stDocName2 = "arcCOOP_GROUP"
56     stDocName3 = "arcHOMICIDE"
57     stDocName4 = "arcNARCOTIC"
58     stDocName5 = "delCOOPERATOR"
59
60     DoCmd.OpenQuery stDocName, acNormal, acEdit
61     DoCmd.OpenQuery stDocName1, acNormal, acEdit
62     DoCmd.OpenQuery stDocName2, acNormal, acEdit
63     DoCmd.OpenQuery stDocName3, acNormal, acEdit
64     DoCmd.OpenQuery stDocName4, acNormal, acEdit
65     DoCmd.OpenQuery stDocName5, acNormal, acEdit

```

```
65
66 Exit_cmdArchive_Click:
67     Exit Sub
68
69 Err_cmdArchive_Click:
70     MsgBox Err.Description
71     Resume Exit_cmdArchive_Click
72
73 End Sub
74
75 Private Sub Combo42_AfterUpdate()
76
77 End Sub
78
79 Private Sub Form_Current()
80
81 'Synchronize combobox and records
82 cboFindName = DefKey
83
84 End Sub
85 Private Sub cmdPrevious_Click()
86 On Error GoTo Err_cmdPrevious_Click
87
88     DoCmd.GoToRecord , , acPrevious
89
90
91 Exit_cmdPrevious_Click:
92     Exit Sub
93
94 Err_cmdPrevious_Click:
95     MsgBox Err.Description
96     Resume Exit_cmdPrevious_Click
97
98 End Sub
99 Private Sub cmdNext_Click()
100 On Error GoTo Err_cmdNext_Click
101
102
103     DoCmd.GoToRecord , , acNext
104
105 Exit_cmdNext_Click:
106     Exit Sub
107
108 Err_cmdNext_Click:
109     MsgBox Err.Description
110     Resume Exit_cmdNext_Click
111
112 End Sub
113 Private Sub cmdFirst_Click()
114 On Error GoTo Err_cmdFirst_Click
115
116
117     DoCmd.GoToRecord , , acFirst
118
119 Exit_cmdFirst_Click:
120     Exit Sub
121
122 Err_cmdFirst_Click:
123     MsgBox Err.Description
124     Resume Exit_cmdFirst_Click
125
126 End Sub
127 Private Sub cmdLast_Click()
128 On Error GoTo Err_cmdLast_Click
129
130
131     DoCmd.GoToRecord , , acLast
132
```

133 Exit_cmdLast_Click:
Form: COOPERATORS

```
134     Exit Sub
135
136 Err_cmdLast_Click:
137     MsgBox Err.Description
138     Resume Exit_cmdLast_Click
139
140 End Sub
141 Private Sub cmdPrntReport_Click()
142 On Error GoTo Err_cmdPrntReport_Click
143
144     Dim stDocName As String
145
146     stDocName = "COOPERATORS"
147     DoCmd.OpenReport stDocName, acNormal
148
149 Exit_cmdPrntReport_Click:
150     Exit Sub
151
152 Err_cmdPrntReport_Click:
153     MsgBox Err.Description
154     Resume Exit_cmdPrntReport_Click
155
156 End Sub
```

Code

```
1 Attribute VB_Name = "Form_COOPERATORSsub"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Topic_NotInList(NewData As String, Response As Integer)
9
10 'If item is not in list, check then add to list if requested
11 Dim strMessage As String
12 Dim dbsRkts As Database
13 Dim rstTypes As Recordset
14
15 strMessage = "Add '" & NewData & "' to the list of Main Topics?"
16
17 If Confirm(strMessage) Then
18
19     'open table and add record
20     Set dbsRkts = CurrentDb()
21     Set rstTypes = dbsRkts.OpenRecordset("coopTopic")
22     rstTypes.AddNew
23     rstTypes!Topic = NewData
24     rstTypes.Update
25     Response = acDataErrAdded 'requery list
26 Else
27     Response = acDataErrDisplay ' display error
28 End If
29
30 End Sub
```


Code

```

1 Attribute VB_Name = "Form_COOPERATORsubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdOpenCoopFrm_Click()
9 On Error GoTo Err_cmdOpenCoopFrm_Click
10
11     'Save record first
12     DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
13
14     Dim stDocName As String
15     Dim stLinkCriteria As String
16     'Dim stMsg As String
17     'Dim stTitle As String
18
19     'Check to see if there is a valid record selected before opening form
20     'If IsNull(Me![CoopKey]) Then 'Notify user and then exit sub
21     '    stMsg = "You must enter basic Cooperator information before you can select
this option"
22     '    stTitle = "Select a Record and then Continue"
23     '    If MsgBox(stMsg, vbOKOnly, stTitle) = vbOK Then
24     '        Exit Sub
25     '    End If
26     'End If
27
28     stDocName = "COOPERATORS"
29
30     stLinkCriteria = "[CoopKey]=" & Me![CoopKey]
31     DoCmd.OpenForm stDocName, , , stLinkCriteria
32
33 Exit_cmdOpenCoopFrm_Click:
34     Exit Sub
35
36 Err_cmdOpenCoopFrm_Click:
37     MsgBox Err.Description
38     Resume Exit_cmdOpenCoopFrm_Click
39
40 End Sub
41
42 Private Sub Form_Current()
43
44 Me!CoopDate.SetFocus
45
46 End Sub
47
48 Private Sub Form_Unload(Cancel As Integer)
49
50 Me!CoopDate.SetFocus
51 Me.Repaint
52
53 End Sub

```

Code

```
1 Attribute VB_Name = "Form_CurrentPrisoners"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Combo21_BeforeUpdate(Cancel As Integer)
9
10 End Sub
11
12
13 Private Sub Form_Current()
14
15 'Synchronize combobox with form
16 Combo21 = Me!FullName
17
18 End Sub
19
20
21 Private Sub Form_Load()
22
23 End Sub
24
25 Private Sub cmdImportRackets_Click()
26 On Error GoTo Err_cmdImportRackets_Click
27
28 'Confirm transfer
29 Dim strMsg As String
30 strMsg = "You are about to import a record from PTS to RACKETS, Continue?"
31 If MsgBox(strMsg, vbOKCancel, "Error!") = vbCancel Then
32     Exit Sub
33 Else
34 End If
35
36 End Sub
37
38 Private Sub Form_Open(Cancel As Integer)
39
40 Me.pr_street.SetFocus
41
42 End Sub
43
44 Private Sub FormFooter_Click()
45
46 End Sub
47
48 Private Sub FormHeader_Click()
49
50 End Sub
51 Sub Combo21_AfterUpdate()
52     ' Find the record that matches the control.
53     Me.RecordsetClone.FindFirst "[FullName] = '" & Me![Combo21] & "'"
54     Me.Bookmark = Me.RecordsetClone.Bookmark
55 End Sub
56
57
58 Private Sub pr_citizen_code_BeforeUpdate(Cancel As Integer)
59
60 End Sub
61
62 Private Sub pr_citizen_code_Click()
63
64 End Sub
```

```
65 Private Sub cmdUpdateRec_Click()
66 On Error GoTo Err_cmdUpdateRec_Click
67
68     Dim stDocName As String
69     'Dim stDocName2 As String
70
71     'Set dbs = CurrentDb
72     'Set rst = dbs.OpenRecordset("DEFENDANT", dbOpenDynaset)
73
74     stDocName = "qUpdateWPTS"
75     'stDocName2 = "qApndArrest"
76
77     DoCmd.OpenQuery stDocName, acNormal, acEdit
78
79 Exit_cmdUpdateRec_Click:
80     Exit Sub
81
82 Err_cmdUpdateRec_Click:
83     MsgBox Err.Description
84     Resume Exit_cmdUpdateRec_Click
85
86 End Sub
87 Private Sub cmdNext_Click()
88 On Error GoTo Err_cmdNext_Click
89
90
91     DoCmd.GoToRecord , , acNext
92
93 Exit_cmdNext_Click:
94     Exit Sub
95
96 Err_cmdNext_Click:
97     MsgBox Err.Description
98
99     Resume Exit_cmdNext_Click
100 End Sub
101 Private Sub cmdPrevious_Click()
102 On Error GoTo Err_cmdPrevious_Click
103
104
105     DoCmd.GoToRecord , , acPrevious
106
107 Exit_cmdPrevious_Click:
108     Exit Sub
109
110 Err_cmdPrevious_Click:
111     MsgBox Err.Description
112     Resume Exit_cmdPrevious_Click
113
114 End Sub
115 Private Sub cmdClose_Click()
116 On Error GoTo Err_cmdClose_Click
117
118
119     DoCmd.Close
120
121 Exit_cmdClose_Click:
122     Exit Sub
123
124 Err_cmdClose_Click:
125     MsgBox Err.Description
126     Resume Exit_cmdClose_Click
127
128 End Sub
129 Private Sub cmdArrest_Click()
```

```
130 On Error GoTo Err_cmdArrest_Click
131
132     Dim stDocName As String
133
134     stDocName = "qApndArrest"
135     DoCmd.OpenQuery stDocName, acNormal, acEdit
136
137 Exit_cmdArrest_Click:
138     Exit Sub
139
140 Err_cmdArrest_Click:
141     MsgBox Err.Description
142     Resume Exit_cmdArrest_Click
143
144 End Sub
```

Code

```

1 Attribute VB_Name = "Form_DEFENDANT"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cboFindName_BeforeUpdate(Cancel As Integer)
9
10 End Sub
11
12 Private Sub cmdClose_Click()
13 On Error GoTo Err_cmdClose_Click
14
15
16     DoCmd.Close
17
18 Exit_cmdClose_Click:
19     Exit Sub
20
21 Err_cmdClose_Click:
22     MsgBox Err.Description
23     Resume Exit_cmdClose_Click
24
25 End Sub
26
27 Private Sub cmdSubmit_Click()
28
29 Dim rst As Recordset
30
31     Dim strCriteria As String
32     Dim strNYSID As String
33     Dim strUSMS As String
34
35     'Make sure there is an entry in the text box before running
36     If IsNull(txtFindBy) Then
37         MsgBox "Enter a NYSID or USMS numer and then submit query"
38         Exit Sub
39     End If
40
41 Select Case grpFindOptions
42
43
44     Case 1 'Find case using NYSID number
45         Set rst = Me.RecordsetClone
46         strNYSID = Me!txtFindBy
47
48         strCriteria = "[NYSID]='" & strNYSID & "'"
49         rst.FindFirst strCriteria
50         If rst.NoMatch Then
51             MsgBox "The number you entered is not valid."
52         Else
53             Me.Bookmark = Me.RecordsetClone.Bookmark
54         End If
55     Case 2 'Find case using USMS number
56         Set rst = Me.RecordsetClone
57         strUSMS = Me!txtFindBy
58         strCriteria = "[USMSNum]='" & strUSMS & "'"
59         rst.FindFirst strCriteria
60         If rst.NoMatch Then
61             MsgBox "The number you entered is not valid."
62         Else
63             Me.Bookmark = Me.RecordsetClone.Bookmark
64         End If

```

```
65         Me!txtFindBy.Value = ""
66 End Select
67
68 End Sub
69
70 Sub cboFindName_AfterUpdate()
71
72     Dim rst As Recordset
73
74     ' Find the record that matches the control.
75     If IsNull(Me![cboFindName]) Then
76         Exit Sub
77     End If
78
79     Set rst = Me.RecordsetClone
80     rst.FindFirst "[DefKey] = " & Me![cboFindName]
81     'Me.RecordsetClone.FindFirst "[DefKey] = " & Me![cboFindName]
82     If rst.NoMatch Then
83         rst.Close
84     End If
85     Me.Bookmark = rst.Bookmark
86 End Sub
87
88 Private Sub cmdNext_Click()
89 On Error GoTo Err_cmdNext_Click
90
91     DoCmd.GoToRecord , , acNext
92
93 Exit_cmdNext_Click:
94     Exit Sub
95
96 Err_cmdNext_Click:
97     MsgBox Err.Description
98     Resume Exit_cmdNext_Click
99 End Sub
100 Private Sub cmdPrevious_Click()
101 On Error GoTo Err_cmdPrevious_Click
102
103     DoCmd.GoToRecord , , acPrevious
104
105 Exit_cmdPrevious_Click:
106     Exit Sub
107
108 Err_cmdPrevious_Click:
109     MsgBox Err.Description
110     Resume Exit_cmdPrevious_Click
111 End Sub
112
113 Private Sub Coop_AfterUpdate()
114
115     'If cooperator is true set cooperator page visible
116     If (Me!Coop = True) Then
117         Me!tpgCooperator.Visible = True
118     ElseIf (Me!Coop = False) Then
119         Me!tpgCooperator.Visible = False
120     Else
121     End If
122 End Sub
123
124 Private Sub Coop_BeforeUpdate(Cancel As Integer)
```

```
133
134
135 End Sub
136
137 Private Sub Coop_Click()
138
139 End Sub
140
141 Private Sub CrimeType_NotInList(NewData As String, Response As Integer)
142
143 Dim strMessage As String
144 Dim dbsRkts As Database
145 Dim rstTypes As Recordset
146
147 strMessage = "Add '" & NewData & "to the list of Crime Types?"
148
149 If Confirm(strMessage) Then
150
151     'open table and add record
152     Set dbsRkts = CurrentDb()
153     Set rstTypes = dbsRkts.OpenRecordset("crime_type")
154     rstTypes.AddNew
155     rstTypes!CrimeType = NewData
156     rstTypes.Update
157     Response = acDataErrAdded 'requery list
158 Else
159     Response = acDataErrDisplay ' display error
160 End If
161
162 End Sub
163
164 Private Sub Detail_Click()
165
166 End Sub
167
168 Private Sub Form_AfterUpdate()
169
170 'Enable or disable mugshots button
171 If IsNull(Me!pr_key) Or IsNull(Me!txtMugshot) Then
172     Me!cmdPhoto.Enabled = False
173 Else
174     Me!cmdPhoto.Enabled = True
175 End If
176
177 'Enable or disable insert mugshots button
178 If IsNull(Me!txtMugshot) And Not (IsNull(Me!pr_key)) Then
179     Me!cmdEnterPhoto.Enabled = True
180 Else
181     Me!cmdEnterPhoto.Enabled = False
182 End If
183
184 End Sub
185
186 Private Sub Form_BeforeUpdate(Cancel As Integer)
187
188 End Sub
189
190 Private Sub Form_Current()
191
192 'Set Find Record to current
193 cboFindName = DefKey
194
195 'If cooperator is true set cooperator page visible
196 If (Me!Coop = True) Then
197     Me!tpgCooperator.Visible = True
198
199 ElseIf (Me!Coop = False) Then
```

```
200     Me!tpgCooperator.Visible = False
201 Else
202 End If
203
204 'If Weapon seized is true set cooperator page visible
205 If (Me!Weapon = True) Then
206     Me!tpgFIREARMS.Visible = True
207
208 ElseIf (Me!Weapon = False) Then
209     Me!tpgFIREARMS.Visible = False
210 Else
211 End If
212
213 'Enable or disable mugshots button
214 If IsNull(Me!pr_key) Or IsNull(Me!txtMugshot) Then
215     Me!cmdPhoto.Enabled = False
216 Else
217     Me!cmdPhoto.Enabled = True
218 End If
219
220 'Enable or disable insert mugshots button
221 If IsNull(Me!txtMugshot) And Not (IsNull(Me!pr_key)) Then
222     Me!cmdEnterPhoto.Enabled = True
223 Else
224     Me!cmdEnterPhoto.Enabled = False
225 End If
226
227 End Sub
228 Private Sub cmdOpenCooperatorFrm_Click()
229 On Error GoTo Err_cmdOpenCooperatorFrm_Click
230
231     Dim stDocName As String
232     Dim stLinkCriteria As String
233
234     stDocName = "COOPERATORS"
235
236     stLinkCriteria = "[DefKey]=" & Me![DefKey]
237     DoCmd.OpenForm stDocName, , , stLinkCriteria
238
239 Exit_cmdOpenCooperatorFrm_Click:
240     Exit Sub
241
242 Err_cmdOpenCooperatorFrm_Click:
243     MsgBox Err.Description
244     Resume Exit_cmdOpenCooperatorFrm_Click
245
246 End Sub
247 Private Sub cmdOpenCooperatorForm_Click()
248 On Error GoTo Err_cmdOpenCooperatorForm_Click
249
250     Dim stDocName As String
251
252     Dim stLinkCriteria As String
253
254     stDocName = "COOPERATORS"
255
256     stLinkCriteria = "[DefKey]=" & Me!DefKey
257     DoCmd.OpenForm stDocName, , , stLinkCriteria, acAdd, , Me!DefKey
258
259 Exit_cmdOpenCooperatorForm_Click:
260     Exit Sub
261
262 Err_cmdOpenCooperatorForm_Click:
263     MsgBox Err.Description
264     Resume Exit_cmdOpenCooperatorForm_Click
265
```


266 End Sub
Form: DEFENDANT

```
267
268 Private Sub Form_Error(DataErr As Integer, Response As Integer)
269     Debug.Print "DataErr = "; DataErr
270 End Sub
271
272
273 Private Sub Frame99_BeforeUpdate(Cancel As Integer)
274
275 End Sub
276
277 Private Sub Frame112_BeforeUpdate(Cancel As Integer)
278
279 End Sub
280
281
282 Private Sub grpFindOptions_BeforeUpdate(Cancel As Integer)
283
284 End Sub
285
286 Private Sub LeadDef_AfterUpdate()
287
288 End Sub
289
290 Private Sub LeadDef_Click()
291
292 End Sub
293
294 Private Sub txtFindBy_BeforeUpdate(Cancel As Integer)
295
296 End Sub
297 Private Sub cmdOpenCaseFrm_Click()
298 On Error GoTo Err_cmdOpenCaseFrm_Click
299
300     Dim stDocName As String
301     Dim stLinkCriteria As String
302
303     stDocName = "CASE"
304     'stDocName1 = "DEFENDANT"
305
306     stLinkCriteria = "[CaseKey]=" & Me![CaseKey]
307     DoCmd.OpenForm stDocName, , , stLinkCriteria
308
309
310     'Close form
311     DoCmd.Close acForm, "DEFENDANT", acSaveYes
312
313
314 Exit_cmdOpenCaseFrm_Click:
315     Exit Sub
316
317 Err_cmdOpenCaseFrm_Click:
318     MsgBox Err.Description
319     Resume Exit_cmdOpenCaseFrm_Click
320
321 End Sub
322
323 Private Sub Weapon_AfterUpdate()
324
325 'If Weapon seized is true set cooperator page visible
326 If (Me!Weapon = True) Then
327     Me!tpgFIREARMS.Visible = True
328
329 ElseIf (Me!Weapon = False) Then
330     Me!tpgFIREARMS.Visible = False
331 Else
332 End If
```

```
334 End Sub
335
336 Private Sub Weapon_Click()
337
338 End Sub
339 Private Sub cmdPrintRpt_Click()
340 On Error GoTo Err_cmdPrintRpt_Click
341
342     Dim stDocName As String
343
344     stDocName = "DEFENDANT"
345     DoCmd.OpenReport stDocName, acNormal
346
347 Exit_cmdPrintRpt_Click:
348     Exit Sub
349
350 Err_cmdPrintRpt_Click:
351     MsgBox Err.Description
352     Resume Exit_cmdPrintRpt_Click
353
354 End Sub
355 Private Sub cmdPhoto_Click()
356 On Error GoTo Err_cmdPhoto_Click
357
358     Dim stDocName As String
359     Dim stLinkCriteria As String
360
361     stDocName = "frmDEF_PHOTOS"
362
363     stLinkCriteria = "[DefKey]=" & Me![DefKey]
364     DoCmd.OpenForm stDocName, , , stLinkCriteria
365
366     'do not allow records to be edited
367     Forms![frmDEF_PHOTOS].oleMugA.Locked = True
368     Forms![frmDEF_PHOTOS].oleMugB.Locked = True
369
370 Exit_cmdPhoto_Click:
371     Exit Sub
372
373 Err_cmdPhoto_Click:
374     MsgBox Err.Description
375     Resume Exit_cmdPhoto_Click
376
377 End Sub
378 Private Sub cmdEnterPhoto_Click()
379 On Error GoTo Err_cmdEnterPhoto_Click
380
381     Dim stDocName As String
382     Dim stLinkCriteria As String
383
384     stDocName = "frmDEF_PHOTOS"
385
386     stLinkCriteria = "[DefKey]=" & Me![DefKey]
387     DoCmd.OpenForm stDocName, , , stLinkCriteria
388
389     'Allow records to be edited
390     Forms![frmDEF_PHOTOS].oleMugA.Locked = False
391     Forms![frmDEF_PHOTOS].oleMugB.Locked = False
392
393 Exit_cmdEnterPhoto_Click:
394     Exit Sub
395
396 Err_cmdEnterPhoto_Click:
397     MsgBox Err.Description
398     Resume Exit_cmdEnterPhoto_Click
399
```

400 End Sub

Code

```

1 Attribute VB_Name = "Form_DEFENDANTList"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8
9 Private Sub Form_Current()
10
11 'Enable or disable Details button
12 If IsNull(Me![DefKey]) Then
13     Me!cmdOpenDefFrm.Enabled = False
14 Else
15     Me!cmdOpenDefFrm.Enabled = True
16 End If
17
18 'Enable or disable PTS button
19 If IsNull(Me!USMSNum) Then
20     Me!cmdSearchPTS.Enabled = True
21 Else
22     Me!cmdSearchPTS.Enabled = False
23 End If
24 End Sub
25
26 Private Sub Form_Load()
27
28 'Enable or disable Details button
29 If IsNull(Me![DefKey]) Then
30     Me!cmdOpenDefFrm.Enabled = False
31 Else
32     Me!cmdOpenDefFrm.Enabled = True
33 End If
34
35 End Sub
36
37 Private Sub Form_Open(Cancel As Integer)
38
39 End Sub
40 Private Sub cmdOpenDefFrm_Click()
41 On Error GoTo Err_cmdOpenDefFrm_Click
42
43
44     Dim stDocName As String
45     Dim stLinkCriteria As String
46     Dim stTitle As String
47
48     Dim stMsg As String
49     Dim stMsgTrans As String
50     Dim stTitleTrans As String
51     Dim IndctNum As Variant
52     Dim TopChg As Variant
53
54     stDocName = "DEFENDANT"
55
56     stLinkCriteria = "[DefKey]=" & Me![DefKey]
57
58     'Save record first
59 DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
60
61     'Check to see if there is a valid record selected before opening form
62     'If IsNull(Me![DefKey]) Then 'Notify user and then exit sub
63     '    stMsg = "You must be at a valid record before you can select this option"

```

```

64     '     stTitle = "Select a Record and then Continue"
65     '     If MsgBox(stMsg, vbOKOnly, stTitle) = vbOK Then
66     '         Exit Sub
67     '     End If
68     'End If
69
70     'get incictment# and top charge for use in transfer
71     IndctNum = Forms![CASE]!IndictmentNum.Value
72     TopChg = Forms![CASE]!StatuteNum.Value
73
74
75     DoCmd.OpenForm stDocName, , , stLinkCriteria
76
77     'Set focus to Middle Name
78     Forms![Defendant]!DefMName.SetFocus
79
80     'Disable all FindBy options
81     Forms![Defendant]!grpFindOptions.Enabled = False
82     Forms![Defendant]!cboFindName.Enabled = False
83     Forms![Defendant]!cmdSubmit.Enabled = False
84     Forms![Defendant]!txtFindBy.Enabled = False
85     'Remove navigation buttons
86     Forms![Defendant]!cmdPrevious.Visible = False
87     Forms![Defendant]!cmdNext.Visible = False
88
89     'Check if Case Instrument# & TopCharge should be entered for defendant
90     'If IsNull(Forms!Defendant!tpgStatutes.Form.Controls!Count) And
IsNull(Forms!Defendant!tpgStatutes.Form.Controls!StatuteNum) Then
91     '     stMsgTrans = "Enter Case Instrument no. " & IndctNum & " and Top Charge " &
TopChg & " for Defendant?"
92     '     stTitleTrans = "RACKETS"
93     '     If MsgBox(stMsgTrans, vbYesNo, stTitleTrans) = vbYes Then
94     '         Forms![Defendant]![STATUTESsubform].Count.Value = 1
95     '         Forms![Defendant]![STATUTESsubform].StatuteNum.Value = TopChg
96     '     Else
97     '         End If
98     'End If
99
100 Exit_cmdOpenDefFrm_Click:
101     Exit Sub
102
103 Err_cmdOpenDefFrm_Click:
104     MsgBox Err.Description
105     Resume Exit_cmdOpenDefFrm_Click
106
107 End Sub
108 Private Sub cmdSearchPTS_Click()
109 On Error GoTo Err_cmdSearchPTS_Click
110
111     Dim stDocName As String
112     Dim stLinkCriteria As String
113
114     stDocName = "CurrentPrisoners"
115     DoCmd.OpenForm stDocName, , , stLinkCriteria
116
117 Exit_cmdSearchPTS_Click:
118     Exit Sub
119
120 Err_cmdSearchPTS_Click:
121     MsgBox Err.Description
122     Resume Exit_cmdSearchPTS_Click
123
124 End Sub

```

Form: EVENTsubform

Code

```
1 Attribute VB_Name = "Form_EVENTsubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8
9
10 Private Sub Form_AfterInsert()
11
12 Dim strMessage As String
13 Dim strLinkCriteria As String
14
15 strMessage = "Enter Proffer information now? At the very least, a topic must be entered
    to continue."
16
17 If Me.event_type = "Proffer" Then
18     If Forms![Defendant].Coop.Value = False Then
19         Forms![Defendant].Coop.Value = True
20         Forms![Defendant]!tpgCooperator.Visible = True
21     End If
22     Forms![Defendant]!subCooperator.Form.Status = "PROF"
23     'If Confirm(strMessage) Then
24         Forms![Defendant]!tpgCooperator.SetFocus
25         Forms![Defendant]!subCooperator.Form.Topic.SetFocus
26     'Else
27     'End If
28 End If
29
30 End Sub
```

Code

```
1 Attribute VB_Name = "Form_FIREARMSsubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub FirearmType_NotInList(NewData As String, Response As Integer)
9
10 Dim strMessage As String
11 Dim dbsRkts As Database
12 Dim rstTypes As Recordset
13
14 strMessage = "Add '" & NewData & "' to the list of Firearm Types?"
15
16 If Confirm(strMessage) Then
17     'open table and add record
18     Set dbsRkts = CurrentDb()
19     Set rstTypes = dbsRkts.OpenRecordset("firearm_type")
20     rstTypes.AddNew
21     rstTypes!FirearmType = NewData
22     rstTypes.Update
23     Response = acDataErrAdded 'requery list
24 Else
25     Response = acDataErrDisplay ' display error
26 End If
27
28
29
30 End Sub
31
32 Private Sub Make_NotInList(NewData As String, Response As Integer)
33
34 Dim strMessage As String
35 Dim dbsRkts As Database
36 Dim rstTypes As Recordset
37
38 strMessage = "Add '" & NewData & "' to the list of Firearm Makes?"
39
40 If Confirm(strMessage) Then
41
42     'open table and add record
43     Set dbsRkts = CurrentDb()
44     Set rstTypes = dbsRkts.OpenRecordset("firearm_make")
45     rstTypes.AddNew
46     rstTypes!Make = NewData
```

Form: FIREARMSsubform

```
47     rstTypes.Update
48     Response = acDataErrAdded 'requery list
49 Else
50     Response = acDataErrDisplay ' display error
51 End If
52
53
54 End Sub
55
56 Private Sub Model_NotInList(NewData As String, Response As Integer)
57
58 Dim strMessage As String
59 Dim dbsRkts As Database
60 Dim rstTypes As Recordset
61
62 strMessage = "Add '" & NewData & "to the list of Firearm Models?"
63
64 If Confirm(strMessage) Then
65
66     'open table and add record
67     Set dbsRkts = CurrentDb()
68     Set rstTypes = dbsRkts.OpenRecordset("firearm_model")
69     rstTypes.AddNew
70     rstTypes!Model = NewData
71     rstTypes.Update
72     Response = acDataErrAdded 'requery list
73 Else
74     Response = acDataErrDisplay ' display error
75 End If
76
77
78 End Sub
```


Code

```
1 Attribute VB_Name = "Form_frmDEF_PHOTOS"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdPrint_Click()
9 On Error GoTo Err_cmdPrint_Click
10
11
12     DoCmd.PrintOut
13
14 Exit_cmdPrint_Click:
15     Exit Sub
16
17 Err_cmdPrint_Click:
18     MsgBox Err.Description
19     Resume Exit_cmdPrint_Click
20
21 End Sub
22 Private Sub cmdClose_Click()
23 On Error GoTo Err_cmdClose_Click
24
25     'save record first
26     DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
27
28     DoCmd.Close
29
30 Exit_cmdClose_Click:
31     Exit Sub
32
33 Err_cmdClose_Click:
34     MsgBox Err.Description
35     Resume Exit_cmdClose_Click
36
37 End Sub
```

Code

```
1 Attribute VB_Name = "Form_frmDefsByAUSA"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdPreview_Click()
9 On Error GoTo Err_cmdPreview_Click
10
11     Dim stDocName As String
12
13     stDocName = "rptDefsByAUSA"
14     DoCmd.OpenReport stDocName, acPreview
15     DoCmd.Close acForm, "frmDefsByAUSA"
16
17 Exit_cmdPreview_Click:
18     Exit Sub
19
20 Err_cmdPreview_Click:
21     MsgBox Err.Description
22     Resume Exit_cmdPreview_Click
23
24 End Sub
25 Private Sub cmdPrint_Click()
26 On Error GoTo Err_cmdPrint_Click
27
28     Dim stDocName As String
29
30     stDocName = "rptDefsByAUSA"
31     DoCmd.OpenReport stDocName, acNormal
32     DoCmd.Close acForm, "frmDefsByAUSA"
33
34 Exit_cmdPrint_Click:
35     Exit Sub
36
37 Err_cmdPrint_Click:
38     MsgBox Err.Description
39     Resume Exit_cmdPrint_Click
40
41 End Sub
42 Private Sub cmdCancel_Click()
43 On Error GoTo Err_cmdCancel_Click
44
45
46     DoCmd.Close
47
48 Exit_cmdCancel_Click:
49     Exit Sub
50
51 Err_cmdCancel_Click:
52     MsgBox Err.Description
53     Resume Exit_cmdCancel_Click
54
55 End Sub
```

Code

```
1 Attribute VE_Name = "Form_frmDefsByUnit"
2 Attribute VE_Creatable = True
3 Attribute VE_PredeclaredId = True
4 Attribute VE_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdPreview_Click()
9 On Error GoTo Err_cmdPreview_Click
10
11     Dim stDocName As String
12
13     stDocName = "rptDefsByUnit"
14     DoCmd.OpenReport stDocName, acPreview
15     DoCmd.Close acForm, "frmDefsByUnit"
16
17 Exit_cmdPreview_Click:
18     Exit Sub
19
20 Err_cmdPreview_Click:
21     MsgBox Err.Description
22     Resume Exit_cmdPreview_Click
23
24 End Sub
25 Private Sub cmdPrint_Click()
26 On Error GoTo Err_cmdPrint_Click
27
28     Dim stDocName As String
29
30     stDocName = "rptDefsByUnit"
31     DoCmd.OpenReport stDocName, acNormal
32     DoCmd.Close acForm, "frmDefsByUnit"
33
34 Exit_cmdPrint_Click:
35     Exit Sub
36
37 Err_cmdPrint_Click:
38     MsgBox Err.Description
39     Resume Exit_cmdPrint_Click
40
41 End Sub
42 Private Sub cmdCancel_Click()
43 On Error GoTo Err_cmdCancel_Click
44
45
46     DoCmd.Close
47
48 Exit_cmdCancel_Click:
49     Exit Sub
50
51 Err_cmdCancel_Click:
52     MsgBox Err.Description
53     Resume Exit_cmdCancel_Click
54
55 End Sub
```

Code

```

1 Attribute VB_Name = "Form_frmFindAlias"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdClear_Click()
9
10 Dim strSql As String
11 Dim strSqlFull As String
12
13 Me!txtAlias = ""
14
15 strSql = "WHERE FALSE"
16 strSqlFull = "Select * From qFindAlias " & strSql
17 Me!subFindAlias.Form.RecordSource = strSqlFull
18
19 strSql = "Select * From qFindAKA " & strSql
20 Me!subFindAKA.Form.RecordSource = strSqlFull
21
22 End Sub
23
24 Private Sub cmdSubmit_Click()
25
26 Dim strAlias As String
27 Dim strSqlAlias As String
28 Dim strSqlFull As String
29 Dim strMsg As String
30 Dim strTitle As String
31 Dim strWhere As String
32
33 If IsNull(Me!txtAlias) Or Me!txtAlias = "" Then
34     strMsg = "Query is empty. Enter an alias and try again."
35     strTitle = "Empty Query!"
36     If MsgBox(strMsg, vbOKOnly, strTitle) = vbOK Then
37         Exit Sub
38     End If
39 End If
40
41 strWhere = "Where "
42
43 Select Case grpDatabase
44 'Search PTS
45 Case 1
46     If grpMatch.Value = 1 Then
47         strSqlAlias = "([name_fname] = " & "Forms!frmFindAlias![txtAlias]OR[name_lname] = " & "Forms!frmFindAlias![txtAlias]OR[name_mname] = " & "Forms!frmFindAlias![txtAlias]OR[rm_desc] = " & "Forms!frmFindAlias![txtAlias])"
48
49         strSqlAliasSql = strWhere & strSqlAlias
50         strSqlFull = "Select * From qFindAKA " & strSqlAliasSql
51         Me!subFindAKA.Form.RecordSource = strSqlFull
52     Else
53         strSqlAliasSql = " Where [name_fname] Like '" & Forms!frmFindAlias![txtAlias] & "' or [name_lname] Like '" & Forms!frmFindAlias![txtAlias] & "' or [name_mname] Like '" & Forms!frmFindAlias![txtAlias] & "' or [rm_desc] Like '" & Forms!frmFindAlias![txtAlias] & "' or " & "Forms!frmFindAlias![txtAlias] & "'"
54         strSqlFull = "Select * From qFindAKA " & strSqlAliasSql
55         Me!subFindAKA.Form.RecordSource = strSqlFull
56     End If
57 End Sub

```

57 'Search Rackets
Form: frmFindAlias

```
58 Case 2
59     If grpMatch.Value = 1 Then
60         strAlias = "([Alias] = " & "Forms!frmFindAlias![txtAlias]OR[AliasLN] = " &
"Forms!frmFindAlias![txtAlias])"
61         strAliasSql = strWhere & strAlias
62         strFullSql = "Select * From qFindAlias " & strAliasSql
63         Me!subFindAlias.Form.RecordSource = strFullSql
64     Else
65         strAliasSql = " Where [Alias] Like '" & Forms!frmFindAlias![txtAlias] & "' or
[AliasLN] Like '" & Forms!frmFindAlias![txtAlias] & "'"
66         strFullSql = "Select * From qFindAlias " & strAliasSql
67         Me!subFindAlias.Form.RecordSource = strFullSql
68     End If
69 End Select
70
71 End Sub
72
73 Private Sub Form_AfterUpdate()
74
75 End Sub
76
77 Private Sub Form_Load()
78
79 End Sub
80
81 Private Sub Frame4_BeforeUpdate(Cancel As Integer)
82
83 End Sub
84
85 Private Sub grpDatabase_AfterUpdate()
86
87 'Set subform visible based on selection of search
88 Select Case grpDatabase
89
90 Case 1
91 Me!subFindAKA.Form.Visible = True
92 Me!subFindAlias.Form.Visible = False
93
94 Case 2
95 Me!subFindAKA.Form.Visible = False
96 Me!subFindAlias.Form.Visible = True
97
98 End Select
99
100 End Sub
101
102 Private Sub txtAlias_BeforeUpdate(Cancel As Integer)
103
104 End Sub
105 Private Sub cmdClose_Click()
106 On Error GoTo Err_cmdClose_Click
107
108
109     DoCmd.Close
110
111 Exit_cmdClose_Click:
112     Exit Sub
113
114 Err_cmdClose_Click:
115     MsgBox Err.Description
116     Resume Exit_cmdClose_Click
117
118 End Sub
119 Private Sub cmdOpenDefFrm_Click()
120 On Error GoTo Err_cmdOpenDefFrm_Click
121
122     Dim stDocName As String
```

```
123 Dim stLinkCriteria As String
124
Form: frmFindAlias
```

```
125 'stLinkCriteria = Me!subFindAKA.Form.
126
127 stDocName = "DEFENDANT"
128 DoCmd.OpenForm stDocName, , , stLinkCriteria
129
130 Exit_cmdOpenDefFrm_Click:
131 Exit Sub
132
133 Err_cmdOpenDefFrm_Click:
134 MsgBox Err.Description
135 Resume Exit_cmdOpenDefFrm_Click
136
137 End Sub
```

Code

```

1 Attribute VB_Name = "Form_frmFindAssoc"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdClear_Click()
9 Dim strSql As String
10 Dim strFull As String
11
12 Me!txtFind = ""
13
14 strSql = "Where False"
15 strFull = "Select * From qFindAssoc " & strSql
16 Me!subFindAssoc.Form.RecordSource = strFull
17 Me!subFindLocale.Form.RecordSource = strFull
18 Me!subFindCohort.Form.RecordSource = strFull
19
20 End Sub
21
22 Private Sub cmdClose_Click()
23 On Error GoTo Err_cmdClose_Click
24
25
26     DoCmd.Close
27
28 Exit_cmdClose_Click:
29     Exit Sub
30
31 Err_cmdClose_Click:
32     MsgBox Err.Description
33     Resume Exit_cmdClose_Click
34
35 End Sub
36
37 Private Sub cmdSubmit_Click()
38
39
40 Dim strAssoc As String
41 Dim strSql As String
42 Dim strMsg As String
43 Dim strTitle As String
44
45 'clear variables
46 'strAssoc = ""
47
48 'strSql = ""
49
50 'First check to see if a string was entered, if not end routine
51 strMsg = "Query is empty. Enter a Search String and try again"
52 strTitle = "RACKETS"
53 If IsNull(Me!txtFind) Or Me!txtFind = "" Then
54     If MsgBox(strMsg, vbOKOnly, strTitle) = vbOK Then
55         Exit Sub
56     End If
57 End If
58 'run query based on selection type
59 Select Case grpType
60
61 'search Group table via qFindAssoc
62 Case 1
63     strAssoc = "Where [GroupName] Like '" & Forms!frmFindAssoc![txtFind] & " '*'"
64     strSql = "Select * From qFindAssoc " & strAssoc

```

```

65     Me!subFindAssoc.Form.RecordSource = strSql
66
67 'search locale via q FindLocale
68 Case 2
69
70     strAssoc = "Where [lo_name] Like '" & Forms!frmFindAssoc![txtFind] & "'"
71     strSql = "Select * From qFindLocale " & strAssoc
72     Me!subFindLocale.Form.RecordSource = strSql
73
74 'search cohort via qFindCohort
75 Case 3
76     strAssoc = "Where [co_lname] Like '" & Forms!frmFindAssoc![txtFind] & "'" or
[co_lname] Like '" & Forms!frmFindAssoc![txtFind] & "'"
77     strSql = "Select * From qFindCohort " & strAssoc
78     Me!subFindCohort.Form.RecordSource = strSql
79
80 End Select
81 End Sub
82
83 Private Sub Form_Close()
84 Me!txtFind = ""
85 End Sub
86
87 Private Sub Form_Open(Cancel As Integer)
88
89 Dim strSql As String
90 Dim strFull As String
91
92 'Clear last query
93 Me!txtFind = ""
94
95 strSql = "Where False"
96 strFull = "Select * From qFindAssoc " & strSql
97
98 Me!subFindAssoc.Form.RecordSource = strFull
99
100 'Set default visible and all others not visible
101 Me!subFindAssoc.Form.Visible = True
102 Me!subFindCohort.Form.Visible = False
103 Me!subFindLocale.Form.Visible = False
104
105 End Sub
106
107 Private Sub grpType_AfterUpdate()
108
109 'Set the subform based on the query type
110 Select Case grpType
111
112 Case 1
113     Me!subFindAssoc.Form.Visible = True
114     Me!subFindLocale.Form.Visible = False
115     Me!subFindCohort.Form.Visible = False
116
117 Case 2
118     Me!subFindLocale.Form.Visible = True
119     Me!subFindAssoc.Form.Visible = False
120     Me!subFindCohort.Form.Visible = False
121
122 Case 3
123     Me!subFindCohort.Form.Visible = True
124     Me!subFindAssoc.Form.Visible = False
125     Me!subFindLocale.Form.Visible = False
126
127 End Select
128
129 End Sub

```


Code

```
1 Attribute VB_Name = "Form_frmFindCounsel"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cboFind_AfterUpdate()
9
10 Dim strSql As String
11 Dim strFull As String
12
13 strSql = "Select * From qFindCounsel Where [Counsel] = " &
  "Forms!frmFindCounsel![cboFind]"
14 Me!subFindCounsel.Form.RecordSource = strSql
15
16
17 End Sub
18
19 Private Sub cboFind_BeforeUpdate(Cancel As Integer)
20
21 End Sub
22
23 Private Sub cboFind_Change()
24 Dim strSql As String
25 Dim strFull As String
26
27 'strSql = "Select * From qFindCounsel Where [] = " & "Forms!frmFindCounsel![cboFind]"
28 'Me!subFindCounsel.Form.RecordSource = strSql
29
30
31 End Sub
32
33 Private Sub Detail_Click()
34
35 End Sub
36 Private Sub cmdClose_Click()
37 On Error GoTo Err_cmdClose_Click
38
39
40     DoCmd.Close
41
42 Exit_cmdClose_Click:
43     Exit Sub
44
45 Err_cmdClose_Click:
46     MsgBox Err.Description
47     Resume Exit_cmdClose_Click
48
49 End Sub
```

Code

```

1 Attribute VB_Name = "Form_frmFindEmployer"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdClear_Click()
9
10 Dim strSql As String
11 Dim strFull As String
12
13 Me!txtFind = ""
14
15 strSql = "Where False"
16 strFull = "Select * From qFindEmployer " & strSql
17 Me!subFindEmployer.Form.RecordSource = strFull
18 Me!subFindEmployerPTS.Form.RecordSource = strFull
19
20 End Sub
21
22 Private Sub cmdSubmit_Click()
23
24 Dim strEmploy As String
25 Dim strSql As String
26 Dim strMsg As String
27 Dim strTitle As String
28
29
30 'First check to see if a string was entered, if not end routine
31 strMsg = "Query is empty. Enter a Search String and try again"
32 strTitle = "RACKETS"
33 If IsNull(Me!txtFind) Or Me!txtFind = "" Then
34     If MsgBox(strMsg, vbOKOnly, strTitle) = vbOK Then
35         Exit Sub
36     End If
37 End If
38
39 'run query based on selection type
40 Select Case grpType
41
42 'search by employer
43 Case 1
44
45     If grpSearch.Value = 1 Then
46         strEmploy = "Where [em_name] Like '" & Forms!frmFindEmployer![txtFind] & "'"
47
48         strSql = "Select * From qFindEmployer " & strEmploy
49         Me!subFindEmployer.Form.RecordSource = strSql
50     Else
51         strEmploy = "Where [em_name] Like '" & Forms!frmFindEmployer![txtFind] & "'"
52
53         strSql = "Select * From qFindEmployPTS " & strEmploy
54         Me!subFindEmployerPTS.Form.RecordSource = strSql
55     End If
56
57 'search by owner/supervisor
58 Case 2
59
60     If grpSearch.Value = 1 Then
61         strEmploy = "Where [em_poc] Like '" & Forms!frmFindEmployer![txtFind] & "'"
62
63         strSql = "Select * From qFindEmployer " & strEmploy
64         Me!subFindEmployer.Form.RecordSource = strSql
65     Else
66         strEmploy = "Where [em_poc] Like '" & Forms!frmFindEmployer![txtFind] & "'"

```

```

63         strSQL = "Select * From qFindEmployPTS " & strEmploy
64         Me!subFindEmployerPTS.Form.RecordSource = strSQL
65     End If
66
67     'search by occupation
68     Case 3
69         If grpSearch.Value = 1 Then
70             strEmploy = "Where [em_occ] Like '" & Forms!frmFindEmployer![txtFind] & "''"
71
72             strSQL = "Select * From qFindEmployer " & strEmploy
73             Me!subFindEmployer.Form.RecordSource = strSQL
74         Else
75             strEmploy = "Where [em_occ] Like '" & Forms!frmFindEmployer![txtFind] & "''"
76
77             strSQL = "Select * From qFindEmployPTS " & strEmploy
78             Me!subFindEmployerPTS.Form.RecordSource = strSQL
79         End If
80     End Select
81
82 End Sub
83
84 Private Sub Form_Open(Cancel As Integer)
85
86 Me!subFindEmployer.Form.Visible = True
87 Me!subFindEmployerPTS.Form.Visible = False
88
89 End Sub
90
91 Private Sub grpSearch_AfterUpdate()
92
93 'Set subform based on search selection
94
95 Select Case grpSearch
96 Case 1
97 'Search records linked to defendants in Rackets
98 Me!subFindEmployer.Form.Visible = True
99 Me!subFindEmployerPTS.Form.Visible = False
100
101 Case 2
102 'Search records not linked to an active defendant
103 Me!subFindEmployer.Form.Visible = False
104 Me!subFindEmployerPTS.Form.Visible = True
105
106 End Select
107 End Sub
108
109 Private Sub grpSearch_BeforeUpdate(Cancel As Integer)
110
111 End Sub
112 Private Sub cmdClose_Click()
113 On Error GoTo Err_cmdClose_Click
114
115
116     DoCmd.Close
117
118 Exit_cmdClose_Click:
119     Exit Sub
120
121 Err_cmdClose_Click:
122     MsgBox Err.Description
123     Resume Exit_cmdClose_Click
124
125 End Sub

```

Code

```

1 Attribute VB_Name = "Form_frmFindNarcs"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdClear_Click()
9
10 Dim strSql As String
11 Dim strSqlFull As String
12 Me!txtFind = ""
13 Me!cboType = ""
14
15 strSql = "Where False"
16 strSqlFull = "Select * From qFindNarc " & strSql
17 Me!subFindNarc.Form.RecordSource = strSqlFull
18
19 End Sub
20
21 Private Sub cmdSubmit_Click()
22
23 Dim strType As String
24 Dim strNarc As String
25 Dim strBoth As String
26 Dim strSql As String
27 Dim strMsg As String
28 Dim strTitle As String
29
30
31
32 Select Case grpSearch
33
34 Case 1
35     strType = "Where [NarcType]= " & "Forms!frmFindNarcs![cboType]"
36     strSql = "Select * From qFindNarc " & strType
37     Me!subFindNarc.Form.RecordSource = strSql
38
39 Case 2
40     strNarc = "Where [Brand]= " & "Forms!frmFindNarcs![txtFind]"
41     strSql = "Select * From qFindNarc " & strNarc
42     Me!subFindNarc.Form.RecordSource = strSql
43
44
45 Case 3
46     strType = "[NarcType]= " & "Forms!frmFindNarcs![cboType]"
47     strNarc = "[Brand]= " & "Forms!frmFindNarcs![txtFind]"
48     strBoth = "Where " & strType & " AND " & strNarc
49     strSql = "Select * From qFindNarc " & strBoth
50     Me!subFindNarc.Form.RecordSource = strSql
51
52
53 End Select
54
55 End Sub
56
57 Private Sub Detail_Click()
58
59 End Sub
60 Private Sub cmdClose_Click()
61 On Error GoTo Err_cmdClose_Click
62
63
64     DoCmd.Close

```

```
65
66 Exit_cmdClose_Click:
67     Exit Sub
68
69 Err_cmdClose_Click:
70     MsgBox Err.Description
71     Resume Exit_cmdClose_Click
72
73 End Sub
74
75 Private Sub Form_Open(Cancel As Integer)
76
77
78 Select Case grpSearch
79
80 Case 1
81     cboType.BackColor = 16777215
82     txtFind.BackColor = 12632256
83     Me.cboType.SetFocus
84 Case 2
85     txtFind.BackColor = 16777215
86     cboType.BackColor = 12632256
87     Me.txtFind.SetFocus
88 Case 3
89     cboType.BackColor = 16777215
90     txtFind.BackColor = 16777215
91     Me.cboType.SetFocus
92 End Select
93
94 End Sub
95
96 Private Sub grpSearch_AfterUpdate()
97 Select Case grpSearch
98
99 Case 1
100     cboType.BackColor = 16777215
101     txtFind.BackColor = 12632256
102     Me.cboType.SetFocus
103     Me!txtFind = ""
104
105 Case 2
106     txtFind.BackColor = 16777215
107     cboType.BackColor = 12632256
108     Me.txtFind.SetFocus
109     Me!cboType = ""
110
111 Case 3
112     cboType.BackColor = 16777215
113     txtFind.BackColor = 16777215
114     Me.cboType.SetFocus
115 End Select
116
117 End Sub
118
119 Private Sub grpSearch_BeforeUpdate(Cancel As Integer)
120
121 End Sub
```

Code

```
1 Attribute VB_Name = "Form_frmPrisDetails"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdPreview_Click()
9 On Error GoTo Err_cmdPreview_Click
10
11     Dim stDocName As String
12
13     stDocName = "rptPrisonerDetails"
14     DoCmd.OpenReport stDocName, acPreview
15
16     DoCmd.Close acForm, "frmPrisDetails", acSaveNo
17
18 Exit_cmdPreview_Click:
19     Exit Sub
20
21 Err_cmdPreview_Click:
22     MsgBox Err.Description
23     Resume Exit_cmdPreview_Click
24
25 End Sub
26
27 Private Sub cmdSetDate_Click()
28 'set date equal to date on calender
29 txtDate = dwndDate.Value
30
31 End Sub
32
33 Private Sub dwndDate_Updated(Code As Integer)
34
35 End Sub
36
37 Private Sub Form_Current()
38
39 End Sub
40
41 Private Sub Form_Load()
42 'set controls equal to today
43 Me!dwndDate = Date
44 Me!txtDate = Date
45
46 End Sub
```

Form: frmPrisDetails

```
47 Private Sub cmdPrint_Click()
48 On Error GoTo Err_cmdPrint_Click
49
50     Dim stDocName As String
51
52     stDocName = "rptPrisonerDetails"
53     DoCmd.OpenReport stDocName, acNormal
54
55 Exit_cmdPrint_Click:
56     Exit Sub
57
58 Err_cmdPrint_Click:
59     MsgBox Err.Description
60     Resume Exit_cmdPrint_Click
61
62 End Sub
63 Private Sub cmdCancel_Click()
64 On Error GoTo Err_cmdCancel_Click
65
66
67     DoCmd.Close
68
69 Exit_cmdCancel_Click:
70     Exit Sub
71
72 Err_cmdCancel_Click:
73     MsgBox Err.Description
74     Resume Exit_cmdCancel_Click
75
76 End Sub
```

Code

```
1 Attribute VB_Name = "Form_frmPTS"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdClose_Click()
9 On Error GoTo Err_cmdClose_Click
10
11
12     DoCmd.Close
13
14 Exit_cmdClose_Click:
15     Exit Sub
16
17 Err_cmdClose_Click:
18     MsgBox Err.Description
19     Resume Exit_cmdClose_Click
20
21 End Sub
22
23 Private Sub cmdFindUSMS_Click()
24 Dim rst As Recordset
25
26     Dim strCriteria As String
27     Dim strUSMS As String
28
29     'Make sure there is an entry in the text box before running
30     If IsNull(txtFindUSMS) Then
31         MsgBox "Enter a USMS numer first and then submit query"
32         Exit Sub
33     End If
34
35
36
37     Set rst = Me.RecordsetClone
38     strUSMS = Me!txtFindUSMS
39     strCriteria = "[pr_usms_no]='" & strUSMS & "'"
40     rst.FindFirst strCriteria
41     If rst.NoMatch Then
42         MsgBox "The number you entered was not found."
43     Else
44         Me.Bookmark = Me.RecordsetClone.Bookmark
45     End If
46     Me!txtFindUSMS.Value = ""
47
48 End Sub
49
50 Private Sub cmdNext_Click()
51 On Error GoTo Err_cmdNext_Click
52
53
54     DoCmd.GoToRecord , , acNext
55
56 Exit_cmdNext_Click:
57     Exit Sub
58
59 Err_cmdNext_Click:
60     MsgBox Err.Description
61     Resume Exit_cmdNext_Click
62
63 End Sub
64 Private Sub cmdPrevious_Click()
```



```
65 On Error GoTo Err_cmdPrevious_Click
66
67
68     DoCmd.GoToRecord , , acPrevious
69
70 Exit_cmdPrevious_Click:
71     Exit Sub
72
73 Err_cmdPrevious_Click:
74     MsgBox Err.Description
75     Resume Exit_cmdPrevious_Click
76
77 End Sub
78 Sub Combo33_AfterUpdate()
79
80     ' Find the record that matches the control.
81     If IsNull(Me![Combo33]) Then
82         Exit Sub
83     End If
84
85     ' Find the record that matches the control.
86     Me.RecordsetClone.FindFirst "[pr_key] = " & Me![Combo33]
87     Me.Bookmark = Me.RecordsetClone.Bookmark
88 End Sub
89
90 Private Sub Form_Current()
91
92     'synchronize combo box with record
93     Combo33 = pr_key
94 End Sub
95 Private Sub cmdPrintReport_Click()
96 On Error GoTo Err_cmdPrintReport_Click
97
98     Dim stDocName As String
99
100     stDocName = "rptPrisonerDetailsPTS"
101     DoCmd.OpenReport stDocName, acNormal
102
103 Exit_cmdPrintReport_Click:
104     Exit Sub
105
106 Err_cmdPrintReport_Click:
107     MsgBox Err.Description
108     Resume Exit_cmdPrintReport_Click
109
110 End Sub
```

Code

```
1 Attribute VB_Name = "Form_frmPTSArrests"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7 Const ConErrRptCanceled = 2501
8
9
10 Private Sub cmdCancel_Click()
11
12 ' Close form
13 On Error GoTo Err_cmdCancel_Click
14
15     DoCmd.Close
16
17 Exit_cmdCancel_Click:
18     Exit Sub
19
20 Err_cmdCancel_Click:
21     MsgBox Err.Description
22     Resume Exit_cmdCancel_Click
23
24 End Sub
25
26
27
28 Private Sub cmdPreview_Click()
29
30 ' Preview report.
31 Dim strDocName As String
32
33     On Error GoTo Err_cmdPreview_Click
34
35     ' Check to see that ending date is later than beginning date.
36     If IsDate(BeginDate) And IsDate(EndDate) Then
37         If EndDate < BeginDate Then
38             MsgBox "The end date must be later than the begin date."
39             cmdSetDate.Caption = "Set End Date"
40             SelectDate.SetFocus
41             Exit Sub
42         End If
43     Else
44         MsgBox "Please use a valid date for the beginning date and the ending date
45 values."
46 Exit Sub
47     End If
48
49     strDocName = "PTS Arrests"
50     DoCmd.OpenReport strDocName, acViewPreview
51
52 Exit_cmdPreview_Click:
53     Exit Sub
54
55 Err_cmdPreview_Click:
56     If Err = ConErrRptCanceled Then
57         Resume Exit_cmdPreview_Click
58     Else
59         MsgBox Err.Description
60         Resume Exit_cmdPreview_Click
61     End If
62
```

```
64
65 Private Sub cmdPrint_Click()
66     ' Print report.
67     Dim strDocName As String
68
69     On Error GoTo Err_cmdPrint_Click
70
71     ' Check to see that ending date is later than beginning date.
72     If EndDate < BeginDate Then
73         MsgBox "The end date must be later than the begin date."
74         cmdSetDate.Caption = "Set End Date"
75         SelectDate.SetFocus
76         Exit Sub
77     End If
78
79     strDocName = "PTS Arrests"
80     DoCmd.OpenReport strDocName, acViewNormal
81
82 Exit_cmdPrint_Click:
83     Exit Sub
84
85 Err_cmdPrint_Click:
86     If Err = ConErrRptCanceled Then
87         Resume Exit_cmdPrint_Click
88     Else
89         MsgBox Err.Description
90         Resume Exit_cmdPrint_Click
91     End If
92
93 End Sub
94
95
96 Private Sub cmdSetDate_Click()
97
98     On Error GoTo cmdSetDate_Error
99
100    If cmdSetDate.Caption = "Set Begin Date" Then
101        BeginDate = SelectDate.Value
102        cmdSetDate.Caption = "Set End Date"
103    Else
104        EndDate = SelectDate.Value
105        cmdSetDate.Caption = "Set Begin Date"
106    End If
107
108    Exit Sub
109
110 cmdSetDate_Error:
111    MsgBox Err.Description
112    Exit Sub
113
114 End Sub
115
116 Function IsLoaded(strfrmName As String) As Integer
117
118     ' Determines if a form is loaded.
119
120     Const conFormDesign = 0
121     Dim intX As Integer
122
123     IsLoaded = False
124     For intX = 0 To Forms.Count - 1
125         If Forms(intX).FormName = strfrmName Then
126             If Forms(intX).CurrentView <> conFormDesign Then
127                 IsLoaded = True
128                 Exit Function ' Quit function once form has been found.
129             End If
130         End If
```

```
131     Next
132
Form: frmPTSArrests
```

```
133 End Function
134
135
136 Private Sub Detail_Click()
137
138 End Sub
139
140 Private Sub Form_Load()
141
142 'Enter parameters in date boxes
143 Me!SelectDate = Now()
144 Me!BeginDate = Now() - 7
145 Me!EndDate = Now()
146
147 End Sub
```

Code

```
1 Attribute VB_Name = "Form_frmREPORTS"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdOpenRptPtsDwn_Click()
9 On Error GoTo Err_cmdOpenRptPtsDwn_Click
10
11     Dim stDocName As String
12     Dim stLinkCriteria As String
13
14     stDocName = "PTSDownloads"
15     DoCmd.OpenForm stDocName, , , stLinkCriteria
16
17 Exit_cmdOpenRptPtsDwn_Click:
18     Exit Sub
19
20 Err_cmdOpenRptPtsDwn_Click:
21     MsgBox Err.Description
22     Resume Exit_cmdOpenRptPtsDwn_Click
23
24 End Sub
```

Code

```
1 Attribute VB_Name = "Form_frmTARGET_PHOTOsub"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdClose_Click()
9 On Error GoTo Err_cmdClose_Click
10
11
12     DoCmd.Close
13
14 Exit_cmdClose_Click:
15     Exit Sub
16
17 Err_cmdClose_Click:
18     MsgBox Err.Description
19     Resume Exit_cmdClose_Click
20
21 End Sub
22 Private Sub cmdPrint_Click()
23 On Error GoTo Err_cmdPrint_Click
24
25
26     DoCmd.PrintOut
27
28 Exit_cmdPrint_Click:
29     Exit Sub
30
31 Err_cmdPrint_Click:
32     MsgBox Err.Description
33     Resume Exit_cmdPrint_Click
34
35 End Sub
```

Code

```

1 Attribute VB_Name = "Form_frmTARGETS"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdPhoto_Click()
9 On Error GoTo Err_cmdPhoto_Click
10
11     Dim stDocName As String
12     Dim stLinkCriteria As String
13
14     stDocName = "frmTARGET_PHOTOsub"
15
16     stLinkCriteria = "[TargetKey]=" & Me![TargetKey]
17
18     DoCmd.OpenForm stDocName, , , stLinkCriteria
19
20 Exit_cmdPhoto_Click:
21     Exit Sub
22
23 Err_cmdPhoto_Click:
24     MsgBox Err.Description
25     Resume Exit_cmdPhoto_Click
26
27 End Sub
28
29 Private Sub Form_AfterUpdate()
30
31 'enable or disable photo buttons based on record status
32     If IsNull(Me![TargetKey]) And IsNull(Me![txtPhoto]) Then
33         Me!cmdPhoto.Enabled = False
34     Else
35         Me!cmdPhoto.Enabled = True
36     End If
37
38     If Not (IsNull(Me![TargetKey])) And Not (IsNull(Me![txtPhoto])) Then
39         Me!cmdInsertPhoto.Enabled = True
40     Else
41         Me!cmdInsertPhoto.Enabled = False
42     End If
43 'enable or disable photo buttons based on record status
44     If IsNull(Me![TargetKey]) And IsNull(Me![txtPhoto]) Then
45         Me!cmdPhoto.Enabled = False
46     Else
47         Me!cmdPhoto.Enabled = True
48     End If
49
50     If Not (IsNull(Me![TargetKey])) And Not (IsNull(Me![txtPhoto])) Then
51         Me!cmdInsertPhoto.Enabled = True
52     Else
53         Me!cmdInsertPhoto.Enabled = False
54     End If
55
56 End Sub
57
58 Private Sub Form_Current()
59
60 'enable or disable photo buttons based on record status
61     If IsNull(Me![TargetKey]) Or IsNull(Me![txtPhoto]) Then
62         Me!cmdPhoto.Enabled = False
63     Else
64         Me!cmdPhoto.Enabled = True

```

```
65     End If
66
67     If Not (IsNull(Me![TargetKey])) And (IsNull(Me![txtPhoto])) Then
68         Me!cmdInsertPhoto.Enabled = True
69     Else
70         Me!cmdInsertPhoto.Enabled = False
71     End If
72
73 End Sub
74
75 Private Sub Form_Load()
76     If IsNull(Me![TargetKey]) Then
77         Me!cmdPhoto.Enabled = False
78     Else
79         Me!cmdPhoto.Enabled = True
80     End If
81
82 End Sub
83 Private Sub cmdInsertPhoto_Click()
84 On Error GoTo Err_cmdInsertPhoto_Click
85
86     Dim stDocName As String
87     Dim stLinkCriteria As String
88
89     stDocName = "frmTARGET_PHOTOSub"
90
91     stLinkCriteria = "[TargetKey]=" & Me![TargetKey]
92     DoCmd.OpenForm stDocName, , , stLinkCriteria
93
94 Exit_cmdInsertPhoto_Click:
95     Exit Sub
96
97 Err_cmdInsertPhoto_Click:
98     MsgBox Err.Description
99     Resume Exit_cmdInsertPhoto_Click
100
101 End Sub
```


Form: HOMICIDESubform

Code

```
1 Attribute VB_Name = "Form_HOMICIDESubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Form_Open(Cancel As Integer)
9
10 'On opening set focus to Homicide date
11 Me!HomDate.SetFocus
12
13 End Sub
```

Code

```
1 Attribute VB_Name = "Form_LOCALEsubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Form_Load()
9
10 Me!LocName.SetFocus
11 Me.Repaint
12
13 End Sub
```

Code

```
1 Attribute VB_Name = "Form_NARCOTICSsubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Form_Current()
9
10 End Sub
11
12 Private Sub NarcType_AfterUpdate()
13
14 If IsNull(Me!NarcType) Then
15     Forms!COOPERATORS!subNARCOTICS!subNARC_BRAND!Brand.Enabled = False
16 Else
17     Forms!COOPERATORS!subNARCOTICS!subNARC_BRAND!Brand.Enabled = True
18 End If
19
20 End Sub
21
22 Private Sub NarcType_BeforeUpdate(Cancel As Integer)
23
24 End Sub
```

Code

```

1 Attribute VB_Name = "Form_PTSDownloads"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7 Const ConErrRptCanceled = 2501
8
9
10 Private Sub cmdCancel_Click()
11
12 ' Close form
13 On Error GoTo Err_cmdCancel_Click
14
15     DoCmd.Close
16
17 Exit_cmdCancel_Click:
18     Exit Sub
19
20 Err_cmdCancel_Click:
21     MsgBox Err.Description
22     Resume Exit_cmdCancel_Click
23
24 End Sub
25
26
27
28 Private Sub cmdPreview_Click()
29
30 ' Preview report.
31 Dim strDocName As String
32
33     On Error GoTo Err_cmdPreview_Click
34
35     ' Check to see that ending date is later than beginning date.
36     If IsDate(BeginDate) And IsDate(EndDate) Then
37         If EndDate < BeginDate Then
38             MsgBox "The end date must be later than the begin date."
39             cmdSetDate.Caption = "Set End Date"
40             SelectDate.SetFocus
41             Exit Sub
42         End If
43     Else
44         MsgBox "Please use a valid date for the beginning date and the ending date
45         values."
46     Exit Sub
47     End If
48
49     strDocName = "PTS Downloads"
50     DoCmd.OpenReport strDocName, acViewPreview
51
52 Exit_cmdPreview_Click:
53     Exit Sub
54
55 Err_cmdPreview_Click:
56     If Err = ConErrRptCanceled Then
57         Resume Exit_cmdPreview_Click
58     Else
59         MsgBox Err.Description
60         Resume Exit_cmdPreview_Click
61     End If
62

```

```
64
65 Private Sub cmdPrint_Click()
66     ' Print report.
67     Dim strDocName As String
68
69     On Error GoTo Err_cmdPrint_Click
70
71     ' Check to see that ending date is later than beginning date.
72     If EndDate < BeginDate Then
73         MsgBox "The end date must be later than the begin date."
74         cmdSetDate.Caption = "Set End Date"
75         SelectDate.SetFocus
76         Exit Sub
77     End If
78
79     strDocName = "PTS Downloads"
80     DoCmd.OpenReport strDocName, acViewNormal
81
82 Exit_cmdPrint_Click:
83     Exit Sub
84
85 Err_cmdPrint_Click:
86     If Err = ConErrRptCanceled Then
87         Resume Exit_cmdPrint_Click
88     Else
89         MsgBox Err.Description
90         Resume Exit_cmdPrint_Click
91     End If
92
93 End Sub
94
95
96 Private Sub cmdSetDate_Click()
97
98     On Error GoTo cmdSetDate_Error
99
100    If cmdSetDate.Caption = "Set Begin Date" Then
101        BeginDate = SelectDate.Value
102        cmdSetDate.Caption = "Set End Date"
103    Else
104        EndDate = SelectDate.Value
105        cmdSetDate.Caption = "Set Begin Date"
106    End If
107
108    Exit Sub
109
110 cmdSetDate_Error:
111    MsgBox Err.Description
112    Exit Sub
113
114 End Sub
115
116 Function IsLoaded(strfrmName As String) As Integer
117
118     ' Determines if a form is loaded.
119
120     Const conFormDesign = 0
121     Dim intX As Integer
122
123     IsLoaded = False
124     For intX = 0 To Forms.Count - 1
125         If Forms(intX).FormName = strfrmName Then
126             If Forms(intX).CurrentView <> conFormDesign Then
127                 IsLoaded = True
128                 Exit Function ' Quit function once form has been found.
129             End If
130         End If
131     End If
```

131 Next
132
Form: PTSDownloads

```
133 End Function
134
135
136 Private Sub Detail_Click()
137
138 End Sub
139
140 Private Sub Form_Load()
141
142 'Enter parameters in date boxes
143 Me!SelectDate = Now()
144 Me!BeginDate = Now() - 7
145 Me!EndDate = Now()
146
147 End Sub
```

Code

```

1 Attribute VB_Name = "Form_StartUp"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdOpenAlias_Click()
9 On Error GoTo Err_cmdOpenAlias_Click
10
11     Dim stDocName As String
12     Dim stLinkCriteria As String
13
14     stDocName = "frmFindAlias"
15     DoCmd.OpenForm stDocName, , , stLinkCriteria
16     DoCmd.Close acForm, "StartUp"
17
18 Exit_cmdOpenAlias_Click:
19     Exit Sub
20
21 Err_cmdOpenAlias_Click:
22     MsgBox Err.Description
23     Resume Exit_cmdOpenAlias_Click
24
25 End Sub
26
27 Private Sub cmdOpenAssoc_Click()
28 On Error GoTo Err_cmdOpenAssoc_Click
29
30     Dim stDocName As String
31     Dim stLinkCriteria As String
32
33     stDocName = "frmFindAssoc"
34     DoCmd.OpenForm stDocName, , , stLinkCriteria
35     DoCmd.Close acForm, "StartUp"
36
37 Exit_cmdOpenAssoc_Click:
38     Exit Sub
39
40 Err_cmdOpenAssoc_Click:
41     MsgBox Err.Description
42     Resume Exit_cmdOpenAssoc_Click
43
44 End Sub
45
46 Private Sub cmdOpenCoopFrm_Click()
47 On Error GoTo Err_cmdOpenCoopFrm_Click
48
49     Dim stDocName As String
50     Dim stLinkCriteria As String
51
52     stDocName = "COOPERATORS"
53     DoCmd.OpenForm stDocName, , , stLinkCriteria
54     DoCmd.Close acForm, "StartUp"
55
56 Exit_cmdOpenCoopFrm_Click:
57     Exit Sub
58
59 Err_cmdOpenCoopFrm_Click:
60     MsgBox Err.Description
61     Resume Exit_cmdOpenCoopFrm_Click
62
63
64 End Sub

```

```
65
66 Private Sub cmdOpenCounsel_Click()
67 On Error GoTo Err_cmdOpenCounsel_Click
68
69     Dim stDocName As String
70     Dim stLinkCriteria As String
71
72     stDocName = "frmFindCounsel"
73     DoCmd.OpenForm stDocName, , , stLinkCriteria
74     DoCmd.Close acForm, "StartUp"
75
76 Exit_cmdOpenCounsel_Click:
77     Exit Sub
78
79 Err_cmdOpenCounsel_Click:
80     MsgBox Err.Description
81     Resume Exit_cmdOpenCounsel_Click
82
83 End Sub
84
85 Private Sub cmdOpenDefFrm_Click()
86 On Error GoTo Err_cmdOpenDefFrm_Click
87
88     Dim stDocName As String
89     Dim stLinkCriteria As String
90
91     stDocName = "DEFENDANT"
92     DoCmd.OpenForm stDocName, , , stLinkCriteria
93     DoCmd.Close acForm, "StartUp"
94
95 Exit_cmdOpenDefFrm_Click:
96     Exit Sub
97
98 Err_cmdOpenDefFrm_Click:
99     MsgBox Err.Description
100    Resume Exit_cmdOpenDefFrm_Click
101
102
103 End Sub
104
105 Private Sub cmdOpenFindEmploy_Click()
106
107 End Sub
108
109 Private Sub cmdOpenMaint_Click()
110
111 Me.GoToPage 4
112
113 End Sub
114
115 Private Sub cmdOpenNarc_Click()
116 On Error GoTo Err_cmdOpenNarc_Click
117
118     Dim stDocName As String
119     Dim stLinkCriteria As String
120
121     stDocName = "frmFindNarcs"
122     DoCmd.OpenForm stDocName, , , stLinkCriteria
123     DoCmd.Close acForm, "StartUp"
124
125 Exit_cmdOpenNarc_Click:
126     Exit Sub
127
128 Err_cmdOpenNarc_Click:
129     MsgBox Err.Description
130     Resume Exit_cmdOpenNarc_Click
131
132 End Sub
```



```

134 Private Sub cmdOpenPg3_Click()
135
136 Me.GoToPage 3
137
138 End Sub
139
140 Private Sub cmdOpenRptfrm_Click()
141 Me.GoToPage 2
142 End Sub
143
144 Private Sub cmdOpenTQfrm_Click()
145
146 If Me!lblMain.Visible = True Then
147     Me.lblMain.Visible = False
148     Me.lblRpt.Visible = False
149
150     Me.lblSQ.Visible = False
151     Me.lblTQ.Visible = False
152     Me.lblRecords.Visible = False
153     Me.cmdOpenMaint.Visible = False
154     Me.cmdOpenPg3.Visible = False
155     Me.cmdOpenSQfrm.Visible = False
156     Me.cmdOpenRptfrm.Visible = False
157     Me.lblAlias.Visible = True
158     Me.lblNarc.Visible = True
159     Me.lblAssoc.Visible = True
160     Me.lblCounsel.Visible = True
161     Me.lblEmploy.Visible = True
162     Me.lblReturnMain.Visible = True
163     Me.cmdOpenAlias.Visible = True
164     Me.cmdOpenNarc.Visible = True
165     Me.cmdOpenAssoc.Visible = True
166     Me.cmdOpenCounsel.Visible = True
167     Me.cmdOpenEmploy.Visible = True
168     Me.cmdOpenAlias.SetFocus
169     Me.cmdOpenTQfrm.Picture = "c:\Program
Files\DevStudio\VB\Graphics\Icons\Arrows\Arw091t.ico"
170
171 ElseIf Me!lblMain.Visible = False Then
172     Me.lblMain.Visible = True
173     Me.lblRpt.Visible = True
174     Me.lblSQ.Visible = True
175     Me.lblTQ.Visible = True
176     Me.lblRecords.Visible = True
177     Me.cmdOpenPg3.Visible = True
178     Me.cmdOpenSQfrm.Visible = True
179     Me.cmdOpenRptfrm.Visible = True
180     Me.lblMaint.Visible = True
181     Me.cmdOpenMaint.Visible = True
182     Me.lblAlias.Visible = False
183     Me.lblNarc.Visible = False
184     Me.lblAssoc.Visible = False
185     Me.lblCounsel.Visible = False
186     Me.lblEmploy.Visible = False
187     Me.lblReturnMain.Visible = False
188     Me.cmdOpenAlias.Visible = False
189     Me.cmdOpenNarc.Visible = False
190     Me.cmdOpenAssoc.Visible = False
191     Me.cmdOpenCounsel.Visible = False
192     Me.cmdOpenEmploy.Visible = False
193     Me.cmdOpenTQfrm.Picture = "C:\Program
Files\DevStudio\VB\Graphics\Icons\Misc\LIGHTON.ICO"
194 End If
195 End Sub
196
197 Private Sub Command30_Click()

```

```
198 Me.GoToPage 1
199 End Sub
Form: StartUp
```

```
200
201 Private Sub Command34_Click()
202
203 End Sub
204
205 Private Sub Command40_Click()
206
207 Me.GoToPage 1
208
209 End Sub
210
211 Private Sub Command82_Click()
212
213 'run transfer.exe - vb app that inputs email messages into database
214 Dim RetVal
215 RetVal = Shell("D:\NETWORK\RACKETS\MISC\TRANSLATE.EXE", 1)
216
217 DoCmd.Close acForm, "StartUp"
218
219 End Sub
220
221 Private Sub Form_Load()
222 Me.GoToPage 1
223
224 Me.lblAlias.Visible = False
225 Me.lblNarc.Visible = False
226 Me.lblAssoc.Visible = False
227 Me.lblCounsel.Visible = False
228 Me.lblEmploy.Visible = False
229 Me.lblReturnMain.Visible = False
230 Me.cmdOpenAlias.Visible = False
231 Me.cmdOpenNarc.Visible = False
232 Me.cmdOpenAssoc.Visible = False
233 Me.cmdOpenCounsel.Visible = False
234 Me.cmdOpenEmploy.Visible = False
235 Me.lblMain.Visible = True
236 Me.lblRpt.Visible = True
237 Me.lblSQ.Visible = True
238 Me.lblTQ.Visible = True
239 Me.lblRecords.Visible = True
240 Me.lblMaint.Visible = True
241 Me.cmdOpenMaint.Visible = True
242 Me.cmdOpenPg3.Visible = True
243 Me.cmdOpenTQfrm.Visible = True
244 Me.cmdOpenSQfrm.Visible = True
245 Me.cmdOpenRptfrm.Visible = True
246
247 End Sub
248
249 Private Sub Form_Open(Cancel As Integer)
250
251 Me.GoToPage 1
252
253 Me.lblAlias.Visible = False
254 Me.lblNarc.Visible = False
255 Me.lblAssoc.Visible = False
256 Me.lblCounsel.Visible = False
257 Me.lblEmploy.Visible = False
258 Me.lblReturnMain.Visible = False
259 Me.cmdOpenAlias.Visible = False
260 Me.cmdOpenNarc.Visible = False
261 Me.cmdOpenAssoc.Visible = False
262 Me.cmdOpenCounsel.Visible = False
263 Me.cmdOpenEmploy.Visible = False
264 Me.lblMain.Visible = True
265 Me.lblRpt.Visible = True
```

```
266 Me.lblSQ.Visible = True
267 Me.lblTQ.Visible = True
268 Me.lblMaint.Visible = True
```

Form: StartUp

```
269 Me.cmdOpenMaint.Visible = True
270 Me.lblRecords.Visible = True
271 Me.cmdOpenPg3.Visible = True
272 Me.cmdOpenTQfrm.Visible = True
273 Me.cmdOpenSQfrm.Visible = True
274 Me.cmdOpenRptfrm.Visible = True
275
276 End Sub
277 Private Sub cmdExit_Click()
278 On Error GoTo Err_cmdExit_Click
279
280
281     DoCmd.Quit
282
283 Exit_cmdExit_Click:
284     Exit Sub
285
286 Err_cmdExit_Click:
287     MsgBox Err.Description
288     Resume Exit_cmdExit_Click
289
290 End Sub
291 Private Sub cmdOpenCaseFrm_Click()
292 On Error GoTo Err_cmdOpenCaseFrm_Click
293
294     Dim stDocName As String
295     Dim stLinkCriteria As String
296
297     stDocName = "CASE"
298     DoCmd.OpenForm stDocName, , , stLinkCriteria
299     DoCmd.Close acForm, "StartUp"
300
301 Exit_cmdOpenCaseFrm_Click:
302     Exit Sub
303
304 Err_cmdOpenCaseFrm_Click:
305     MsgBox Err.Description
306     Resume Exit_cmdOpenCaseFrm_Click
307
308 End Sub
309 Private Sub cmdOpenNewCase_Click()
310 On Error GoTo Err_cmdOpenNewCase_Click
311
312     Dim stDocName As String
313     Dim stLinkCriteria As String
314
315     stDocName = "CASE"
316     DoCmd.OpenForm stDocName, , , stLinkCriteria
317     DoCmd.Close acForm, "StartUp"
318
319     Forms!CASE.DataEntry = True
320     Forms!CASE.cmdArchive.Enabled = False
321     Forms!CASE.cmdArchive.Visible = False
322     Forms!CASE.Combol5.Enabled = False
323     Forms!CASE.Combol5.Visible = False
324     Forms!CASE.lblUSV.Visible = False
325
326 Exit_cmdOpenNewCase_Click:
327     Exit Sub
328
329 Err_cmdOpenNewCase_Click:
330     MsgBox Err.Description
331     Resume Exit_cmdOpenNewCase_Click
332
333 End Sub
```

```

334 Private Sub cmdOpenPTS_Click()
335 On Error GoTo Err_cmdOpenPTS_Click
336
337     Dim stDocName As String
Form: StartUp

338     Dim stLinkCriteria As String
339
340     stDocName = "frmPTS"
341     DoCmd.Close acForm, "StartUp"
342     DoCmd.OpenForm stDocName, , , stLinkCriteria
343
344 Exit_cmdOpenPTS_Click:
345     Exit Sub
346
347 Err_cmdOpenPTS_Click:
348     MsgBox Err.Description
349     Resume Exit_cmdOpenPTS_Click
350
351 End Sub
352 Private Sub cmdOpenEmploy_Click()
353 On Error GoTo Err_cmdOpenEmploy_Click
354
355     Dim stDocName As String
356     Dim stLinkCriteria As String
357
358     stDocName = "frmFindEmployer"
359     DoCmd.OpenForm stDocName, , , stLinkCriteria
360     DoCmd.Close acForm, "StartUp"
361
362 Exit_cmdOpenEmploy_Click:
363     Exit Sub
364
365 Err_cmdOpenEmploy_Click:
366     MsgBox Err.Description
367     Resume Exit_cmdOpenEmploy_Click
368
369 End Sub
370 Private Sub cmdOpenclnArrest_Click()
371 On Error GoTo Err_cmdOpenclnArrest_Click
372
373     Dim stDocName As String
374     Dim stLinkCriteria As String
375
376     stDocName = "cln_ARREST"
377     DoCmd.OpenForm stDocName, , , stLinkCriteria
378
379 Exit_cmdOpenclnArrest_Click:
380     Exit Sub
381
382 Err_cmdOpenclnArrest_Click:
383     MsgBox Err.Description
384     Resume Exit_cmdOpenclnArrest_Click
385
386 End Sub
387 Private Sub cmdOpenPTSarr_Click()
388 End Sub

```

Code

```
1 Attribute VB_Name = "Form_STATUTESsubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Count_BeforeUpdate(Cancel As Integer)
9
10 End Sub
11
12 Private Sub Form_AfterUpdate()
13 End Sub
14
15 Private Sub Form_BeforeUpdate(Cancel As Integer)
16
17
18 End Sub
19
20 Private Sub Form_Current()
21
22
23 End Sub
24
25 Private Sub Form_Error(DataErr As Integer, Response As Integer)
26
27     Const conErrFieldRequired = 3314
28
29     If DataErr = conErrFieldRequired Then
30         DisplayMessage "You must enter a Count and Statute number to save the record."
31
32         Response = acDataErrContinue
33     Else
34         Response = acDataErrDisplay
35     End If
36 End Sub
37
38 Private Sub Form_Load()
39
40 End Sub
41
42 Private Sub Location_AfterUpdate()
43
44
45 End Sub
46
47 Private Sub Location_Click()
48
49
50 End Sub
51
52 Private Sub StatuteDesc_AfterUpdate()
53
54
55
56 End Sub
```

Form: WARRANTsubform

Code

```
1 Attribute VB_Name = "Form_WARRANTsubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Form_Open(Cancel As Integer)
9
10
11 End Sub
```

Form: WIRETAPsubform

Code

```
1 Attribute VB_Name = "Form_WIRETAPsubform"  
2 Attribute VB_Creatable = True  
3 Attribute VB_PredeclaredId = True  
4 Attribute VB_Exposed = False  
5 Option Compare Database  
6 Option Explicit  
7  
8 Private Sub Form_Open(Cancel As Integer)  
9  
10  
11  
12 End Sub
```

Code

```

1 Attribute VB_Name = "Form_CASE"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8
9 Private Sub AUSA_NotInList(NewData As String, Response As Integer)
10
11 Dim strMessage As String
12 Dim dbsRkts As Database
13 Dim rstTypes As Recordset
14
15 strMessage = "Add '" & NewData & "to the list of AUSAs?"
16
17 If Confirm(strMessage) Then
18
19     'open table and add record
20     Set dbsRkts = CurrentDb()
21     Set rstTypes = dbsRkts.OpenRecordset("ausa")
22     rstTypes.AddNew
23     rstTypes!AUSA = NewData
24     rstTypes.Update
25     Response = acDataErrAdded 'requery list
26 Else
27     Response = acDataErrDisplay ' display error
28 End If
29
30 End Sub
31
32 Sub Combo15_AfterUpdate()
33     ' Find the record that matches the control.
34     If IsNull(Me![Combo15]) Then
35         Exit Sub
36     End If
37
38     Me.RecordsetClone.FindFirst "[CaseKey] = " & Me![Combo15]
39     Me.Bookmark = Me.RecordsetClone.Bookmark
40 End Sub
41
42
43
44 Private Sub Command50_Click()
45
46 End Sub
47
48 Private Sub Combo15_BeforeUpdate(Cancel As Integer)
49
50 End Sub
51
52 Private Sub Form_Current()
53     Combo15 = CaseKey
54 End Sub
55 Private Sub cmdClose_Click()
56 On Error GoTo Err_cmdClose_Click
57
58
59     DoCmd.Close
60
61 Exit_cmdClose_Click:
62     Exit Sub
63
64 Err_cmdClose_Click:
65     MsgBox Err.Description

```



```
66     Resume Exit_cmdClose_Click
67
68 End Sub
69 Private Sub cmdPrevious_Click()
70 On Error GoTo Err_cmdPrevious_Click
71
72
73     DoCmd.GoToRecord , , acPrevious
74
75 Exit_cmdPrevious_Click:
76     Exit Sub
77
78 Err_cmdPrevious_Click:
79     MsgBox Err.Description
80     Resume Exit_cmdPrevious_Click
81
82 End Sub
83 Private Sub cmdNext_Click()
84 On Error GoTo Err_cmdNext_Click
85
86
87     DoCmd.GoToRecord , , acNext
88
89 Exit_cmdNext_Click:
90     Exit Sub
91
92 Err_cmdNext_Click:
93     MsgBox Err.Description
94     Resume Exit_cmdNext_Click
95
96 End Sub
97 Private Sub cmdAddRecord_Click()
```

Form: CASE

```
98 On Error GoTo Err_cmdAddRecord_Click
99
100
101     DoCmd.GoToRecord , , acNewRec
102     Me!USAONum.SetFocus
103
104 Exit_cmdAddRecord_Click:
105     Exit Sub
106
107 Err_cmdAddRecord_Click:
108     MsgBox Err.Description
109     Resume Exit_cmdAddRecord_Click
110
111 End Sub
112 Private Sub cmdArchive_Click()
113 On Error GoTo Err_cmdArchive_Click
114
115     Dim stDocName As String
116     Dim stDocName1 As String
117     Dim stDocName2 As String
118     Dim stDocName3 As String
119     Dim stDocName4 As String
120     Dim stDocName5 As String
121     Dim stDocName6 As String
122
123     stDocName = "arcCASE"
124     stDocName1 = "arcDEFENDANT"
125     stDocName2 = "arcALIAS"
126     stDocName3 = "arcSTATUTES"
127     stDocName4 = "arcEVENTS"
128     stDocName5 = "arcGROUPS"
129     stDocName6 = "delCASE"
130
131     DoCmd.OpenQuery stDocName, acNormal, acEdit
132     DoCmd.OpenQuery stDocName1, acNormal, acEdit
133     DoCmd.OpenQuery stDocName2, acNormal, acEdit
134     DoCmd.OpenQuery stDocName3, acNormal, acEdit
135     DoCmd.OpenQuery stDocName4, acNormal, acEdit
136     DoCmd.OpenQuery stDocName5, acNormal, acEdit
137     DoCmd.OpenQuery stDocName6, acNormal, acEdit
138
139 Exit_cmdArchive_Click:
140     Exit Sub
141
142 Err_cmdArchive_Click:
143     MsgBox Err.Description
144     Resume Exit_cmdArchive_Click
145
146 End Sub
147
148 Private Sub Form_Error(DataErr As Integer, Response As Integer)
```

Form: CASE

```
149
150 Const conErrFieldRequired = 3314
151
152 If DataErr = conErrFieldRequired Then
153     DisplayMessage "You must enter a Case name and USAO number to save the record."

154     Response = acDataErrContinue
155 Else
156     Response = acDataErrDisplay
157 End If
158
159 End Sub
160
161 Private Sub Form_Load()
162 End Sub
```

Code

```
1 Attribute VB_Name = "Form_COOP_AREA"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Topic_NotInList(NewData As String, Response As Integer)
9 Dim strMessage As String
10 Dim dbsRkts As Database
11 Dim rstTypes As Recordset
12
13 strMessage = "Add '" & NewData & "' to the list of Activities?"
14
15 If Confirm(strMessage) Then
16
17     'open table and add record
18     Set dbsRkts = CurrentDb()
19     Set rstTypes = dbsRkts.OpenRecordset("coop_activity")
20     rstTypes.AddNew
21     rstTypes!Activity = NewData
22     rstTypes.Update
23     Response = acDataErrAdded 'requery list
24 Else
25     Response = acDataErrDisplay ' display error
26 End If
27
28 End Sub
```

Form: COOP_PEOPLEsubArea

Code

```
1 Attribute VB_Name = "Form_COOP_PEOPLEsubArea"  
2 Attribute VB_Creatable = True  
3 Attribute VB_PredeclaredId = True  
4 Attribute VB_Exposed = False  
5 Option Compare Database  
6 Option Explicit  
7  
8  
9  
10 Private Sub Form_BeforeUpdate(Cancel As Integer)  
11 Me![CoopKey] = Me.Parent![CoopKey]  
12 End Sub
```

Code

```

1 Attribute VB_Name = "Form_COOPERATORS"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cboFindName_AfterUpdate()
9
10 Me.RecordsetClone.FindFirst "[DefKey] = " & Me![cboFindName]
11 Me.Bookmark = Me.RecordsetClone.Bookmark
12
13
14 End Sub
15
16 Private Sub cmdClose_Click()
17 On Error GoTo Err_cmdClose_Click
18
19
20     DoCmd.Close
21
22 Exit_cmdClose_Click:
23     Exit Sub
24
25 Err_cmdClose_Click:
26     MsgBox Err.Description
27     Resume Exit_cmdClose_Click
28
29 End Sub
30
31 Private Sub LName_AfterUpdate()
32
33 'Fill in Last name from defendant table
34 strFilter = "Me!DefKey = Forms!DEFENDANT!DefKey"
35 Me!LName = DLookup("[DefLName]", "DEFENDANT", strFilter)
36
37 End Sub
38
39 Private Sub LName_BeforeUpdate(Cancel As Integer)
40
41 End Sub
42 Private Sub cmdArchive_Click()
43 On Error GoTo Err_cmdArchive_Click
44
45     Dim stDocName As String
46     Dim stDocName1 As String
47
48     Dim stDocName2 As String
49     Dim stDocName3 As String
50     Dim stDocName4 As String
51     Dim stDocName5 As String
52
53     stDocName = "arcCOOPERATOR"
54     stDocName1 = "arcCOOP_AREA"
55     stDocName2 = "arcCOOP_GROUP"
56     stDocName3 = "arcHOMICIDE"
57     stDocName4 = "arcNARCOTIC"
58     stDocName5 = "delCOOPERATOR"
59
60     DoCmd.OpenQuery stDocName, acNormal, acEdit
61     DoCmd.OpenQuery stDocName1, acNormal, acEdit
62     DoCmd.OpenQuery stDocName2, acNormal, acEdit
63     DoCmd.OpenQuery stDocName3, acNormal, acEdit
64     DoCmd.OpenQuery stDocName4, acNormal, acEdit
65     DoCmd.OpenQuery stDocName5, acNormal, acEdit

```

```
65
66 Exit_cmdArchive_Click:
67     Exit Sub
68
69 Err_cmdArchive_Click:
70     MsgBox Err.Description
71     Resume Exit_cmdArchive_Click
72
73 End Sub
74
75 Private Sub Combo42_AfterUpdate()
76
77 End Sub
78
79 Private Sub Form_Current()
80
81 'Synchronize combobox and records
82 cboFindName = DefKey
83
84 End Sub
85 Private Sub cmdPrevious_Click()
86 On Error GoTo Err_cmdPrevious_Click
87
88     DoCmd.GoToRecord , , acPrevious
89
90
91 Exit_cmdPrevious_Click:
92     Exit Sub
93
94 Err_cmdPrevious_Click:
95     MsgBox Err.Description
96     Resume Exit_cmdPrevious_Click
97
98 End Sub
99 Private Sub cmdNext_Click()
100 On Error GoTo Err_cmdNext_Click
101
102
103     DoCmd.GoToRecord , , acNext
104
105 Exit_cmdNext_Click:
106     Exit Sub
107
108 Err_cmdNext_Click:
109     MsgBox Err.Description
110     Resume Exit_cmdNext_Click
111
112 End Sub
113 Private Sub cmdFirst_Click()
114 On Error GoTo Err_cmdFirst_Click
115
116
117     DoCmd.GoToRecord , , acFirst
118
119 Exit_cmdFirst_Click:
120     Exit Sub
121
122 Err_cmdFirst_Click:
123     MsgBox Err.Description
124     Resume Exit_cmdFirst_Click
125
126 End Sub
127 Private Sub cmdLast_Click()
128 On Error GoTo Err_cmdLast_Click
129
130
131     DoCmd.GoToRecord , , acLast
132
```

133 Exit_cmdLast_Click:
Form: COOPERATORS

```
134     Exit Sub
135
136 Err_cmdLast_Click:
137     MsgBox Err.Description
138     Resume Exit_cmdLast_Click
139
140 End Sub
141 Private Sub cmdPrntReport_Click()
142 On Error GoTo Err_cmdPrntReport_Click
143
144     Dim stDocName As String
145
146     stDocName = "COOPERATORS"
147     DoCmd.OpenReport stDocName, acNormal
148
149 Exit_cmdPrntReport_Click:
150     Exit Sub
151
152 Err_cmdPrntReport_Click:
153     MsgBox Err.Description
154     Resume Exit_cmdPrntReport_Click
155
156 End Sub
```


Code

```
1 Attribute VB_Name = "Form_COOPERATORSsub"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Topic_NotInList(NewData As String, Response As Integer)
9
10 'If item is not in list, check then add to list if requested
11 Dim strMessage As String
12 Dim dbsRkts As Database
13 Dim rstTypes As Recordset
14
15 strMessage = "Add '" & NewData & "to the list of Main Topics?"
16
17 If Confirm(strMessage) Then
18
19     'open table and add record
20     Set dbsRkts = CurrentDb()
21     Set rstTypes = dbsRkts.OpenRecordset("coopTopic")
22     rstTypes.AddNew
23     rstTypes!Topic = NewData
24     rstTypes.Update
25     Response = acDataErrAdded 'requery list
26 Else
27     Response = acDataErrDisplay ' display error
28 End If
29
30 End Sub
```

Code

```

1 Attribute VB_Name = "Form_COOPERATORsubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdOpenCoopFrm_Click()
9 On Error GoTo Err_cmdOpenCoopFrm_Click
10
11     'Save record first
12     DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
13
14     Dim stDocName As String
15     Dim stLinkCriteria As String
16     'Dim stMsg As String
17     'Dim stTitle As String
18
19     'Check to see if there is a valid record selected before opening form
20     'If IsNull(Me![CoopKey]) Then 'Notify user and then exit sub
21     '    stMsg = "You must enter basic Cooperator information before you can select
this option"
22     '    stTitle = "Select a Record and then Continue"
23     '        If MsgBox(stMsg, vbOKOnly, stTitle) = vbOK Then
24     '            Exit Sub
25     '        End If
26     'End If
27
28     stDocName = "COOPERATORS"
29
30     stLinkCriteria = "[CoopKey]=" & Me![CoopKey]
31     DoCmd.OpenForm stDocName, , , stLinkCriteria
32
33 Exit_cmdOpenCoopFrm_Click:
34     Exit Sub
35
36 Err_cmdOpenCoopFrm_Click:
37     MsgBox Err.Description
38     Resume Exit_cmdOpenCoopFrm_Click
39
40 End Sub
41
42 Private Sub Form_Current()
43
44 Me!CoopDate.SetFocus
45
46 End Sub
47
48 Private Sub Form_Unload(Cancel As Integer)
49
50 Me!CoopDate.SetFocus
51 Me.Repaint
52
53 End Sub

```

Code

```
1 Attribute VB_Name = "Form_CurrentPrisoners"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Combo21_BeforeUpdate(Cancel As Integer)
9
10 End Sub
11
12
13 Private Sub Form_Current()
14
15 'Synchronize combobox with form
16 Combo21 = Me!FullName
17
18 End Sub
19
20
21 Private Sub Form_Load()
22
23 End Sub
24
25 Private Sub cmdImportRackets_Click()
26 On Error GoTo Err_cmdImportRackets_Click
27
28 'Confirm transfer
29 Dim strMsg As String
30 strMsg = "You are about to import a record from PTS to RACKETS, Continue?"
31 If MsgBox(strMsg, vbOKCancel, "Error!") = vbCancel Then
32     Exit Sub
33 Else
34 End If
35
36 End Sub
37
38 Private Sub Form_Open(Cancel As Integer)
39
40 Me.pr_street.SetFocus
41
42 End Sub
43
44 Private Sub FormFooter_Click()
45
46 End Sub
47
48 Private Sub FormHeader_Click()
49
50 End Sub
51 Sub Combo21_AfterUpdate()
52     ' Find the record that matches the control.
53     Me.RecordsetClone.FindFirst "[FullName] = '" & Me![Combo21] & "'"
54     Me.Bookmark = Me.RecordsetClone.Bookmark
55 End Sub
56
57
58 Private Sub pr_citizen_code_BeforeUpdate(Cancel As Integer)
59
60 End Sub
61
62 Private Sub pr_citizen_code_Click()
63
64 End Sub
```

```
65 Private Sub cmdUpdateRec_Click()
66 On Error GoTo Err_cmdUpdateRec_Click
67
68     Dim stDocName As String
69     'Dim stDocName2 As String
70
71     'Set dbs = CurrentDb
72     'Set rst = dbs.OpenRecordset("DEFENDANT", dbOpenDynaset)
73
74     stDocName = "qUpdateWPTS"
75     'stDocName2 = "qApndArrest"
76
77     DoCmd.OpenQuery stDocName, acNormal, acEdit
78
79 Exit_cmdUpdateRec_Click:
80     Exit Sub
81
82 Err_cmdUpdateRec_Click:
83     MsgBox Err.Description
84     Resume Exit_cmdUpdateRec_Click
85
86 End Sub
87 Private Sub cmdNext_Click()
88 On Error GoTo Err_cmdNext_Click
89
90
91     DoCmd.GoToRecord , , acNext
92
93 Exit_cmdNext_Click:
94     Exit Sub
95
96 Err_cmdNext_Click:
97     MsgBox Err.Description
98
99     Resume Exit_cmdNext_Click
100 End Sub
101 Private Sub cmdPrevious_Click()
102 On Error GoTo Err_cmdPrevious_Click
103
104
105     DoCmd.GoToRecord , , acPrevious
106
107 Exit_cmdPrevious_Click:
108     Exit Sub
109
110 Err_cmdPrevious_Click:
111     MsgBox Err.Description
112     Resume Exit_cmdPrevious_Click
113
114 End Sub
115 Private Sub cmdClose_Click()
116 On Error GoTo Err_cmdClose_Click
117
118
119     DoCmd.Close
120
121 Exit_cmdClose_Click:
122     Exit Sub
123
124 Err_cmdClose_Click:
125     MsgBox Err.Description
126     Resume Exit_cmdClose_Click
127
128 End Sub
129 Private Sub cmdArrest_Click()
```

```
130 On Error GoTo Err_cmdArrest_Click
131
132     Dim stDocName As String
133
134     stDocName = "qApndArrest"
135     DoCmd.OpenQuery stDocName, acNormal, acEdit
136
137 Exit_cmdArrest_Click:
138     Exit Sub
139
140 Err_cmdArrest_Click:
141     MsgBox Err.Description
142     Resume Exit_cmdArrest_Click
143
144 End Sub
```

Code

```

1 Attribute VB_Name = "Form_DEFENDANT"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cboFindName_BeforeUpdate(Cancel As Integer)
9
10 End Sub
11
12 Private Sub cmdClose_Click()
13 On Error GoTo Err_cmdClose_Click
14
15
16     DoCmd.Close
17
18 Exit_cmdClose_Click:
19     Exit Sub
20
21 Err_cmdClose_Click:
22     MsgBox Err.Description
23     Resume Exit_cmdClose_Click
24
25 End Sub
26
27 Private Sub cmdSubmit_Click()
28
29 Dim rst As Recordset
30
31     Dim strCriteria As String
32     Dim strNYSID As String
33     Dim strUSMS As String
34
35     'Make sure there is an entry in the text box before running
36     If IsNull(txtFindBy) Then
37         MsgBox "Enter a NYSID or USMS numer and then submit query"
38         Exit Sub
39     End If
40
41 Select Case grpFindOptions
42
43
44     Case 1 'Find case using NYSID number
45         Set rst = Me.RecordsetClone
46         strNYSID = Me!txtFindBy
47
48         strCriteria = "[NYSID]='" & strNYSID & "'"
49         rst.FindFirst strCriteria
50         If rst.NoMatch Then
51             MsgBox "The number you entered is not valid."
52         Else
53             Me.Bookmark = Me.RecordsetClone.Bookmark
54         End If
55         Me!txtFindBy.Value = ""
56     Case 2 'Find case using USMS number
57         Set rst = Me.RecordsetClone
58         strUSMS = Me!txtFindBy
59         strCriteria = "[USMSNum]='" & strUSMS & "'"
60         rst.FindFirst strCriteria
61         If rst.NoMatch Then
62             MsgBox "The number you entered is not valid."
63         Else
64             Me.Bookmark = Me.RecordsetClone.Bookmark
65         End If

```

```

65         Me!txtFindBy.Value = ""
66 End Select
67
68 End Sub
69
70 Sub cboFindName_AfterUpdate()
71
72     Dim rst As Recordset
73
74     ' Find the record that matches the control.
75     If IsNull(Me![cboFindName]) Then
76         Exit Sub
77     End If
78
79     Set rst = Me.RecordsetClone
80     rst.FindFirst "[DefKey] = " & Me![cboFindName]
81     'Me.RecordsetClone.FindFirst "[DefKey] = " & Me![cboFindName]
82     If rst.NoMatch Then
83         rst.Close
84         Exit Sub
85     End If
86     Me.Bookmark = rst.Bookmark
87
88 End Sub
89
90 Private Sub cmdNext_Click()
91 On Error GoTo Err_cmdNext_Click
92
93
94     DoCmd.GoToRecord , , acNext
95
96 Exit_cmdNext_Click:
97     Exit Sub
98
99 Err_cmdNext_Click:
100     MsgBox Err.Description
101     Resume Exit_cmdNext_Click
102
103 End Sub
104 Private Sub cmdPrevious_Click()
105 On Error GoTo Err_cmdPrevious_Click
106
107
108     DoCmd.GoToRecord , , acPrevious
109
110 Exit_cmdPrevious_Click:
111     Exit Sub
112
113 Err_cmdPrevious_Click:
114     MsgBox Err.Description
115     Resume Exit_cmdPrevious_Click
116
117 End Sub
118
119 Private Sub Coop_AfterUpdate()
120
121 'If cooperator is true set cooperator page visible
122 If (Me!Coop = True) Then
123     Me!tpgCooperator.Visible = True
124
125 ElseIf (Me!Coop = False) Then
126     Me!tpgCooperator.Visible = False
127 Else
128 End If
129
130 End Sub
131
132 Private Sub Coop_BeforeUpdate(Cancel As Integer)

```

```
133
134
135 End Sub
136
137 Private Sub Coop_Click()
138
139 End Sub
140
141 Private Sub CrimeType_NotInList(NewData As String, Response As Integer)
142
143 Dim strMessage As String
144 Dim dbsRkts As Database
145 Dim rstTypes As Recordset
146
147 strMessage = "Add '" & NewData & "to the list of Crime Types?"
148
149 If Confirm(strMessage) Then
150
151     'open table and add record
152     Set dbsRkts = CurrentDb()
153     Set rstTypes = dbsRkts.OpenRecordset("crime_type")
154     rstTypes.AddNew
155     rstTypes!CrimeType = NewData
156     rstTypes.Update
157     Response = acDataErrAdded 'requery list
158 Else
159     Response = acDataErrDisplay 'display error
160 End If
161
162 End Sub
163
164 Private Sub Detail_Click()
165
166 End Sub
167
168 Private Sub Form_AfterUpdate()
169
170 'Enable or disable mugshots button
171 If IsNull(Me!pr_key) Or IsNull(Me!txtMugshot) Then
172     Me!cmdPhoto.Enabled = False
173 Else
174     Me!cmdPhoto.Enabled = True
175 End If
176
177 'Enable or disable insert mugshots button
178 If IsNull(Me!txtMugshot) And Not (IsNull(Me!pr_key)) Then
179     Me!cmdEnterPhoto.Enabled = True
180 Else
181     Me!cmdEnterPhoto.Enabled = False
182 End If
183
184 End Sub
185
186 Private Sub Form_BeforeUpdate(Cancel As Integer)
187
188 End Sub
189
190 Private Sub Form_Current()
191
192 'Set Find Record to current
193 cboFindName = DefKey
194
195 'If cooperator is true set cooperator page visible
196 If (Me!Coop = True) Then
197     Me!tpgCooperator.Visible = True
198
199 ElseIf (Me!Coop = False) Then
```



```

200     Me!tpgCooperator.Visible = False
201 Else
202 End If
203
204 'If Weapon seized is true set cooperator page visible
205 If (Me!Weapon = True) Then
206     Me!tpgFIREARMS.Visible = True
207
208 ElseIf (Me!Weapon = False) Then
209     Me!tpgFIREARMS.Visible = False
210 Else
211 End If
212
213 'Enable or disable mugshots button
214 If IsNull(Me!pr_key) Or IsNull(Me!txtMugshot) Then
215     Me!cmdPhoto.Enabled = False
216 Else
217     Me!cmdPhoto.Enabled = True
218 End If
219
220 'Enable or disable insert mugshots button
221 If IsNull(Me!txtMugshot) And Not (IsNull(Me!pr_key)) Then
222     Me!cmdEnterPhoto.Enabled = True
223 Else
224     Me!cmdEnterPhoto.Enabled = False
225 End If
226
227 End Sub
228 Private Sub cmdOpenCooperatorFrm_Click()
229 On Error GoTo Err_cmdOpenCooperatorFrm_Click
230
231     Dim stDocName As String
232     Dim stLinkCriteria As String
233
234     stDocName = "COOPERATORS"
235
236     stLinkCriteria = "[DefKey]=" & Me![DefKey]
237     DoCmd.OpenForm stDocName, , , stLinkCriteria
238
239 Exit_cmdOpenCooperatorFrm_Click:
240     Exit Sub
241
242 Err_cmdOpenCooperatorFrm_Click:
243     MsgBox Err.Description
244     Resume Exit_cmdOpenCooperatorFrm_Click
245
246 End Sub
247 Private Sub cmdOpenCooperatorForm_Click()
248 On Error GoTo Err_cmdOpenCooperatorForm_Click
249
250     Dim stDocName As String
251
252     Dim stLinkCriteria As String
253
254     stDocName = "COOPERATORS"
255
256     stLinkCriteria = "[DefKey]=" & Me!DefKey
257     DoCmd.OpenForm stDocName, , , stLinkCriteria, acAdd, , Me!DefKey
258
259 Exit_cmdOpenCooperatorForm_Click:
260     Exit Sub
261
262 Err_cmdOpenCooperatorForm_Click:
263     MsgBox Err.Description
264     Resume Exit_cmdOpenCooperatorForm_Click
265

```

266 End Sub
Form: DEFENDANT

```
267
268 Private Sub Form_Error(DataErr As Integer, Response As Integer)
269     Debug.Print "DataErr = "; DataErr
270 End Sub
271
272
273 Private Sub Frame99_BeforeUpdate(Cancel As Integer)
274
275 End Sub
276
277 Private Sub Frame112_BeforeUpdate(Cancel As Integer)
278
279 End Sub
280
281
282 Private Sub grpFindOptions_BeforeUpdate(Cancel As Integer)
283
284 End Sub
285
286 Private Sub LeadDef_AfterUpdate()
287
288 End Sub
289
290 Private Sub LeadDef_Click()
291
292 End Sub
293
294 Private Sub txtFindBy_BeforeUpdate(Cancel As Integer)
295
296 End Sub
297 Private Sub cmdOpenCaseFrm_Click()
298 On Error GoTo Err_cmdOpenCaseFrm_Click
299
300     Dim stDocName As String
301     Dim stLinkCriteria As String
302
303     stDocName = "CASE"
304     'stDocName1 = "DEFENDANT"
305
306     stLinkCriteria = "[CaseKey]=" & Me![CaseKey]
307     DoCmd.OpenForm stDocName, , , stLinkCriteria
308
309
310     'Close form
311     DoCmd.Close acForm, "DEFENDANT", acSaveYes
312
313
314 Exit_cmdOpenCaseFrm_Click:
315     Exit Sub
316
317 Err_cmdOpenCaseFrm_Click:
318     MsgBox Err.Description
319     Resume Exit_cmdOpenCaseFrm_Click
320
321 End Sub
322
323 Private Sub Weapon_AfterUpdate()
324
325 'If Weapon seized is true set cooperator page visible
326 If (Me!Weapon = True) Then
327     Me!tpgFIREARMS.Visible = True
328
329 ElseIf (Me!Weapon = False) Then
330     Me!tpgFIREARMS.Visible = False
331 Else
332 End If
```

```
334 End Sub
335
336 Private Sub Weapon_Click()
337
338 End Sub
339 Private Sub cmdPrintRpt_Click()
340 On Error GoTo Err_cmdPrintRpt_Click
341
342     Dim stDocName As String
343
344     stDocName = "DEFENDANT"
345     DoCmd.OpenReport stDocName, acNormal
346
347 Exit_cmdPrintRpt_Click:
348     Exit Sub
349
350 Err_cmdPrintRpt_Click:
351     MsgBox Err.Description
352     Resume Exit_cmdPrintRpt_Click
353
354 End Sub
355 Private Sub cmdPhoto_Click()
356 On Error GoTo Err_cmdPhoto_Click
357
358     Dim stDocName As String
359     Dim stLinkCriteria As String
360
361     stDocName = "frmDEF_PHOTOS"
362
363     stLinkCriteria = "[DefKey]=" & Me![DefKey]
364     DoCmd.OpenForm stDocName, , , stLinkCriteria
365
366     'do not allow records to be edited
367     Forms![frmDEF_PHOTOS].oleMugA.Locked = True
368     Forms![frmDEF_PHOTOS].oleMugB.Locked = True
369
370 Exit_cmdPhoto_Click:
371     Exit Sub
372
373 Err_cmdPhoto_Click:
374     MsgBox Err.Description
375     Resume Exit_cmdPhoto_Click
376
377 End Sub
378 Private Sub cmdEnterPhoto_Click()
379 On Error GoTo Err_cmdEnterPhoto_Click
380
381     Dim stDocName As String
382     Dim stLinkCriteria As String
383
384     stDocName = "frmDEF_PHOTOS"
385
386     stLinkCriteria = "[DefKey]=" & Me![DefKey]
387     DoCmd.OpenForm stDocName, , , stLinkCriteria
388
389     'Allow records to be edited
390     Forms![frmDEF_PHOTOS].oleMugA.Locked = False
391     Forms![frmDEF_PHOTOS].oleMugB.Locked = False
392
393 Exit_cmdEnterPhoto_Click:
394     Exit Sub
395
396 Err_cmdEnterPhoto_Click:
397     MsgBox Err.Description
398     Resume Exit_cmdEnterPhoto_Click
399
```

400 End Sub

Code

```

1 Attribute VB_Name = "Form_DEFENDANTList"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8
9 Private Sub Form_Current()
10
11 'Enable or disable Details button
12 If IsNull(Me![DefKey]) Then
13     Me!cmdOpenDefFrm.Enabled = False
14 Else
15     Me!cmdOpenDefFrm.Enabled = True
16 End If
17
18 'Enable or disable PTS button
19 If IsNull(Me!USMSNum) Then
20     Me!cmdSearchPTS.Enabled = True
21 Else
22     Me!cmdSearchPTS.Enabled = False
23 End If
24 End Sub
25
26 Private Sub Form_Load()
27
28 'Enable or disable Details button
29 If IsNull(Me![DefKey]) Then
30     Me!cmdOpenDefFrm.Enabled = False
31 Else
32     Me!cmdOpenDefFrm.Enabled = True
33 End If
34
35 End Sub
36
37 Private Sub Form_Open(Cancel As Integer)
38
39 End Sub
40 Private Sub cmdOpenDefFrm_Click()
41 On Error GoTo Err_cmdOpenDefFrm_Click
42
43
44     Dim stDocName As String
45     Dim stLinkCriteria As String
46     Dim stTitle As String
47
48     Dim stMsg As String
49     Dim stMsgTrans As String
50     Dim stTitleTrans As String
51     Dim IndctNum As Variant
52     Dim TopChg As Variant
53
54     stDocName = "DEFENDANT"
55
56     stLinkCriteria = "[DefKey]=" & Me![DefKey]
57
58     'Save record first
59     DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
60
61     'Check to see if there is a valid record selected before opening form
62     'If IsNull(Me![DefKey]) Then 'Notify user and then exit sub
63     *     stMsg = "You must be at a valid record before you can select this option"

```

```

64     ' stTitle = "Select a Record and then Continue"
65     '     If MsgBox(stMsg, vbOKOnly, stTitle) = vbOK Then
66     '         Exit Sub
67     '     End If
68     'End If
69
70     'get incictment# and top charge for use in transfer
71     IndctNum = Forms![CASE]!IndictmentNum.Value
72     TopChg = Forms![CASE]!StatuteNum.Value
73
74
75     DoCmd.OpenForm stDocName, , , stLinkCriteria
76
77     'Set focus to Middle Name
78     Forms![Defendant]!DefMName.SetFocus
79
80     'Disable all FindBy options
81     Forms![Defendant]!grpFindOptions.Enabled = False
82     Forms![Defendant]!cboFindName.Enabled = False
83     Forms![Defendant]!cmdSubmit.Enabled = False
84     Forms![Defendant]!txtFindBy.Enabled = False
85     'Remove navigation buttons
86     Forms![Defendant]!cmdPrevious.Visible = False
87     Forms![Defendant]!cmdNext.Visible = False
88
89     'Check if Case Instrument# & TopCharge should be entered for defendant
90     'If IsNull(Forms!Defendant!tpgStatutes.Form.Controls!Count) And
IsNull(Forms!Defendant!tpgStatutes.Form.Controls!StatuteNum) Then
91     '     stMsgTrans = "Enter Case Instrument no. " & IndctNum & " and Top Charge " &
TopChg & " for Defendant?"
92     '     stTitleTrans = "RACKETS"
93     '     If MsgBox(stMsgTrans, vbYesNo, stTitleTrans) = vbYes Then
94     '         Forms![Defendant]![STATUTESsubform].Count.Value = 1
95     '         Forms![Defendant]![STATUTESsubform].StatuteNum.Value = TopChg
96     '     Else
97     '     End If
98     'End If
99
100 Exit_cmdOpenDefFrm_Click:
101     Exit Sub
102
103 Err_cmdOpenDefFrm_Click:
104     MsgBox Err.Description
105     Resume Exit_cmdOpenDefFrm_Click
106
107 End Sub
108 Private Sub cmdSearchPTS_Click()
109 On Error GoTo Err_cmdSearchPTS_Click
110
111     Dim stDocName As String
112     Dim stLinkCriteria As String
113
114     stDocName = "CurrentPrisoners"
115     DoCmd.OpenForm stDocName, , , stLinkCriteria
116
117 Exit_cmdSearchPTS_Click:
118     Exit Sub
119
120 Err_cmdSearchPTS_Click:
121     MsgBox Err.Description
122     Resume Exit_cmdSearchPTS_Click
123
124 End Sub

```

Code

```
1 Attribute VB_Name = "Form_EVENTsubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8
9
10 Private Sub Form_AfterInsert()
11
12 Dim strMessage As String
13 Dim strLinkCriteria As String
14
15 strMessage = "Enter Proffer information now? At the very least, a topic must be entered
    to continue."
16
17 If Me.event_type = "Proffer" Then
18     If Forms![Defendant].Coop.Value = False Then
19         Forms![Defendant].Coop.Value = True
20         Forms![Defendant]!tpgCooperator.Visible = True
21     End If
22     Forms![Defendant]!subCooperator.Form.Status = "PROF"
23     'If Confirm(strMessage) Then
24         Forms![Defendant]!tpgCooperator.SetFocus
25         Forms![Defendant]!subCooperator.Form.Topic.SetFocus
26     'Else
27     'End If
28 End If
29
30 End Sub
```

Code

```
1 Attribute VB_Name = "Form_FIREARMSsubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub FirearmType_NotInList(NewData As String, Response As Integer)
9
10 Dim strMessage As String
11 Dim dbsRkts As Database
12 Dim rstTypes As Recordset
13
14 strMessage = "Add '" & NewData & "to the list of Firearm Types?"
15
16 If Confirm(strMessage) Then
17
18     'open table and add record
19     Set dbsRkts = CurrentDb()
20     Set rstTypes = dbsRkts.OpenRecordset("firearm_type")
21     rstTypes.AddNew
22     rstTypes!FirearmType = NewData
23     rstTypes.Update
24     Response = acDataErrAdded 'requery list
25 Else
26     Response = acDataErrDisplay ' display error
27 End If
28
29
30 End Sub
31
32 Private Sub Make_NotInList(NewData As String, Response As Integer)
33
34 Dim strMessage As String
35 Dim dbsRkts As Database
36 Dim rstTypes As Recordset
37
38 strMessage = "Add '" & NewData & "to the list of Firearm Makes?"
39
40 If Confirm(strMessage) Then
41
42     'open table and add record
43     Set dbsRkts = CurrentDb()
44     Set rstTypes = dbsRkts.OpenRecordset("firearm_make")
45     rstTypes.AddNew
46     rstTypes!Make = NewData
```


Form: FIREARMSsubform

```
47     rstTypes.Update
48     Response = acDataErrAdded 'requery list
49 Else
50     Response = acDataErrDisplay ' display error
51 End If
52
53
54 End Sub
55
56 Private Sub Model_NotInList(NewData As String, Response As Integer)
57
58 Dim strMessage As String
59 Dim dbsRkts As Database
60 Dim rstTypes As Recordset
61
62 strMessage = "Add '" & NewData & "to the list of Firearm Models?"
63
64 If Confirm(strMessage) Then
65
66     'open table and add record
67     Set dbsRkts = CurrentDb()
68     Set rstTypes = dbsRkts.OpenRecordset("firearm_model")
69     rstTypes.AddNew
70     rstTypes!Model = NewData
71     rstTypes.Update
72     Response = acDataErrAdded 'requery list
73 Else
74     Response = acDataErrDisplay ' display error
75 End If
76
77
78 End Sub
```

Code

```
1 Attribute VB_Name = "Form_frmDEF_PHOTOS"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdPrint_Click()
9 On Error GoTo Err_cmdPrint_Click
10
11
12     DoCmd.PrintOut
13
14 Exit_cmdPrint_Click:
15     Exit Sub
16
17 Err_cmdPrint_Click:
18     MsgBox Err.Description
19     Resume Exit_cmdPrint_Click
20
21 End Sub
22 Private Sub cmdClose_Click()
23 On Error GoTo Err_cmdClose_Click
24
25     'save record first
26     DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
27
28     DoCmd.Close
29
30 Exit_cmdClose_Click:
31     Exit Sub
32
33 Err_cmdClose_Click:
34     MsgBox Err.Description
35     Resume Exit_cmdClose_Click
36
37 End Sub
```

Code

```
1 Attribute VB_Name = "Form_frmDefsByAUSA"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdPreview_Click()
9 On Error GoTo Err_cmdPreview_Click
10
11 Dim stDocName As String
12
13 stDocName = "rptDefsByAUSA"
14 DoCmd.OpenReport stDocName, acPreview
15 DoCmd.Close acForm, "frmDefsByAUSA"
16
17 Exit_cmdPreview_Click:
18 Exit Sub
19
20 Err_cmdPreview_Click:
21 MsgBox Err.Description
22 Resume Exit_cmdPreview_Click
23
24 End Sub
25 Private Sub cmdPrint_Click()
26 On Error GoTo Err_cmdPrint_Click
27
28 Dim stDocName As String
29
30 stDocName = "rptDefsByAUSA"
31 DoCmd.OpenReport stDocName, acNormal
32 DoCmd.Close acForm, "frmDefsByAUSA"
33
34 Exit_cmdPrint_Click:
35 Exit Sub
36
37 Err_cmdPrint_Click:
38 MsgBox Err.Description
39 Resume Exit_cmdPrint_Click
40
41 End Sub
42 Private Sub cmdCancel_Click()
43 On Error GoTo Err_cmdCancel_Click
44
45
46 DoCmd.Close
47
48 Exit_cmdCancel_Click:
49 Exit Sub
50
51 Err_cmdCancel_Click:
52 MsgBox Err.Description
53 Resume Exit_cmdCancel_Click
54
55 End Sub
```

Code

```
1 Attribute VB_Name = "Form_frmDefsByUnit"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdPreview_Click()
9 On Error GoTo Err_cmdPreview_Click
10
11     Dim stDocName As String
12
13     stDocName = "rptDefsByUnit"
14     DoCmd.OpenReport stDocName, acPreview
15     DoCmd.Close acForm, "frmDefsByUnit"
16
17 Exit_cmdPreview_Click:
18     Exit Sub
19
20 Err_cmdPreview_Click:
21     MsgBox Err.Description
22     Resume Exit_cmdPreview_Click
23
24 End Sub
25 Private Sub cmdPrint_Click()
26 On Error GoTo Err_cmdPrint_Click
27
28     Dim stDocName As String
29
30     stDocName = "rptDefsByUnit"
31     DoCmd.OpenReport stDocName, acNormal
32     DoCmd.Close acForm, "frmDefsByUnit"
33
34 Exit_cmdPrint_Click:
35     Exit Sub
36
37 Err_cmdPrint_Click:
38     MsgBox Err.Description
39     Resume Exit_cmdPrint_Click
40
41 End Sub
42 Private Sub cmdCancel_Click()
43 On Error GoTo Err_cmdCancel_Click
44
45
46     DoCmd.Close
47
48 Exit_cmdCancel_Click:
49     Exit Sub
50
51 Err_cmdCancel_Click:
52     MsgBox Err.Description
53     Resume Exit_cmdCancel_Click
54
55 End Sub
```

Code

```

1 Attribute VB_Name = "Form_frmFindAlias"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdClear_Click()
9
10 Dim strSql As String
11 Dim strSqlFull As String
12
13 Me!txtAlias = ""
14
15 strSql = "WHERE FALSE"
16 strSqlFull = "Select * From qFindAlias " & strSql
17 Me!subFindAlias.Form.RecordSource = strSqlFull
18
19 strSql = "Select * From qFindAKA " & strSql
20 Me!subFindAKA.Form.RecordSource = strSqlFull
21
22 End Sub
23
24 Private Sub cmdSubmit_Click()
25
26 Dim strAlias As String
27 Dim strSqlAlias As String
28 Dim strSqlFull As String
29 Dim strMsg As String
30 Dim strTitle As String
31 Dim strWhere As String
32
33 If IsNull(Me!txtAlias) Or Me!txtAlias = "" Then
34     strMsg = "Query is empty. Enter an alias and try again."
35     strTitle = "Empty Query!"
36     If MsgBox(strMsg, vbOKOnly, strTitle) = vbOK Then
37         Exit Sub
38     End If
39 End If
40
41 strWhere = "Where "
42
43 Select Case grpDatabase
44 'Search PTS
45 Case 1
46     If grpMatch.Value = 1 Then
47         strSqlAlias = "([name_fname] = " & "Forms!frmFindAlias![txtAlias]OR[name_lname] = " & "Forms!frmFindAlias![txtAlias]OR[name_mname] = " & "Forms!frmFindAlias![txtAlias]OR[rm_desc] = " & "Forms!frmFindAlias![txtAlias])"
48
49         strSqlAliasSql = strWhere & strSqlAlias
50         strSqlFull = "Select * From qFindAKA " & strSqlAliasSql
51         Me!subFindAKA.Form.RecordSource = strSqlFull
52     Else
53         strSqlAliasSql = " Where [name_fname] Like '" & Forms!frmFindAlias![txtAlias] & "' or [name_lname] Like '" & Forms!frmFindAlias![txtAlias] & "' or [name_mname] Like '" & Forms!frmFindAlias![txtAlias] & "' or [rm_desc] Like '" & Forms!frmFindAlias![txtAlias] & "'"
54         strSqlFull = "Select * From qFindAKA " & strSqlAliasSql
55         Me!subFindAKA.Form.RecordSource = strSqlFull
56     End If
57 End Sub

```

57 'Search Rackets
Form: frmFindAlias

```
58 Case 2
59     If grpMatch.Value = 1 Then
60         strAlias = "[Alias] = " & "Forms!frmFindAlias![txtAlias]OR[AliasLN] = " &
"Forms!frmFindAlias![txtAlias]"
61         strAliasSql = strWhere & strAlias
62         strFullSql = "Select * From qFindAlias " & strAliasSql
63         Me!subFindAlias.Form.RecordSource = strFullSql
64     Else
65         strAliasSql = " Where [Alias] Like '" & Forms!frmFindAlias![txtAlias] & "' or
[AliasLN] Like '" & Forms!frmFindAlias![txtAlias] & "'"
66         strFullSql = "Select * From qFindAlias " & strAliasSql
67         Me!subFindAlias.Form.RecordSource = strFullSql
68     End If
69 End Select
70
71 End Sub
72
73 Private Sub Form_AfterUpdate()
74
75 End Sub
76
77 Private Sub Form_Load()
78
79 End Sub
80
81 Private Sub Frame4_BeforeUpdate(Cancel As Integer)
82
83 End Sub
84
85 Private Sub grpDatabase_AfterUpdate()
86
87 'Set subform visible based on selection of search
88 Select Case grpDatabase
89
90 Case 1
91 Me!subFindAKA.Form.Visible = True
92 Me!subFindAlias.Form.Visible = False
93
94 Case 2
95 Me!subFindAKA.Form.Visible = False
96 Me!subFindAlias.Form.Visible = True
97
98 End Select
99
100 End Sub
101
102 Private Sub txtAlias_BeforeUpdate(Cancel As Integer)
103
104 End Sub
105 Private Sub cmdClose_Click()
106 On Error GoTo Err_cmdClose_Click
107
108
109     DoCmd.Close
110
111 Exit_cmdClose_Click:
112     Exit Sub
113
114 Err_cmdClose_Click:
115     MsgBox Err.Description
116     Resume Exit_cmdClose_Click
117
118 End Sub
119 Private Sub cmdOpenDefFrm_Click()
120 On Error GoTo Err_cmdOpenDefFrm_Click
121
122     Dim stDocName As String
```

```
123     Dim stLinkCriteria As String
124
Form: frmFindAlias
```

```
125     'stLinkCriteria = Me!subFindAKA.Form.
126
127     stDocName = "DEFENDANT"
128     DoCmd.OpenForm stDocName, , , stLinkCriteria
129
130 Exit_cmdOpenDefFrm_Click:
131     Exit Sub
132
133 Err_cmdOpenDefFrm_Click:
134     MsgBox Err.Description
135     Resume Exit_cmdOpenDefFrm_Click
136
137 End Sub
```

Code

```

1 Attribute VB_Name = "Form_frmFindAssoc"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdClear_Click()
9 Dim strSql As String
10 Dim strFull As String
11
12 Me!txtFind = ""
13
14 strSql = "Where False"
15 strFull = "Select * From qFindAssoc " & strSql
16 Me!subFindAssoc.Form.RecordSource = strFull
17 Me!subFindLocale.Form.RecordSource = strFull
18 Me!subFindCohort.Form.RecordSource = strFull
19
20 End Sub
21
22 Private Sub cmdClose_Click()
23 On Error GoTo Err_cmdClose_Click
24
25
26     DoCmd.Close
27
28 Exit_cmdClose_Click:
29     Exit Sub
30
31 Err_cmdClose_Click:
32     MsgBox Err.Description
33     Resume Exit_cmdClose_Click
34
35 End Sub
36
37 Private Sub cmdSubmit_Click()
38
39
40 Dim strAssoc As String
41 Dim strSql As String
42 Dim strMsg As String
43 Dim strTitle As String
44
45 'clear variables
46 'strAssoc = ""
47
48 'strSql = ""
49
50 'First check to see if a string was entered, if not end routine
51 strMsg = "Query is empty. Enter a Search String and try again"
52 strTitle = "RACKETS"
53 If IsNull(Me!txtFind) Or Me!txtFind = "" Then
54     If MsgBox(strMsg, vbOKOnly, strTitle) = vbOK Then
55         Exit Sub
56     End If
57 End If
58
59 'run query based on selection type
60 Select Case grpType
61 'search Group table via qFindAssoc
62 Case 1
63     strAssoc = "Where [GroupName] Like '" & Forms!frmFindAssoc!{txtFind} & "'"
64     strSql = "Select * From qFindAssoc " & strAssoc

```



```

65     Me!subFindAssoc.Form.RecordSource = strSql
66
67 'search locale via q FindLocale
68 Case 2
69
70     strSql = "Where [lo_name] Like '" & Forms!frmFindAssoc![txtFind] & "'"
71     strSql = "Select * From qFindLocale " & strSql
72     Me!subFindLocale.Form.RecordSource = strSql
73
74 'search cohort via qFindCohort
75 Case 3
76     strSql = "Where [co_name] Like '" & Forms!frmFindAssoc![txtFind] & "'" or
[co_name] Like '" & Forms!frmFindAssoc![txtFind] & "'"
77     strSql = "Select * From qFindCohort " & strSql
78     Me!subFindCohort.Form.RecordSource = strSql
79
80 End Select
81 End Sub
82
83 Private Sub Form_Close()
84 Me!txtFind = ""
85 End Sub
86
87 Private Sub Form_Open(Cancel As Integer)
88
89 Dim strSql As String
90 Dim strFull As String
91
92 'Clear last query
93 Me!txtFind = ""
94
95 strSql = "Where False"
96 strFull = "Select * From qFindAssoc " & strSql
97
98 Me!subFindAssoc.Form.RecordSource = strFull
99
100 'Set default visible and all others not visible
101 Me!subFindAssoc.Form.Visible = True
102 Me!subFindCohort.Form.Visible = False
103 Me!subFindLocale.Form.Visible = False
104
105 End Sub
106
107 Private Sub grpType_AfterUpdate()
108
109 'Set the subform based on the query type
110 Select Case grpType
111
112 Case 1
113     Me!subFindAssoc.Form.Visible = True
114     Me!subFindLocale.Form.Visible = False
115     Me!subFindCohort.Form.Visible = False
116
117 Case 2
118     Me!subFindLocale.Form.Visible = True
119     Me!subFindAssoc.Form.Visible = False
120     Me!subFindCohort.Form.Visible = False
121
122 Case 3
123     Me!subFindCohort.Form.Visible = True
124     Me!subFindAssoc.Form.Visible = False
125     Me!subFindLocale.Form.Visible = False
126
127 End Select
128
129 End Sub

```

Code

```
1 Attribute VB_Name = "Form_frmFindCounsel"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cboFind_AfterUpdate()
9
10 Dim strSql As String
11 Dim strFull As String
12
13 strSql = "Select * From qFindCounsel Where [Counsel] = " &
14 "Forms!frmFindCounsel![cboFind]"
15 Me!subFindCounsel.Form.RecordSource = strSql
16
17 End Sub
18
19 Private Sub cboFind_BeforeUpdate(Cancel As Integer)
20
21 End Sub
22
23 Private Sub cboFind_Change()
24 Dim strSql As String
25 Dim strFull As String
26
27 'strSql = "Select * From qFindCounsel Where [] = " & "Forms!frmFindCounsel![cboFind]"
28 'Me!subFindCounsel.Form.RecordSource = strSql
29
30
31 End Sub
32
33 Private Sub Detail_Click()
34
35 End Sub
36 Private Sub cmdClose_Click()
37 On Error GoTo Err_cmdClose_Click
38
39
40 DoCmd.Close
41
42 Exit_cmdClose_Click:
43 Exit Sub
44
45 Err_cmdClose_Click:
46 MsgBox Err.Description
47 Resume Exit_cmdClose_Click
48
49 End Sub
```

Code

```

1 Attribute VB_Name = "Form_frmFindEmployer"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdClear_Click()
9
10 Dim strSql As String
11 Dim strFull As String
12
13 Me!txtFind = ""
14
15 strSql = "Where False"
16 strFull = "Select * From qFindEmployer " & strSql
17 Me!subFindEmployer.Form.RecordSource = strFull
18 Me!subFindEmployerPTS.Form.RecordSource = strFull
19
20 End Sub
21
22 Private Sub cmdSubmit_Click()
23
24 Dim strEmploy As String
25 Dim strSql As String
26 Dim strMsg As String
27 Dim strTitle As String
28
29
30 'First check to see if a string was entered, if not end routine
31 strMsg = "Query is empty. Enter a Search String and try again"
32 strTitle = "RACKETS"
33 If IsNull(Me!txtFind) Or Me!txtFind = "" Then
34     If MsgBox(strMsg, vbOKOnly, strTitle) = vbOK Then
35         Exit Sub
36     End If
37 End If
38
39 'run query based on selection type
40 Select Case grpType
41
42 'search by employer
43 Case 1
44
45     If grpSearch.Value = 1 Then
46         strEmploy = "Where [em_name] Like '" & Forms!frmFindEmployer![txtFind] & "'"
47
48         strSql = "Select * From qFindEmployer " & strEmploy
49         Me!subFindEmployer.Form.RecordSource = strSql
50     Else
51         strEmploy = "Where [em_name] Like '" & Forms!frmFindEmployer![txtFind] & "'"
52
53         strSql = "Select * From qFindEmployPTS " & strEmploy
54         Me!subFindEmployerPTS.Form.RecordSource = strSql
55     End If
56
57 'search by owner/supervisor
58 Case 2
59     If grpSearch.Value = 1 Then
60         strEmploy = "Where [em_poc] Like '" & Forms!frmFindEmployer![txtFind] & "'"
61
62         strSql = "Select * From qFindEmployer " & strEmploy
63         Me!subFindEmployer.Form.RecordSource = strSql
64     Else
65         strEmploy = "Where [em_poc] Like '" & Forms!frmFindEmployer![txtFind] & "'"

```

```

63     strSql = "Select * From qFindEmployPTS " & strEmploy
64     Me!subFindEmployerPTS.Form.RecordSource = strSql
65     End If
66
67 'search by occupation
68 Case 3
69     If grpSearch.Value = 1 Then
70         strEmploy = "Where [em_occ] Like '" & Forms!frmFindEmployer![txtFind] & "'"
71
72         strSql = "Select * From qFindEmployer " & strEmploy
73         Me!subFindEmployerPTS.Form.RecordSource = strSql
74     Else
75         strEmploy = "Where [em_occ] Like '" & Forms!frmFindEmployer![txtFind] & "'"
76
77         strSql = "Select * From qFindEmployPTS " & strEmploy
78         Me!subFindEmployerPTS.Form.RecordSource = strSql
79     End If
80 End Select
81
82 End Sub
83
84 Private Sub Form_Open(Cancel As Integer)
85
86 Me!subFindEmployer.Form.Visible = True
87 Me!subFindEmployerPTS.Form.Visible = False
88
89 End Sub
90
91 Private Sub grpSearch_AfterUpdate()
92
93 'Set subform based on search selection
94
95 Select Case grpSearch
96 Case 1
97 'Search records linked to defendants in Rackets
98 Me!subFindEmployer.Form.Visible = True
99 Me!subFindEmployerPTS.Form.Visible = False
100
101 Case 2
102 'Search records not linked to an active defendant
103 Me!subFindEmployerPTS.Form.Visible = False
104 Me!subFindEmployerPTS.Form.Visible = True
105
106 End Select
107 End Sub
108
109 Private Sub grpSearch_BeforeUpdate(Cancel As Integer)
110
111 End Sub
112 Private Sub cmdClose_Click()
113 On Error GoTo Err_cmdClose_Click
114
115 DoCmd.Close
116
117 Exit_cmdClose_Click:
118 Exit Sub
119
120 Err_cmdClose_Click:
121 MsgBox Err.Description
122 Resume Exit_cmdClose_Click
123
124 End Sub
125 End Sub

```

Code

```

1 Attribute VB_Name = "Form_frmFindNarcs"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdClear_Click()
9
10 Dim strSql As String
11 Dim strSqlFull As String
12 Me!txtFind = ""
13 Me!cboType = ""
14
15 strSql = "Where False"
16 strSqlFull = "Select * From qFindNarc " & strSql
17 Me!subFindNarc.Form.RecordSource = strSqlFull
18
19 End Sub
20
21 Private Sub cmdSubmit_Click()
22
23 Dim strType As String
24 Dim strNarc As String
25 Dim strBoth As String
26 Dim strSql As String
27 Dim strMsg As String
28 Dim strTitle As String
29
30
31
32 Select Case grpSearch
33
34 Case 1
35     strType = "Where [NarcType]= " & "Forms!frmFindNarcs![cboType]"
36     strSql = "Select * From qFindNarc " & strType
37     Me!subFindNarc.Form.RecordSource = strSql
38
39 Case 2
40     strNarc = "Where [Brand]= " & "Forms!frmFindNarcs![txtFind]"
41     strSql = "Select * From qFindNarc " & strNarc
42     Me!subFindNarc.Form.RecordSource = strSql
43
44
45 Case 3
46     strType = "[NarcType]= " & "Forms!frmFindNarcs![cboType]"
47
48     strNarc = "[Brand]= " & "Forms!frmFindNarcs![txtFind]"
49     strBoth = "Where " & strType & " AND " & strNarc
50     strSql = "Select * From qFindNarc " & strBoth
51     Me!subFindNarc.Form.RecordSource = strSql
52
53 End Select
54
55 End Sub
56
57 Private Sub Detail_Click()
58
59 End Sub
60 Private Sub cmdClose_Click()
61 On Error GoTo Err_cmdClose_Click
62
63
64     DoCmd.Close

```

```
65
66 Exit_cmdClose_Click:
67     Exit Sub
68
69 Err_cmdClose_Click:
70     MsgBox Err.Description
71     Resume Exit_cmdClose_Click
72
73 End Sub
74
75 Private Sub Form_Open(Cancel As Integer)
76
77
78 Select Case grpSearch
79
80 Case 1
81     cboType.BackColor = 16777215
82     txtFind.BackColor = 12632256
83     Me.cboType.SetFocus
84 Case 2
85     txtFind.BackColor = 16777215
86     cboType.BackColor = 12632256
87     Me.txtFind.SetFocus
88 Case 3
89     cboType.BackColor = 16777215
90     txtFind.BackColor = 16777215
91     Me.cboType.SetFocus
92 End Select
93
94 End Sub
95
96 Private Sub grpSearch_AfterUpdate()
97 Select Case grpSearch
98
99 Case 1
100     cboType.BackColor = 16777215
101     txtFind.BackColor = 12632256
102     Me.cboType.SetFocus
103     Me!txtFind = ""
104
105 Case 2
106     txtFind.BackColor = 16777215
107     cboType.BackColor = 12632256
108     Me.txtFind.SetFocus
109     Me!cboType = ""
110
111 Case 3
112     cboType.BackColor = 16777215
113     txtFind.BackColor = 16777215
114     Me.cboType.SetFocus
115 End Select
116
117 End Sub
118
119 Private Sub grpSearch_BeforeUpdate(Cancel As Integer)
120
121 End Sub
```

Code

```
1 Attribute VB_Name = "Form_frmPrisDetails"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdPreview_Click()
9 On Error GoTo Err_cmdPreview_Click
10
11     Dim stDocName As String
12
13     stDocName = "rptPrisonerDetails"
14     DoCmd.OpenReport stDocName, acPreview
15
16     DoCmd.Close acForm, "frmPrisDetails", acSaveNo
17
18 Exit_cmdPreview_Click:
19     Exit Sub
20
21 Err_cmdPreview_Click:
22     MsgBox Err.Description
23     Resume Exit_cmdPreview_Click
24
25 End Sub
26
27 Private Sub cmdSetDate_Click()
28 'set date equal to date on calender
29 txtDate = dwlndDate.Value
30
31 End Sub
32
33 Private Sub dwlndDate_Updated(Code As Integer)
34
35 End Sub
36
37 Private Sub Form_Current()
38
39 End Sub
40
41 Private Sub Form_Load()
42 'set controls equal to today
43 Me!dwnldDate = Date
44 Me!txtDate = Date
45
46 End Sub
```

Form: frmPrisDetails

```
47 Private Sub cmdPrint_Click()
48 On Error GoTo Err_cmdPrint_Click
49
50     Dim stDocName As String
51
52     stDocName = "rptPrisonerDetails"
53     DoCmd.OpenReport stDocName, acNormal
54
55 Exit_cmdPrint_Click:
56     Exit Sub
57
58 Err_cmdPrint_Click:
59     MsgBox Err.Description
60     Resume Exit_cmdPrint_Click
61
62 End Sub
63 Private Sub cmdCancel_Click()
64 On Error GoTo Err_cmdCancel_Click
65
66
67     DoCmd.Close
68
69 Exit_cmdCancel_Click:
70     Exit Sub
71
72 Err_cmdCancel_Click:
73     MsgBox Err.Description
74     Resume Exit_cmdCancel_Click
75
76 End Sub
```


Code

```

1 Attribute VB_Name = "Form_frmPTS"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdClose_Click()
9 On Error GoTo Err_cmdClose_Click
10
11
12     DoCmd.Close
13
14 Exit_cmdClose_Click:
15     Exit Sub
16
17 Err_cmdClose_Click:
18     MsgBox Err.Description
19     Resume Exit_cmdClose_Click
20
21 End Sub
22
23 Private Sub cmdFindUSMS_Click()
24 Dim rst As Recordset
25
26     Dim strCriteria As String
27     Dim strUSMS As String
28
29     'Make sure there is an entry in the text box before running
30     If IsNull(txtFindUSMS) Then
31         MsgBox "Enter a USMS numer first and then submit query"
32         Exit Sub
33     End If
34
35
36
37     Set rst = Me.RecordsetClone
38     strUSMS = Me!txtFindUSMS
39     strCriteria = "[pr_usms_no]='" & strUSMS & "'"
40     rst.FindFirst strCriteria
41     If rst.NoMatch Then
42         MsgBox "The number you entered was not found."
43     Else
44         Me.Bookmark = Me.RecordsetClone.Bookmark
45     End If
46     Me!txtFindUSMS.Value = ""
47
48 End Sub
49
50 Private Sub cmdNext_Click()
51 On Error GoTo Err_cmdNext_Click
52
53
54     DoCmd.GoToRecord , , acNext
55
56 Exit_cmdNext_Click:
57     Exit Sub
58
59 Err_cmdNext_Click:
60     MsgBox Err.Description
61     Resume Exit_cmdNext_Click
62
63 End Sub
64 Private Sub cmdPrevious_Click()

```

```
65 On Error GoTo Err_cmdPrevious_Click
66
67
68     DoCmd.GoToRecord , , acPrevious
69
70 Exit_cmdPrevious_Click:
71     Exit Sub
72
73 Err_cmdPrevious_Click:
74     MsgBox Err.Description
75     Resume Exit_cmdPrevious_Click
76
77 End Sub
78 Sub Combo33_AfterUpdate()
79
80     ' Find the record that matches the control.
81     If IsNull(Me![Combo33]) Then
82         Exit Sub
83     End If
84
85     ' Find the record that matches the control.
86     Me.RecordsetClone.FindFirst "[pr_key] = " & Me![Combo33]
87     Me.Bookmark = Me.RecordsetClone.Bookmark
88 End Sub
89
90 Private Sub Form_Current()
91
92     'synchronize combo box with record
93     Combo33 = pr_key
94 End Sub
95 Private Sub cmdPrintReport_Click()
96 On Error GoTo Err_cmdPrintReport_Click
97
98     Dim stDocName As String
99
100     stDocName = "rptPrisonerDetailsPTS"
101     DoCmd.OpenReport stDocName, acNormal
102
103 Exit_cmdPrintReport_Click:
104     Exit Sub
105
106 Err_cmdPrintReport_Click:
107     MsgBox Err.Description
108     Resume Exit_cmdPrintReport_Click
109
110 End Sub
```

Code

```

1 Attribute VB_Name = "Form_frmPTSArrests"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7 Const ConErrRptCanceled = 2501
8
9
10 Private Sub cmdCancel_Click()
11
12 ' Close form
13 On Error GoTo Err_cmdCancel_Click
14
15     DoCmd.Close
16
17 Exit_cmdCancel_Click:
18     Exit Sub
19
20 Err_cmdCancel_Click:
21     MsgBox Err.Description
22     Resume Exit_cmdCancel_Click
23
24 End Sub
25
26
27
28 Private Sub cmdPreview_Click()
29
30 ' Preview report.
31 Dim strDocName As String
32
33     On Error GoTo Err_cmdPreview_Click
34
35     ' Check to see that ending date is later than beginning date.
36     If IsDate(BeginDate) And IsDate(EndDate) Then
37         If EndDate < BeginDate Then
38             MsgBox "The end date must be later than the begin date."
39             cmdSetDate.Caption = "Set End Date"
40             SelectDate.SetFocus
41             Exit Sub
42         End If
43     Else
44         MsgBox "Please use a valid date for the beginning date and the ending date
45         values."
46     Exit Sub
47     End If
48
49     strDocName = "PTS Arrests"
50     DoCmd.OpenReport strDocName, acViewPreview
51
52 Exit_cmdPreview_Click:
53     Exit Sub
54
55 Err_cmdPreview_Click:
56     If Err = ConErrRptCanceled Then
57         Resume Exit_cmdPreview_Click
58     Else
59         MsgBox Err.Description
60         Resume Exit_cmdPreview_Click
61     End If
62

```

63 End Sub
Form: frmPTSArrests

```
64
65 Private Sub cmdPrint_Click()
66     ' Print report.
67     Dim strDocName As String
68
69     On Error GoTo Err_cmdPrint_Click
70
71     ' Check to see that ending date is later than beginning date.
72     If EndDate < BeginDate Then
73         MsgBox "The end date must be later than the begin date."
74         cmdSetDate.Caption = "Set End Date"
75         SelectDate.SetFocus
76         Exit Sub
77     End If
78
79     strDocName = "PTS Arrests"
80     DoCmd.OpenReport strDocName, acViewNormal
81
82 Exit_cmdPrint_Click:
83     Exit Sub
84
85 Err_cmdPrint_Click:
86     If Err = ConErrRptCanceled Then
87         Resume Exit_cmdPrint_Click
88     Else
89         MsgBox Err.Description
90         Resume Exit_cmdPrint_Click
91     End If
92
93 End Sub
94
95
96 Private Sub cmdSetDate_Click()
97
98     On Error GoTo cmdSetDate_Error
99
100    If cmdSetDate.Caption = "Set Begin Date" Then
101        BeginDate = SelectDate.Value
102        cmdSetDate.Caption = "Set End Date"
103    Else
104        EndDate = SelectDate.Value
105        cmdSetDate.Caption = "Set Begin Date"
106    End If
107
108    Exit Sub
109
110 cmdSetDate_Error:
111    MsgBox Err.Description
112    Exit Sub
113
114 End Sub
115
116 Function IsLoaded(strfrmName As String) As Integer
117
118     ' Determines if a form is loaded.
119
120     Const conFormDesign = 0
121     Dim intX As Integer
122
123     IsLoaded = False
124     For intX = 0 To Forms.Count - 1
125         If Forms(intX).FormName = strfrmName Then
126             If Forms(intX).CurrentView <> conFormDesign Then
127                 IsLoaded = True
128                 Exit Function ' Quit function once form has been found.
129             End If
130         End If

```

131 Next
132

Form: frmPTSArrests

```
133 End Function
134
135
136 Private Sub Detail_Click()
137
138 End Sub
139
140 Private Sub Form_Load()
141
142 'Enter parameters in date boxes
143 Me!SelectDate = Now()
144 Me!BeginDate = Now() - 7
145 Me!EndDate = Now()
146
147 End Sub
```

Code

```
1 Attribute VB_Name = "Form_frmREPORTS"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdOpenRptPtsDwn_Click()
9 On Error GoTo Err_cmdOpenRptPtsDwn_Click
10
11     Dim stDocName As String
12     Dim stLinkCriteria As String
13
14     stDocName = "PTSDownloads"
15     DoCmd.OpenForm stDocName, , , stLinkCriteria
16
17 Exit_cmdOpenRptPtsDwn_Click:
18     Exit Sub
19
20 Err_cmdOpenRptPtsDwn_Click:
21     MsgBox Err.Description
22     Resume Exit_cmdOpenRptPtsDwn_Click
23
24 End Sub
```

Code

```
1 Attribute VB_Name = "Form_frmTARGET_PHOTOSub"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdClose_Click()
9 On Error GoTo Err_cmdClose_Click
10
11
12     DoCmd.Close
13
14 Exit_cmdClose_Click:
15     Exit Sub
16
17 Err_cmdClose_Click:
18     MsgBox Err.Description
19     Resume Exit_cmdClose_Click
20
21 End Sub
22 Private Sub cmdPrint_Click()
23 On Error GoTo Err_cmdPrint_Click
24
25
26     DoCmd.PrintOut
27
28 Exit_cmdPrint_Click:
29     Exit Sub
30
31 Err_cmdPrint_Click:
32     MsgBox Err.Description
33     Resume Exit_cmdPrint_Click
34
35 End Sub
```

Code

```

1 Attribute VB_Name = "Form_frmTARGETS"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdPhoto_Click()
9 On Error GoTo Err_cmdPhoto_Click
10
11     Dim stDocName As String
12     Dim stLinkCriteria As String
13
14     stDocName = "frmTARGET_PHOTOsub"
15
16     stLinkCriteria = "[TargetKey]=" & Me![TargetKey]
17
18     DoCmd.OpenForm stDocName, , , stLinkCriteria
19
20 Exit_cmdPhoto_Click:
21     Exit Sub
22
23 Err_cmdPhoto_Click:
24     MsgBox Err.Description
25     Resume Exit_cmdPhoto_Click
26
27 End Sub
28
29 Private Sub Form_AfterUpdate()
30
31 'enable or disable photo buttons based on record status
32     If IsNull(Me![TargetKey]) And IsNull(Me![txtPhoto]) Then
33         Me!cmdPhoto.Enabled = False
34     Else
35         Me!cmdPhoto.Enabled = True
36     End If
37
38     If Not (IsNull(Me![TargetKey])) And Not (IsNull(Me![txtPhoto])) Then
39         Me!cmdInsertPhoto.Enabled = True
40     Else
41         Me!cmdInsertPhoto.Enabled = False
42     End If
43 'enable or disable photo buttons based on record status
44     If IsNull(Me![TargetKey]) And IsNull(Me![txtPhoto]) Then
45         Me!cmdPhoto.Enabled = False
46     Else
47         Me!cmdPhoto.Enabled = True
48     End If
49
50     If Not (IsNull(Me![TargetKey])) And Not (IsNull(Me![txtPhoto])) Then
51         Me!cmdInsertPhoto.Enabled = True
52     Else
53         Me!cmdInsertPhoto.Enabled = False
54     End If
55
56 End Sub
57
58 Private Sub Form_Current()
59
60 'enable or disable photo buttons based on record status
61     If IsNull(Me![TargetKey]) Or IsNull(Me![txtPhoto]) Then
62         Me!cmdPhoto.Enabled = False
63     Else
64         Me!cmdPhoto.Enabled = True

```



```
65     End If
66
67     If Not (IsNull(Me![TargetKey])) And (IsNull(Me![txtPhoto])) Then
68         Me!cmdInsertPhoto.Enabled = True
69     Else
70         Me!cmdInsertPhoto.Enabled = False
71     End If
72
73 End Sub
74
75 Private Sub Form_Load()
76     If IsNull(Me![TargetKey]) Then
77         Me!cmdPhoto.Enabled = False
78     Else
79         Me!cmdPhoto.Enabled = True
80     End If
81
82 End Sub
83 Private Sub cmdInsertPhoto_Click()
84 On Error GoTo Err_cmdInsertPhoto_Click
85
86     Dim stDocName As String
87     Dim stLinkCriteria As String
88
89     stDocName = "frmTARGET_PHOTOsub"
90
91     stLinkCriteria = "[TargetKey]=" & Me![TargetKey]
92     DoCmd.OpenForm stDocName, , , stLinkCriteria
93
94 Exit_cmdInsertPhoto_Click:
95     Exit Sub
96
97 Err_cmdInsertPhoto_Click:
98     MsgBox Err.Description
99     Resume Exit_cmdInsertPhoto_Click
100
101 End Sub
```

Form: HOMICIDESubform

Code

```
1 Attribute VB_Name = "Form_HOMICIDESubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Form_Open(Cancel As Integer)
9
10 'On opening set focus to Homicide date
11 Me!HomDate.SetFocus
12
13 End Sub
```

Form: LOCALEsubform

Code

```
1 Attribute VB_Name = "Form_LOCALEsubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Form_Load()
9
10 Me!LocName.SetFocus
11 Me.Repaint
12
13 End Sub
```

Code

```
1 Attribute VB_Name = "Form_NARCOTICSsubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredID = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Form_Current()
9
10 End Sub
11
12 Private Sub NarcType_AfterUpdate()
13
14 If IsNull(Me!NarcType) Then
15     Forms!COOPERATORS!subNARCOTICS!subNARC_BRAND!Brand.Enabled = False
16 Else
17     Forms!COOPERATORS!subNARCOTICS!subNARC_BRAND!Brand.Enabled = True
18 End If
19
20 End Sub
21
22 Private Sub NarcType_BeforeUpdate(Cancel As Integer)
23
24 End Sub
```

Code

```

1 Attribute VB_Name = "Form_PTSDownloads"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7 Const ConErrRptCanceled = 2501
8
9
10 Private Sub cmdCancel_Click()
11
12 ' Close form
13 On Error GoTo Err_cmdCancel_Click
14
15     DoCmd.Close
16
17 Exit_cmdCancel_Click:
18     Exit Sub
19
20 Err_cmdCancel_Click:
21     MsgBox Err.Description
22     Resume Exit_cmdCancel_Click
23
24 End Sub
25
26
27
28 Private Sub cmdPreview_Click()
29
30 ' Preview report.
31 Dim strDocName As String
32
33     On Error GoTo Err_cmdPreview_Click
34
35     ' Check to see that ending date is later than beginning date.
36     If IsDate(BeginDate) And IsDate(EndDate) Then
37         If EndDate < BeginDate Then
38             MsgBox "The end date must be later than the begin date."
39             cmdSetDate.Caption = "Set End Date"
40             SelectDate.SetFocus
41             Exit Sub
42         End If
43     Else
44         MsgBox "Please use a valid date for the beginning date and the ending date
45         values."
46     Exit Sub
47     End If
48
49     strDocName = "PTS Downloads"
50     DoCmd.OpenReport strDocName, acViewPreview
51
52 Exit_cmdPreview_Click:
53     Exit Sub
54
55 Err_cmdPreview_Click:
56     If Err = ConErrRptCanceled Then
57         Resume Exit_cmdPreview_Click
58     Else
59         MsgBox Err.Description
60         Resume Exit_cmdPreview_Click
61     End If
62

```

63 End Sub
Form: PTSDownloads

```
64
65 Private Sub cmdPrint_Click()
66     ' Print report.
67     Dim strDocName As String
68
69     On Error GoTo Err_cmdPrint_Click
70
71     ' Check to see that ending date is later than beginning date.
72     If EndDate < BeginDate Then
73         MsgBox "The end date must be later than the begin date."
74         cmdSetDate.Caption = "Set End Date"
75         SelectDate.SetFocus
76         Exit Sub
77     End If
78
79     strDocName = "PTS Downloads"
80     DoCmd.OpenReport strDocName, acViewNormal
81
82 Exit_cmdPrint_Click:
83     Exit Sub
84
85 Err_cmdPrint_Click:
86     If Err = ConErrRptCanceled Then
87         Resume Exit_cmdPrint_Click
88     Else
89         MsgBox Err.Description
90         Resume Exit_cmdPrint_Click
91     End If
92
93 End Sub
94
95
96 Private Sub cmdSetDate_Click()
97
98     On Error GoTo cmdSetDate_Error
99
100    If cmdSetDate.Caption = "Set Begin Date" Then
101        BeginDate = SelectDate.Value
102        cmdSetDate.Caption = "Set End Date"
103    Else
104        EndDate = SelectDate.Value
105        cmdSetDate.Caption = "Set Begin Date"
106    End If
107
108    Exit Sub
109
110 cmdSetDate_Error:
111    MsgBox Err.Description
112    Exit Sub
113
114 End Sub
115
116 Function IsLoaded(strfrmName As String) As Integer
117
118     ' Determines if a form is loaded.
119
120     Const conFormDesign = 0
121     Dim intX As Integer
122
123     IsLoaded = False
124     For intX = 0 To Forms.Count - 1
125         If Forms(intX).FormName = strfrmName Then
126             If Forms(intX).CurrentView <> conFormDesign Then
127                 IsLoaded = True
128                 Exit Function ' Quit function once form has been found.
129             End If
130         End If
```

131 Next
132
Form: PTSDownloads

```
133 End Function
134
135
136 Private Sub Detail_Click()
137
138 End Sub
139
140 Private Sub Form_Load()
141
142 'Enter parameters in date boxes
143 Me!SelectDate = Now()
144 Me!BeginDate = Now() - 7
145 Me!EndDate = Now()
146
147 End Sub
```

Code

```
1 Attribute VB_Name = "Form_StartUp"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub cmdOpenAlias_Click()
9 On Error GoTo Err_cmdOpenAlias_Click
10
11     Dim stDocName As String
12     Dim stLinkCriteria As String
13
14     stDocName = "frmFindAlias"
15     DoCmd.OpenForm stDocName, , , stLinkCriteria
16     DoCmd.Close acForm, "StartUp"
17
18 Exit_cmdOpenAlias_Click:
19     Exit Sub
20
21 Err_cmdOpenAlias_Click:
22     MsgBox Err.Description
23     Resume Exit_cmdOpenAlias_Click
24
25 End Sub
26
27 Private Sub cmdOpenAssoc_Click()
28 On Error GoTo Err_cmdOpenAssoc_Click
29
30     Dim stDocName As String
31     Dim stLinkCriteria As String
32
33     stDocName = "frmFindAssoc"
34     DoCmd.OpenForm stDocName, , , stLinkCriteria
35     DoCmd.Close acForm, "StartUp"
36
37 Exit_cmdOpenAssoc_Click:
38     Exit Sub
39
40 Err_cmdOpenAssoc_Click:
41     MsgBox Err.Description
42     Resume Exit_cmdOpenAssoc_Click
43
44 End Sub
45
46 Private Sub cmdOpenCoopFrm_Click()
47 On Error GoTo Err_cmdOpenCoopFrm_Click
48
49     Dim stDocName As String
50     Dim stLinkCriteria As String
51
52     stDocName = "COOPERATORS"
53     DoCmd.OpenForm stDocName, , , stLinkCriteria
54     DoCmd.Close acForm, "StartUp"
55
56 Exit_cmdOpenCoopFrm_Click:
57     Exit Sub
58
59 Err_cmdOpenCoopFrm_Click:
60     MsgBox Err.Description
61     Resume Exit_cmdOpenCoopFrm_Click
62
63
64 End Sub
```



```
65
66 Private Sub cmdOpenCounsel_Click()
67 On Error GoTo Err_cmdOpenCounsel_Click
68
69     Dim stDocName As String
70     Dim stLinkCriteria As String
71
72     stDocName = "frmFindCounsel"
73     DoCmd.OpenForm stDocName, , , stLinkCriteria
74     DoCmd.Close acForm, "StartUp"
75
76 Exit_cmdOpenCounsel_Click:
77     Exit Sub
78
79 Err_cmdOpenCounsel_Click:
80     MsgBox Err.Description
81     Resume Exit_cmdOpenCounsel_Click
82
83 End Sub
84
85 Private Sub cmdOpenDefFrm_Click()
86 On Error GoTo Err_cmdOpenDefFrm_Click
87
88     Dim stDocName As String
89     Dim stLinkCriteria As String
90
91     stDocName = "DEFENDANT"
92     DoCmd.OpenForm stDocName, , , stLinkCriteria
93     DoCmd.Close acForm, "StartUp"
94
95 Exit_cmdOpenDefFrm_Click:
96     Exit Sub
97
98 Err_cmdOpenDefFrm_Click:
99     MsgBox Err.Description
100    Resume Exit_cmdOpenDefFrm_Click
101
102
103 End Sub
104
105 Private Sub cmdOpenFindEmploy_Click()
106
107 End Sub
108
109 Private Sub cmdOpenMaint_Click()
110
111 Me.GoToPage 4
112
113 End Sub
114
115 Private Sub cmdOpenNarc_Click()
116 On Error GoTo Err_cmdOpenNarc_Click
117
118     Dim stDocName As String
119     Dim stLinkCriteria As String
120
121     stDocName = "frmFindNarcs"
122     DoCmd.OpenForm stDocName, , , stLinkCriteria
123     DoCmd.Close acForm, "StartUp"
124
125 Exit_cmdOpenNarc_Click:
126     Exit Sub
127
128 Err_cmdOpenNarc_Click:
129     MsgBox Err.Description
130     Resume Exit_cmdOpenNarc_Click
131
132 End Sub
```

```

134 Private Sub cmdOpenPg3_Click()
135
136 Me.GoToPage 3
137
138 End Sub
139
140 Private Sub cmdOpenRptfrm_Click()
141 Me.GoToPage 2
142 End Sub
143
144 Private Sub cmdOpenTqfrm_Click()
145
146 If Me!lblMain.Visible = True Then
147     Me.lblMain.Visible = False
148     Me.lblRpt.Visible = False
149     Me.lblSQ.Visible = False
150     Me.lblTQ.Visible = False
151     Me.lblRecords.Visible = False
152     Me.lblMaint.Visible = False
153     Me.cmdOpenMaint.Visible = False
154     Me.cmdOpenPg3.Visible = False
155     Me.cmdOpenSqfrm.Visible = False
156     Me.cmdOpenRptfrm.Visible = False
157     Me.lblAlias.Visible = True
158     Me.lblNarc.Visible = True
159     Me.lblAssoc.Visible = True
160     Me.lblCounsel.Visible = True
161     Me.lblEmploy.Visible = True
162     Me.lblReturnMain.Visible = True
163     Me.cmdOpenAlias.Visible = True
164     Me.cmdOpenNarc.Visible = True
165     Me.cmdOpenAssoc.Visible = True
166     Me.cmdOpenCounsel.Visible = True
167     Me.cmdOpenEmploy.Visible = True
168     Me.cmdOpenAlias.SetFocus
169     Me.cmdOpenTqfrm.Picture = "c:\Program
Files\DevStudio\VB\Graphics\Icons\Arrows\Arw09lt.ico"
170
171 ElseIf Me!lblMain.Visible = False Then
172     Me.lblMain.Visible = True
173     Me.lblRpt.Visible = True
174     Me.lblSQ.Visible = True
175     Me.lblTQ.Visible = True
176     Me.lblRecords.Visible = True
177     Me.cmdOpenPg3.Visible = True
178     Me.cmdOpenSqfrm.Visible = True
179     Me.cmdOpenRptfrm.Visible = True
180     Me.lblMaint.Visible = True
181     Me.cmdOpenMaint.Visible = True
182     Me.lblAlias.Visible = False
183     Me.lblNarc.Visible = False
184     Me.lblAssoc.Visible = False
185     Me.lblCounsel.Visible = False
186     Me.lblEmploy.Visible = False
187     Me.lblReturnMain.Visible = False
188     Me.cmdOpenAlias.Visible = False
189     Me.cmdOpenNarc.Visible = False
190     Me.cmdOpenAssoc.Visible = False
191     Me.cmdOpenCounsel.Visible = False
192     Me.cmdOpenEmploy.Visible = False
193     Me.cmdOpenTqfrm.Picture = "C:\Program
Files\DevStudio\VB\Graphics\Icons\Misc\LIGHTON.ICO"
194 End If
195 End Sub
196
197 Private Sub Command30_Click()

```

```
198 Me.GoToPage 1
199 End Sub
Form: StartUp
```

```
200
201 Private Sub Command34_Click()
202
203 End Sub
204
205 Private Sub Command40_Click()
206
207 Me.GoToPage 1
208
209 End Sub
210
211 Private Sub Command82_Click()
212
213 'run transfer.exe - vb app that inputs email messages into database
214 DimRetVal
215 RetVal = Shell("D:\NETWORK\RACKETS\MISC\TRANSLATE.EXE", 1)
216
217 DoCmd.Close acForm, "StartUp"
218
219 End Sub
220
221 Private Sub Form_Load()
222 Me.GoToPage 1
223
224 Me.lblAlias.Visible = False
225 Me.lblNarc.Visible = False
226 Me.lblAssoc.Visible = False
227 Me.lblCounsel.Visible = False
228 Me.lblEmploy.Visible = False
229 Me.lblReturnMain.Visible = False
230 Me.cmdOpenAlias.Visible = False
231 Me.cmdOpenNarc.Visible = False
232 Me.cmdOpenAssoc.Visible = False
233 Me.cmdOpenCounsel.Visible = False
234 Me.cmdOpenEmploy.Visible = False
235 Me.lblMain.Visible = True
236 Me.lblRpt.Visible = True
237 Me.lblSQ.Visible = True
238 Me.lblTQ.Visible = True
239 Me.lblRecords.Visible = True
240 Me.lblMaint.Visible = True
241 Me.cmdOpenMaint.Visible = True
242 Me.cmdOpenPg3.Visible = True
243 Me.cmdOpenTQfrm.Visible = True
244 Me.cmdOpenSQfrm.Visible = True
245 Me.cmdOpenRptfrm.Visible = True
246
247 End Sub
248
249 Private Sub Form_Open(Cancel As Integer)
250
251 Me.GoToPage 1
252
253 Me.lblAlias.Visible = False
254 Me.lblNarc.Visible = False
255 Me.lblAssoc.Visible = False
256 Me.lblCounsel.Visible = False
257 Me.lblEmploy.Visible = False
258 Me.lblReturnMain.Visible = False
259 Me.cmdOpenAlias.Visible = False
260 Me.cmdOpenNarc.Visible = False
261 Me.cmdOpenAssoc.Visible = False
262 Me.cmdOpenCounsel.Visible = False
263 Me.cmdOpenEmploy.Visible = False
264 Me.lblMain.Visible = True
265 Me.lblRpt.Visible = True
```

```
266 Me.lblSQ.Visible = True
267 Me.lblTQ.Visible = True
268 Me.lblMaint.Visible = True
```

Form: StartUp

```
269 Me.cmdOpenMaint.Visible = True
270 Me.lblRecords.Visible = True
271 Me.cmdOpenPg3.Visible = True
272 Me.cmdOpenTQfrm.Visible = True
273 Me.cmdOpenSQfrm.Visible = True
274 Me.cmdOpenRptfrm.Visible = True
275
276 End Sub
277 Private Sub cmdExit_Click()
278 On Error GoTo Err_cmdExit_Click
279
280
281     DoCmd.Quit
282
283 Exit_cmdExit_Click:
284     Exit Sub
285
286 Err_cmdExit_Click:
287     MsgBox Err.Description
288     Resume Exit_cmdExit_Click
289
290 End Sub
291 Private Sub cmdOpenCaseFrm_Click()
292 On Error GoTo Err_cmdOpenCaseFrm_Click
293
294     Dim stDocName As String
295     Dim stLinkCriteria As String
296
297     stDocName = "CASE"
298     DoCmd.OpenForm stDocName, , , stLinkCriteria
299     DoCmd.Close acForm, "StartUp"
300
301 Exit_cmdOpenCaseFrm_Click:
302     Exit Sub
303
304 Err_cmdOpenCaseFrm_Click:
305     MsgBox Err.Description
306     Resume Exit_cmdOpenCaseFrm_Click
307
308 End Sub
309 Private Sub cmdOpenNewCase_Click()
310 On Error GoTo Err_cmdOpenNewCase_Click
311
312     Dim stDocName As String
313     Dim stLinkCriteria As String
314
315     stDocName = "CASE"
316     DoCmd.OpenForm stDocName, , , stLinkCriteria
317     DoCmd.Close acForm, "StartUp"
318
319     Forms!CASE.DataEntry = True
320     Forms!CASE.cmdArchive.Enabled = False
321     Forms!CASE.cmdArchive.Visible = False
322     Forms!CASE.Combo15.Enabled = False
323     Forms!CASE.Combo15.Visible = False
324     Forms!CASE.lblUSV.Visible = False
325
326 Exit_cmdOpenNewCase_Click:
327     Exit Sub
328
329 Err_cmdOpenNewCase_Click:
330     MsgBox Err.Description
331     Resume Exit_cmdOpenNewCase_Click
332
333 End Sub
```

```

334 Private Sub cmdOpenPTS_Click()
335 On Error GoTo Err_cmdOpenPTS_Click
336
337     Dim stDocName As String
Form: StartUp

338     Dim stLinkCriteria As String
339
340     stDocName = "frmPTS"
341     DoCmd.Close acForm, "StartUp"
342     DoCmd.OpenForm stDocName, , , stLinkCriteria
343
344 Exit_cmdOpenPTS_Click:
345     Exit Sub
346
347 Err_cmdOpenPTS_Click:
348     MsgBox Err.Description
349     Resume Exit_cmdOpenPTS_Click
350
351 End Sub
352 Private Sub cmdOpenEmploy_Click()
353 On Error GoTo Err_cmdOpenEmploy_Click
354
355     Dim stDocName As String
356     Dim stLinkCriteria As String
357
358     stDocName = "frmFindEmployer"
359     DoCmd.OpenForm stDocName, , , stLinkCriteria
360     DoCmd.Close acForm, "StartUp"
361
362 Exit_cmdOpenEmploy_Click:
363     Exit Sub
364
365 Err_cmdOpenEmploy_Click:
366     MsgBox Err.Description
367     Resume Exit_cmdOpenEmploy_Click
368
369 End Sub
370 Private Sub cmdOpenclnArrest_Click()
371 On Error GoTo Err_cmdOpenclnArrest_Click
372
373     Dim stDocName As String
374     Dim stLinkCriteria As String
375
376     stDocName = "cln_ARREST"
377     DoCmd.OpenForm stDocName, , , stLinkCriteria
378
379 Exit_cmdOpenclnArrest_Click:
380     Exit Sub
381
382 Err_cmdOpenclnArrest_Click:
383     MsgBox Err.Description
384     Resume Exit_cmdOpenclnArrest_Click
385
386 End Sub
387 Private Sub cmdOpenPTSarr_Click()
388 End Sub

```

Code

```
1 Attribute VB_Name = "Form_STATUTESsubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Count_BeforeUpdate(Cancel As Integer)
9
10 End Sub
11
12 Private Sub Form_AfterUpdate()
13 End Sub
14
15 Private Sub Form_BeforeUpdate(Cancel As Integer)
16
17
18 End Sub
19
20 Private Sub Form_Current()
21
22
23 End Sub
24
25 Private Sub Form_Error(DataErr As Integer, Response As Integer)
26
27     Const conErrFieldRequired = 3314
28
29     If DataErr = conErrFieldRequired Then
30         DisplayMessage "You must enter a Count and Statute number to save the record."
31
32         Response = acDataErrContinue
33     Else
34         Response = acDataErrDisplay
35     End If
36 End Sub
37
38 Private Sub Form_Load()
39
40 End Sub
41
42 Private Sub Location_AfterUpdate()
43
44
45 End Sub
46
47 Private Sub Location_Click()
48
49
50 End Sub
51
52 Private Sub StatuteDesc_AfterUpdate()
53
54
55 End Sub
```

PROPERTY OF
National Criminal Justice Reference Service (NCJRS)
Box 6000
Washington, DC 20540-6000

Form: WARRANTsubform

Code

```
1 Attribute VB_Name = "Form_WARRANTsubform"  
2 Attribute VB_Creatable = True  
3 Attribute VB_PredeclaredId = True  
4 Attribute VB_Exposed = False  
5 Option Compare Database  
6 Option Explicit  
7  
8 Private Sub Form_Open(Cancel As Integer)  
9  
10  
11 End Sub
```

Form: WIRETAPsubform

Code

```
1 Attribute VB_Name = "Form_WIRETAPsubform"
2 Attribute VB_Creatable = True
3 Attribute VB_PredeclaredId = True
4 Attribute VB_Exposed = False
5 Option Compare Database
6 Option Explicit
7
8 Private Sub Form_Open(Cancel As Integer)
9
10
11
12 End Sub
```

PROPERTY OF
National Criminal Justice Reference Service (NCJRS)
1997-2000
1997-2000 MD 20944-6000