

Direct Access of CAIRS Data Using Microsoft Query



Table of Contents

Introduction	1
Purpose of the Course	1
Objectives	1
Conventions and Icons	2
What is CAIRS Direct Access?	3
Registration	4
System and Software Requirements	5
Direct Access Installation	6
Using the Install Program	6
Structured Query Language	7
What is Structured Query Language (SQL)?	7
SQL Elements	8
Statements, Commands and Clauses	8
Data Types	8
Using the SELECT Statement	9
Selecting All of a Table's Records	9
Selecting Some of a Table's Records	10
Grouping Selected Data	16
Ordering Selected Data	19
Joining Tables	21
CAIRS Data Structure	26
Microsoft Query	27
Creating CAIRS as a Data Source	29
The Microsoft Query Window	33
The Microsoft Query Window Panes	33
The Microsoft Query Menu Bar	34
The Microsoft Query Tool Bar	34
Specifying the Criteria	37
Adding Report Columns	41
Saving the Query	43
Viewing the SQL SELECT Statement	44
Setting Query Properties	45
Sorting Records	48
Adding Calculated Columns	52
Adding and Joining Tables	55

Placing Restrictions on Groups	63
Setting Query Parameters	66
Modifying a Query's Design	70
Opening a Query	70
Modifying Table Information	72
Modifying Criteria Information	74
Modifying Report Columns	76
Displaying and Formatting the Result Set	80
Displaying Report Columns	80
Formatting the Report	80
CAIRS Direct Access Templates	83
The Templates	84
Executing a Template from Microsoft Excel	85
Appendix A - TEST Database	A-1
Appendix B - Database Schema and Cross Reference	A-1
Appendix C - Sample Template Reports	C-1
Exercise Solutions	S-1

Introduction

Purpose of the Course

In this section you will be introduced to the direct access feature of the Computerized Accident/Incident Reporting System (CAIRS). We will discuss the registration process, security issues, and how to setup your PC for CAIRS direct access with Microsoft Query. In addition, you will learn the basics of Structured Query Language (SQL) and how to use Microsoft Query to directly access CAIRS. Finally, you will be provided with the CAIRS database schema and CAIRS direct access templates and given instructions on how to use these tools.

Objectives

Upon completion of this class, you will be able to perform the following activities:

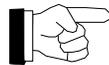
- ▶ State the purpose of CAIRS direct access
- ▶ Describe the registration procedure for obtaining CAIRS direct access
- ▶ Describe the setup procedure for CAIRS direct access with Microsoft Query
- ▶ Describe the basic elements of Structured Query Language (SQL)
- ▶ Use the SQL Select statement to perform a database query
- ▶ Show how to join database tables
- ▶ Describe the CAIRS database structure
- ▶ Describe the Microsoft Query menus and toolbar
- ▶ Link to CAIRS with Microsoft Query
- ▶ Create a new query
- ▶ Save a query
- ▶ Modify a query
- ▶ Execute an SQL template from Microsoft Excel

Conventions and Icons

Throughout this guide, certain conventions and icons are used to help you easily locate and interpret information.

- ▶ When you need to type information or select text or an item from a list or menu, the prompt will appear boldface; for example, **something you must type or select**.
- ▶ Names of menus, icons, buttons, selection boxes, files, directories, disk drives, and windows will be shown in bold uppercase; for example, **THIS SPECIAL TYPEFACE.**)
- ▶ References used to direct you to additional information (often to another section in the guide will be shown in bold italic; for example, ***this special typeface***.
- ▶ The following special symbols are used to call your attention to important or special information applicable to the current topic:

NOTE



This icon appears when we are providing you with *additional information* for completing a task or using a feature.

HINT



This icon indicates a *hint, good tip, or shortcut* that you can use when completing a task.



WARNING

This icon is displayed when we are providing you with *critical information* or to warn you about actions that you *must do* or *must not do*.



TUTORIAL

This icon indicates we are about to give you *step-by-step procedures* to follow to accomplish a task.



TERMINOLOGY

This icon flags any new terminology that is important for you to know and understand.

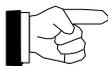
What is CAIRS Direct Access?

A recent survey of *advanced* CAIRS users indicated they needed the capability to perform more complex searches of the CAIRS database than what was provided by the CAIRS Graphical User Interface (GUI). CAIRS Direct Access is the *new tool* that provides this capability and offers advanced users direct SQL access to CAIRS database.

Various software packages (many of which you may already be familiar with -- e.g. Microsoft Excel) may be used to directly access the CAIRS database and obtain needed information. Note that *any* software that supports Open DataBase Connectivity (ODBC) may be used to directly access CAIRS data. For example, you may select Microsoft Access, Excel, FoxPro or any other software package that supports ODBC. The software you choose to use will vary based on your needs and experience. *You will be responsible for providing your own software for direct access.* In most cases, the software you will use already resides on your computer and you may already be proficient in its use.

The CAIRS database resides on a SQL server. If you are currently accessing other SQL servers (or remote ODBC databases in other environments) you will find that the process for setting up and establishing a connection to the CAIRS database is very similar (see the ***Direct Access Installation*** section of this manual).

NOTE



DOE will provide the necessary software drivers for establishing connectivity to CAIRS and will provide guidance on setting up and establishing a connection to the database.

The CAIRS database structure, as well as other information you will need to query CAIRS, is found in this guide. In addition, DOE is currently in the process of developing various templates that will assist you in creating your own queries of the CAIRS database. You may use these templates directly from Microsoft Query, or you may execute them from within Microsoft Excel to create various reports. These templates are very easy to modify, which allows you the capability to customized them by adding new fields or search criteria. More information about these templates may be found in the ***CAIRS Direct Access Templates*** section of this manual.

NOTE



This guide instructs you on how to directly access the CAIRS database using Microsoft Query 8.0 (a Microsoft Office 97 add-in) and makes references to Microsoft Query 2.0 (a Microsoft Office 95 add-in). Although any ODBC compliant software may be used to directly access CAIRS data, use of other software is outside the scope of this guide.

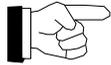
Registration

System and Software Requirements

System requirements for CAIRS direct access are listed below:

- You must have either a network or modem connection to the Internet.
- You must have software that will support Open Database Connectivity (ODBC), such as Microsoft Query.
- PC users must have a 486 or faster CPU with sufficient memory to support SQL server client software and Microsoft applications such as Excel, Access, and Query (or other software that supports ODBC).
- You must run on a platform that will support ODBC and SQL server networking products (e.g., Windows 95 or NT).
- Network services require SQL server access (SQLServer1) through port 1434.
- Your local network logon (both user name and password) *must match exactly* the CAIRS GUI logon.

NOTE



To change your CAIRS GUI password click on the **CAIRS Database Modules** hyperlink on the **CAIRS** home page (<https://cairs.tis.eh.doe.gov/cairs/cairs.asp>) and then click on the **User Tools** hyperlink.

Direct Access Installation

To directly access CAIRS data, ODBC drivers must be installed and configured on your PC. The TIS Helpline (800-473-4375) can provide you with an installation CD for Windows 95, Windows 98, or NT. If you are already a registered CAIRS user with a valid user ID and password, you may download the install program directly from the Internet at the following URL:

https://cairs.tis.eh.doe.gov/cairs/help/help/cairs_h.htm.

Once you've entered your CAIRS user ID and password, click on the **Direct Access** hyperlink and then the **Install** hyperlink to download the install file (named **CairsR1.exe**) to your hard drive. You will need approximately 5 Mg of hard drive space to install the drivers and client utilities.

Using the Install Program

Follow these steps to install and configure the drivers and utilities from Windows 95 or Windows 98. Note that the steps for NT will be similar:

1. Select **Run** from the Windows **START** menu.
2. In the **OPEN** selection box, type the path **c:\...\CairsR1.exe** (if installing from your hard drive) or **d:\CairsR1.exe** (if installing from CD). (If you are not certain what the path is on your hard drive to the file '**CairsR1.exe**', click on the **BROWSE** command button and locate the file.)
3. Click on the **OK** command button in the **RUN** dialog box.
4. The **CAIRS Direct SQL Access Setup** program will launch. Follow the prompts to set up direct access.

NOTE



Once your installation and setup are complete, you may delete the **CairsR1.exe** installation file to free up hard drive space.

Structured Query Language

What is Structured Query Language (SQL)?

Structured Query Language or SQL (pronounced “sequel”) is the most widely used tool for communicating with a relational database, such as CAIRS. A relational database consists of a collection of tables. A table is made of up rows (called records) and columns (called fields). Complex instructions are encoded on the client and then transmitted over a modem or network to the server where the instructions are decoded and then executed. SQL allows a user to retrieve information, create tables, delete tables, grant or revoke access to tables, and perform other tasks for full use of a database. As an end-user of CAIRS, you will only be permitted to retrieve information from CAIRS. When retrieving information from CAIRS with SQL, a *virtual table* (a set of rows and columns) is always returned.



This section provides high-level general information about SQL. The information may be sufficient if you plan to directly access CAIRS information using only pre-defined templates; however, if you plan on creating your own queries to retrieve information from CAIRS, we strongly recommended that you receive additional training and obtain supplementary reference materials.

SQL Elements

Statements, Commands and Clauses

SQL statements, commands, and clauses formulate the instructions that are sent to the CAIRS server. The SELECT statement allows you to examine the contents of CAIRS. Modifying clauses further refine the data specified by a SELECT statement. Modifying clauses are described in the table below.

Modifying Clauses and Functions	Description
FROM	Specifies which table to take data from
WHERE	Filters out rows (records) that do not satisfy a search condition.
GROUP BY	Organizes the selected data into groups
HAVING	Filters out groups that do not satisfy the search condition
ORDER BY	Sorts the results of prior clauses to produce final output

Data Types

In every table, each column stores only a certain type of data. These types of data are stored in the CAIRS database tables:

- ▶ CHARACTER: columns of this type contain strings of characters.
- ▶ NUMERIC: columns of this type contain numbers that may contain a fractional component.

Using the SELECT Statement

In order to use Microsoft Query to directly access CAIRS data, you must be able to interpret the SELECT statement and the modifying clauses which act on the selection. To learn about the SELECT statement, we will use the example database, called **TEST**, in **Appendix A**. **TEST** consists of three tables: **ACCIDENT**, **ORGANIZATION**, **INJURY** and **BODY**. To facilitate the exercise for this section, the table structures are very simple. The **ACCIDENT** table contains 22 records and seven columns. The **NUMBER**, **WDLR**, and **WDL** columns are the numeric data type; the remaining columns are character data types. The **ORGANIZATION** table contains three records and two columns, and the **INJURY** and **BODY** tables contains eight records and two columns. All columns for the **ORGANIZATION**, **INJURY**, and **BODY** tables are character data types.

Selecting All of a Table's Records

The syntax for the simplest SELECT statements uses only the FROM modifier. The general form of this SELECT statement is:

```
SELECT <column names>  
FROM <table>
```

For example, to retrieve all of the rows and all of the columns from the **ACCIDENT** table in **TEST**, you would use this statement:

```
SELECT *  
FROM TEST.ACCIDENT
```

NOTE



The asterisk (*) is a wildcard meaning everything.

Similarly, to retrieve only the **ORG** column from the **ACCIDENT** table in **TEST**, you would use this statement:

```
SELECT ACCIDENT.ORG  
FROM TEST.ACCIDENT
```

Selecting Some of a Table's Records

If you want to see only some of the records in a table, you must use the WHERE modifier. The general form of this SELECT statement is:

```
SELECT <column names>  
FROM <table>  
WHERE <predicate>
```

The table below displays the operators that can be used within a predicate when using Microsoft Query to retrieve information from CAIRS.

Operator	Predicate Syntax	Description
equals	<column name> = <value>	This operator selects only records where the value in the specified column equals <value>.
does not equal	<column name> <> <value>	This operator selects only records where the value in the specified column is not equal to <value>.
is greater than	<column name> > <value>	This operator selects only records where the value in the specified column is greater than <value>.
is greater than or equal to	<column name> >= <value>	This operator selects only records where the value in the specified column is greater than or equal to <value>.
is less than	<column name> < <value>	This operator selects only records where the value in the specified column is less than <value>.
is less than or equal to	<column name> <= <value>	This operator selects only records where the value in the specified column is less than or equal to <value>.

Operator	Predicate Syntax	Description
is one of	<i><column name> IN <values></i>	<p><values> consist of one or more values, separated by commas and enclosed in parentheses.</p> <p>This operator selects only records where the value in the specified column is found in <values>.</p>
is not one of	<i><column name> NOT IN <values></i>	<p><values> consist of one or more values, separated by commas and enclosed in parentheses.</p> <p>This operator selects only records where the value in the specified column is not found in <values>.</p>
is between	<i><column name> BETWEEN <value1> AND <value2></i>	This operator selects only records where the value in the specified column is greater than or equal to <value1> and less than or equal to <value2>.
is not between	<i><column name> NOT BETWEEN <value1> AND <value2></i>	This operator selects only records where the value in the specified column is not in the specified range.
like	<i><column name> LIKE <value></i>	<p>This operator can only be used with character columns.</p> <p>This operator selects only records where the value in the specified column equals <value>.</p>
not like	<i><column name> NOT LIKE <value></i>	<p>This operator can only be used with character columns.</p> <p>This operator selects only records where the value in the specified column is not equal to <value>.</p>

Operator	Predicate Syntax	Description
begins with	<i><column name> LIKE <value>%</i>	<p>This operator can only be used with character columns.</p> <p>This operator selects only records where the value in the specified column begins with <value>.</p>
does not begin with	<i><column name> NOT LIKE <value>%</i>	<p>This operator can only be used with character columns.</p> <p>This operator selects only records where the value in the specified column does not begin with <value>.</p>
ends with	<i><column name> LIKE %<value></i>	<p>This operator can only be used with character columns.</p> <p>This operator selects only records where the value in the specified column ends with <value>.</p>
does not end with	<i><column name> NOT LIKE %<value></i>	<p>This operator can only be used with character columns.</p> <p>This operator selects only records where the value in the specified column does not end with <value>.</p>
contains	<i><column name> LIKE %<value>%</i>	<p>This operator can only be used with character columns.</p> <p>This operator selects only records where the value in the specified column contains <value>.</p>
does not contain	<i><column name> NOT LIKE %<value>%</i>	<p>This operator can only be used with character columns.</p> <p>This operator selects only records where the value in the specified column does not contain <value>.</p>

Operator	Predicate Syntax	Description
is Null	<i><column name> IS NULL</i>	This operator selects only records where the value in the specified column contains the value NULL.
is Not Null	<i><column name> NOT NULL</i>	This operator selects only records where the value in the specified column does not contain the value NULL.



The values cited in the predicate must be the same data types. In addition, any specified character value must be enclosed in single quotes.



In SQL, the percent sign (%) is a wildcard that stands for any string of characters, zero or more in length.



NULL represents the lack of a value in a record. It is **NOT** the numeric value 0 or a blank.

For example, to list all of the records for organization **0502009**, you would use this statement:

```
SELECT *
  FROM TEST.ACCIDENT
 WHERE ACCIDENT.ORG = '0502009'
```

The result of this SELECT statement is displayed in the table below.

NUMBER	DATE (yyyymmdd)	ORG	INJURY_ TYPE	BODY_ PART	WDLR	WDL
1	19980112	0502009	0012	3101	20	10
2	19980115	0502009	0052	3101	0	0
10	19980409	0502009	0035	4102	2	2
13	19980506	0502009	0062	0101	0	256
16	19980619	0502009	0035	3101	0	0
17	19980704	0502009	0037	3101	0	0
22	19981030	0502009	0037	4401	0	1

Exercise 1

1. Write the SQL SELECT statement that selects all records with a date after April 15, 1998 and before May 16, 1998. Show all columns on your report.

2. Write the SQL SELECT statement that selects all records for injury type "0012." Show only the org, WDLR, and WDL columns on your report.



HINT

Multiple column names are separated by commas.

3. Write the SQL SELECT statement that selects all records for body part "3101" and org "0502009." Show all columns on your report.



HINT

Within a WHERE clause, multiple predicates can be joined with the Boolean operators AND, OR, and NOT.

Grouping Selected Data

If you want to organize the selected data into groups, you must use the GROUP BY modifier. The GROUP BY modifier aggregates rows into groups and then sorts the groups into alphabetical order. The general format of this SELECT statement is:

```
SELECT <column names>  
  FROM <table>  
  WHERE <predicate>  
  GROUP BY <column names>
```

The WHERE clause is optional. However, if a WHERE clause is used, it must come before the GROUP BY clause. GROUP BY places restrictions on how we use column names in a query. Specifically, the column names following SELECT must either be part of aggregate functions (e.g., COUNT or AVG) or they must appear as one of the column names in the GROUP BY clause. For example, you can group the **ACCIDENT** table by organization with the following SELECT statement:

```
SELECT ACCIDENT.ORG, ACCIDENT.NUMBER, ACCIDENT.DATE  
  FROM TEST.ACCIDENT  
  GROUP BY ACCIDENT.ORG, ACCIDENT.NUMBER, ACCIDENT.DATE
```

The result of this SELECT statement is displayed in the table below.

ORG	NUMBER	DATE
0502009	1	19980112
0502009	2	19980115
0502009	10	19980409
0502009	13	19980506
0502009	16	19980619
0502009	17	19980704
0502009	22	19981030
3003003	3	19980131
3003003	6	19980217
3003003	7	19980301
3003003	8	19980321
3003003	9	19980322

ORG	NUMBER	DATE
3003003	14	19980526
3003003	15	19980611
3506502	18	19980707
3506502	19	19980807
3506502	20	19980814
3506502	21	19981010
7503409	4	19980202
7503409	5	19980202
7503409	11	19980409
7503409	12	19980505

The GROUP BY clause, when used with the SQL COUNT function, can also be used to count the number of records associated with column values. For example, you can count the number of cases for each organization with the following SELECT statement:

```
SELECT ACCIDENT.ORG, COUNT(ACCIDENT.NUMBER) 'COUNT'
FROM TEST.ACCIDENT
GROUP BY ACCIDENT.ORG
```

Note that the words in parentheses following the COUNT function specifies the column label. The result of this select statement (shown in the table below) includes a user-defined column which displays the count of records associated with each organization.

ORG	COUNT
0502009	7
3003003	7
3506502	4
7503409	4

The HAVING modifier is used in conjunction with the GROUP BY clause to place restrictions on the groups. The general format for this SELECT statement is:

```
SELECT <column names>  
FROM <table>  
WHERE <predicate>  
GROUP BY <column names>  
HAVING <predicate>
```

The HAVING clause works like the WHERE clause, but it is applied to groups rather than to individual records. The HAVING clause is always executed after grouping. The same predicates used in the WHERE clause can be used in the HAVING clause. For example, you can count the number of cases for each organization and then display only the groups with more than four records with the following SELECT statement:

```
SELECT ACCIDENT.ORG, COUNT(ACCIDENT.NUMBER) 'COUNT'  
FROM TEST.ACCIDENT  
GROUP BY ACCIDENT.ORG  
HAVING (COUNT(ACCIDENT.NUMBER) > 4)
```

The result of this select statement is shown in the table below.

ORG	COUNT
0502009	7
3003003	7

Exercise 2

1. Write the SQL SELECT statement that groups records by injury type. Show columns for injury type, org, and date on your report.

2. Modify the SELECT statement to show the count of records by injury type. Label the column "COUNT."

3. Modify the SELECT statement to show only those counts greater than four records.

Ordering Selected Data

While the GROUP BY modifier aggregates rows into groups and then sorts the groups, the ORDER BY modifier sorts individual rows. The general format for this SELECT statement is:

```
SELECT <column names>  
FROM <table>  
WHERE <predicate>  
ORDER BY <column names>
```

The WHERE clause is optional. The ORDER BY clause must be the last clause specified in the SELECT statement. You can specify either an ascending (ASC) or descending (DESC) sort. The default sort order is ascending. The general format for a descending sort is:

```
SELECT <column names>  
FROM <table>  
WHERE <predicate>  
ORDER BY <column names> DESC
```

For example, you could order the rows in the **ACCIDENT** table by descending date with the following SELECT statement:

```
SELECT *  
FROM TEST.ACCIDENT  
ORDER BY ACCIDENT.DATE DESC
```

Exercise 3

Write the SQL SELECT statement that sorts records by organization and then sorts the records in descending order by date. Display only the organization and date columns on your report.

Joining Tables

There may be times when you require information from multiple tables. For example, along with selected information from the **ACCIDENT** table, you may want to include the description of the organization from the **ORGANIZATION** table. This is easy enough to do. You simply list multiple tables in the FROM clause.

```
SELECT *
FROM TEST.ACCIDENT, TEST.ORGANIZATION
```



The above select statement can be very DANGEROUS because the result will be the Cartesian product of the tables you specify. This is not a major problem if the source tables have only a limited number of records; however, if the source tables have tens of thousands or hundreds of thousands of records the result could be disastrous. This is because each record in the first table will be joined with each record in the second table and the virtual table could become so large that it could potentially cause the server to crash. See **Page 60** in this manual for information on how to properly join the CAIRS database tables to avoid the problem of the Cartesian product.

As displayed in the table below, the result of the above select statement is the Cartesian product of the two source tables.

NUMBER	DATE	ORG	INJURY TYPE	BODY PART	WDLR	WDL	CODE	DESCRIPTION
1	19980112	0502009	0012	3101	20	10	0502009	Albuquerque
1	19980112	0502009	0012	3101	20	10	3003003	Idaho
1	19980112	0502009	0012	3101	20	10	3506502	Nevada
1	19980112	0502009	0012	3101	20	10	7503409	Richland
2	19980115	0502009	0052	3101	0	0	0502009	Albuquerque
2	19980115	0502009	0052	3101	0	0	3003003	Idaho
2	19980115	0502009	0052	3101	0	0	3506502	Nevada
2	19980115	0502009	0052	3101	0	0	7503409	Richland
3	19980131	3003003	0072	4401	0	4	0502009	Albuquerque
3	19980131	3003003	0072	4401	0	4	3003003	Idaho
3	19980131	3003003	0072	4401	0	4	3506502	Nevada
3	19980131	3003003	0072	4401	0	4	7503409	Richland
4	19980202	7503409	0012	4102	8	20	0502009	Albuquerque
4	19980202	7503409	0012	4102	8	20	3003003	Idaho

NUMBER	DATE	ORG	INJURY TYPE	BODY PART	WDLR	WDL	CODE	DESCRIPTION
4	19980202	7503409	0012	4102	8	20	3506502	Nevada
4	19980202	7503409	0012	4102	8	20	7503409	Richland
5	19980202	7503409	0035	3301	5	0	0502009	Albuquerque
5	19980202	7503409	0035	3301	5	0	3003003	Idaho
5	19980202	7503409	0035	3301	5	0	3506502	Nevada
5	19980202	7503409	0035	3301	5	0	7503409	Richland
6	19980217	3003003	0052	0305	0	14	0502009	Albuquerque
6	19980217	3003003	0052	0305	0	14	3003003	Idaho
6	19980217	3003003	0052	0305	0	14	3506502	Nevada
6	19980217	3003003	0052	0305	0	14	7503409	Richland
7	19980301	3003003	0012	2301	68	126	0502009	Albuquerque
7	19980301	3003003	0012	2301	68	126	3003003	Idaho
7	19980301	3003003	0012	2301	68	126	3506502	Nevada
7	19980301	3003003	0012	2301	68	126	7503409	Richland
8	19980321	3003003	0083	3101	0	0	0502009	Albuquerque
8	19980321	3003003	0083	3101	0	0	3003003	Idaho
8	19980321	3003003	0083	3101	0	0	3506502	Nevada
8	19980321	3003003	0083	3101	0	0	7503409	Richland
9	19980322	3003003	0052	3101	0	2	0502009	Albuquerque
9	19980322	3003003	0052	3101	0	2	3003003	Idaho
9	19980322	3003003	0052	3101	0	2	3506502	Nevada
9	19980322	3003003	0052	3101	0	2	7503409	Richland
10	19980409	0502009	0035	4102	2	2	0502009	Albuquerque
10	19980409	0502009	0035	4102	2	2	3003003	Idaho
10	19980409	0502009	0035	4102	2	2	3506502	Nevada
10	19980409	0502009	0035	4102	2	2	7503409	Richland
11	19980409	7503409	0035	0305	0	1	0502009	Albuquerque
11	19980409	7503409	0035	0305	0	1	3003003	Idaho
11	19980409	7503409	0035	0305	0	1	3506502	Nevada
11	19980409	7503409	0035	0305	0	1	7503409	Richland
12	19980505	7503409	0052	3301	0	0	0502009	Albuquerque
12	19980505	7503409	0052	3301	0	0	3003003	Idaho
12	19980505	7503409	0052	3301	0	0	3506502	Nevada

NUMBER	DATE	ORG	INJURY TYPE	BODY PART	WDLR	WDL	CODE	DESCRIPTION
12	19980505	7503409	0052	3301	0	0	7503409	Richland
13	19980506	0502009	0062	0101	0	256	0502009	Albuquerque
13	19980506	0502009	0062	0101	0	256	3003003	Idaho
13	19980506	0502009	0062	0101	0	256	3506502	Nevada
13	19980506	0502009	0062	0101	0	256	7503409	Richland
14	19980526	3003003	0012	3101	5	5	0502009	Albuquerque
14	19980526	3003003	0012	3101	5	5	3003003	Idaho
14	19980526	3003003	0012	3101	5	5	3506502	Nevada
14	19980526	3003003	0012	3101	5	5	7503409	Richland
15	19980611	3003003	0052	4102	10	34	0502009	Albuquerque
15	19980611	3003003	0052	4102	10	34	3003003	Idaho
15	19980611	3003003	0052	4102	10	34	3506502	Nevada
15	19980611	3003003	0052	4102	10	34	7503409	Richland
16	19980619	0502009	0035	3101	0	0	0502009	Albuquerque
16	19980619	0502009	0035	3101	0	0	3003003	Idaho
16	19980619	0502009	0035	3101	0	0	3506502	Nevada
16	19980619	0502009	0035	3101	0	0	7503409	Richland
17	19980704	0502009	0037	3101	0	0	0502009	Albuquerque
17	19980704	0502009	0037	3101	0	0	3003003	Idaho
17	19980704	0502009	0037	3101	0	0	3506502	Nevada
17	19980704	0502009	0037	3101	0	0	7503409	Richland
18	19980707	3506502	1051	2503	67	45	0502009	Albuquerque
18	19980707	3506502	1051	2503	67	45	3003003	Idaho
18	19980707	3506502	1051	2503	67	45	3506502	Nevada
18	19980707	3506502	1051	2503	67	45	7503409	Richland
19	19980807	3506502	0035	4102	0	0	0502009	Albuquerque
19	19980807	3506502	0035	4102	0	0	3003003	Idaho
19	19980807	3506502	0035	4102	0	0	3506502	Nevada
19	19980807	3506502	0035	4102	0	0	7503409	Richland
20	19980814	3506502	0035	4102	0	6	0502009	Albuquerque
20	19980814	3506502	0035	4102	0	6	3003003	Idaho
20	19980814	3506502	0035	4102	0	6	3506502	Nevada
20	19980814	3506502	0035	4102	0	6	7503409	Richland

NUMBER	DATE	ORG	INJURY TYPE	BODY PART	WDLR	WDL	CODE	DESCRIPTION
21	19981010	3506502	0012	3301	45	2	0502009	Albuquerque
21	19981010	3506502	0012	3301	45	2	3003003	Idaho
21	19981010	3506502	0012	3301	45	2	3506502	Nevada
21	19981010	3506502	0012	3301	45	2	7503409	Richland
22	19981030	0502009	0037	4401	0	1	0502009	Albuquerque
22	19981030	0502009	0037	4401	0	1	3003003	Idaho
22	19981030	0502009	0037	4401	0	1	3506502	Nevada
22	19981030	0502009	0037	4401	0	1	7503409	Richland

Generally, this type of result is not very useful. In this case, we only need the table rows where the **ORG** column in the **ACCIDENT** table equals the **CODE** column in the **ORGANIZATION** table. To filter out unwanted rows you use the **WHERE** modifier.

```
SELECT *
FROM TEST.ACCIDENT, TEST.ORGANIZATION
WHERE ACCIDENT.ORG = ORGANIZATION.CODE
```

The results of this SELECT statement are shown in the table below.

NUMBER	DATE	ORG	INJURY TYPE	BODY PART	WDLR	WDL	CODE	DESCRIPTION
1	19980112	0502009	0012	3101	20	10	0502009	Albuquerque
2	19980115	0502009	0052	3101	0	0	0502009	Albuquerque
3	19980131	3003003	0072	4401	0	4	3003003	Idaho
4	19980202	7503409	0012	4102	8	20	7503409	Richland
5	19980202	7503409	0035	3301	5	0	7503409	Richland
6	19980217	3003003	0052	0305	0	14	3003003	Idaho
7	19980301	3003003	0012	2301	68	126	3003003	Idaho
8	19980321	3003003	0083	3101	0	0	3003003	Idaho
9	19980322	3003003	0052	3101	0	2	3003003	Idaho
10	19980409	0502009	0035	4102	2	2	0502009	Albuquerque
11	19980409	7503409	0035	0305	0	1	7503409	Richland
12	19980505	7503409	0052	3301	0	0	7503409	Richland
13	19980506	0502009	0062	0101	0	256	0502009	Albuquerque
14	19980526	3003003	0012	3101	5	5	3003003	Idaho

NUMBER	DATE	ORG	INJURY TYPE	BODY PART	WDLR	WDL	CODE	DESCRIPTION
15	19980611	3003003	0052	4102	10	34	3003003	Idaho
16	19980619	0502009	0035	3101	0	0	0502009	Albuquerque
17	19980704	0502009	0037	3101	0	0	0502009	Albuquerque
18	19980707	3506502	1051	2503	67	45	3506502	Nevada
19	19980807	3506502	0035	4102	0	0	3506502	Nevada
20	19980814	3506502	0035	4102	0	6	3506502	Nevada
21	19981010	3506502	0012	3301	45	2	3506502	Nevada
22	19981030	0502009	0037	4401	0	1	0502009	Albuquerque

Exercise 4

Write the SQL **SELECT** statement that joins the **ACCIDENT** table and the **INJURY** table and selects only rows from the joined table when the **INJURY_TYPE** column in the **ACCIDENT** tables equals the **CODE** column in the **INJURY** table. Display only injury type, description, and date columns on your report.

CAIRS Data Structure

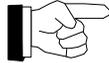
Appendix B describes the CAIRS database schema (the CAIRS tables, with their associated fields). In general, the table and field structure and definitions are identical to those seen on the HP. Additional tables or fields have also been added to new CAIRS database to facilitate queries. Fields that contain privacy information are restricted and therefore, not available for query.

You should be aware that the information presented in this appendix is preliminary and subject to change as the CAIRS design progresses. An updated CAIRS schema will be made available as changes are made.

Microsoft Query

Microsoft Query is a Windows add-in that allows you to ask questions (called a query) about the data in a database and then returns the answer as a set of records (called the result set). With the Microsoft Query you can query almost any type of database if the database drivers are installed.

NOTE



The drivers for performing queries of the CAIRS SQL database are installed during the CAIRS direct access installation procedure (See *Direct Access Installation* under the *Introduction* section of this guide).



TERMINOLOGY

Query

The question or questions asked about data in a database.



TERMINOLOGY

Result Set

The set of records that are returned in response to a query.

To start Microsoft Query, double-click on the **MICROSOFT QUERY SHORTCUT** on your Windows desktop. The **MICROSOFT QUERY** window (shown in **Figure 1**) appears.

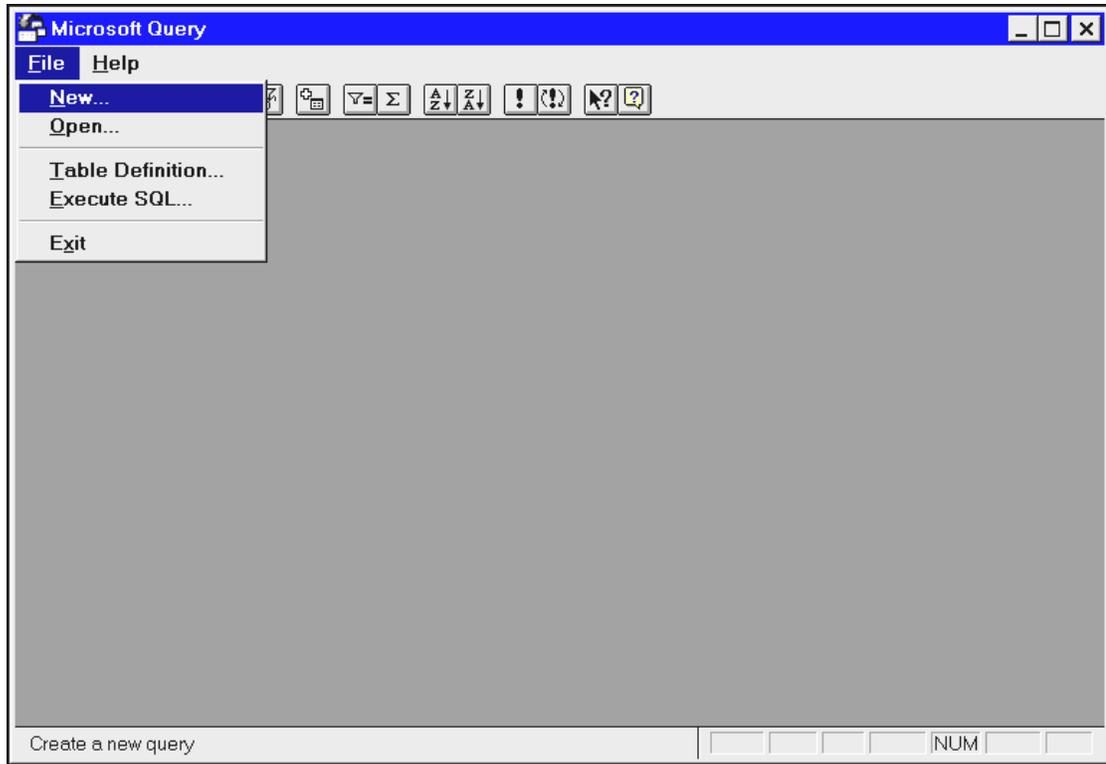


Figure 1 - The **MICROSOFT QUERY** window with **FILE** menu selections displayed.

Creating CAIRS as a Data Source

Before you can establish a link between Microsoft Query and CAIRS you must create CAIRS as a new data source.



TERMINOLOGY

Data Source

Identifies the database you want to gain access to and includes the information needed to connect to the data.

You do this by selecting **New** from the **FILE** menu (**Figure 1**). The **CHOOSE DATA SOURCE** dialog box appears (**Figure 2**). With **<New Data Source>** selected in the **DATABASES** selection area, you click on the **OK** command button.

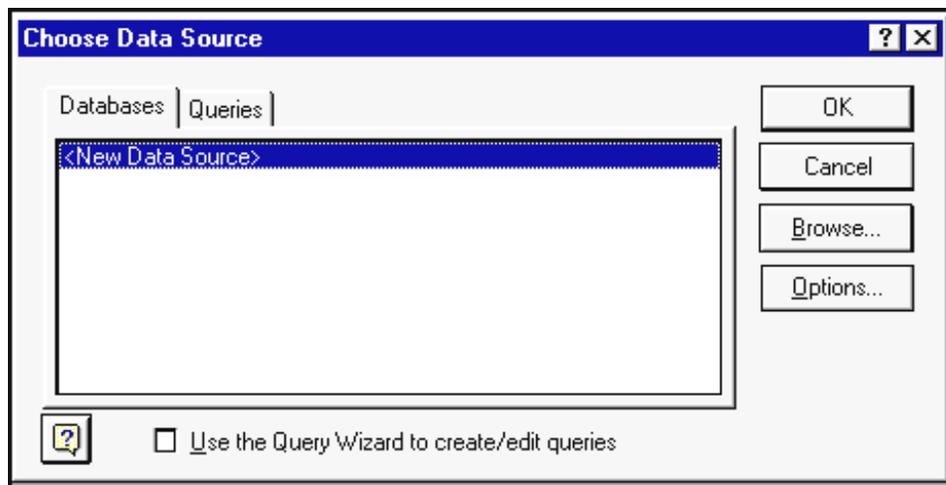


Figure 2 - The **CHOOSE DATA SOURCE** dialog box.

The **CREATE NEW DATA SOURCE** dialog box appears (**Figure 3**). The steps for creating a new data source are numbered in the dialog box.

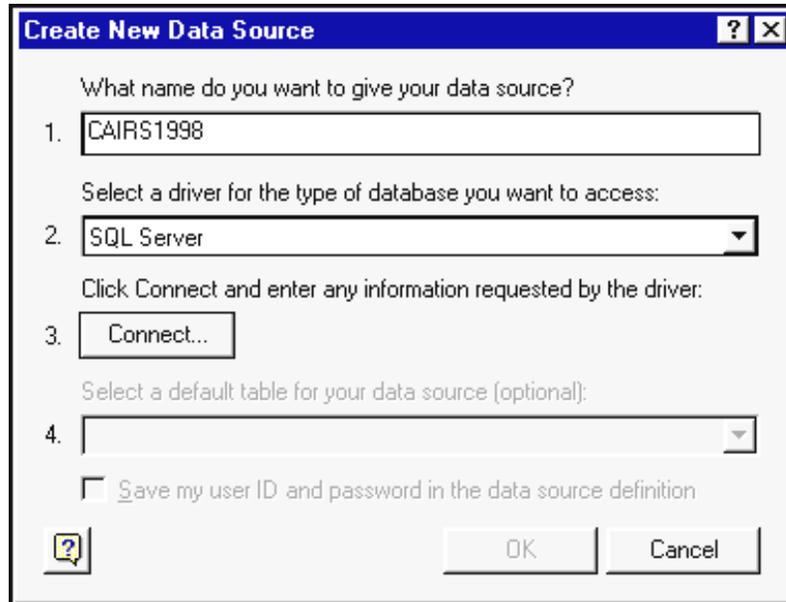


Figure 3 - The **CREATE NEW DATA SOURCE** dialog box.

- Step 1: Enter a source name into the first edit box (e.g., CAIRS1998).
- Step 2: Select the CAIRS driver by clicking on the down arrow () in the selection box and selecting **SQL Server**.
- Step 3: Click on the **CONNECT** command button. The **SQL SERVER LOGIN** dialog box appears (**Figure 4**). Select **CAIRS1998** from the **SERVER** selection box, enter your user ID in the **LOGIN ID** edit box, your password in the **PASSWORD** edit box. Click on the **OK** command button. A link is established to CAIRS and you are returned to the **CREATE NEW DATA SOURCE** dialog box.

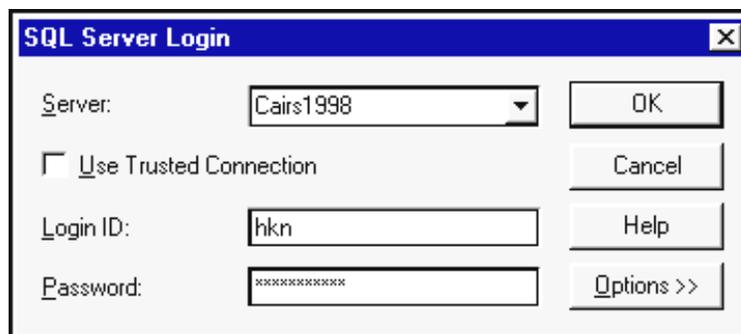


Figure 4 - The **SQL SERVER LOGIN** dialog box.

Step 4: This step, which is optional, allows you to select a default table for your data source. As the **GICSVIEW** table contains most of the record data and therefore, will be the most frequently used table, you may want to select it as your default table. Click on the down arrow () in the selection box and select **gicsView** from the table list.

To save CAIRS as a data source, click on the **OK** command button. CAIRS is now shown as a new data source in the **CHOOSE DATA SOURCE** dialog box.



Do not click on the check box to save your user ID and password. Microsoft Query does not encrypt your password when saving it in the data source definition. It will be visible to anyone who looks at the definition.



You only have to create CAIRS as a database source the first time you link to CAIRS from Microsoft Query. You perform subsequent linkings by selecting the CAIRS database from the **CHOOSE DATABASE SOURCE** dialog box and clicking on the **OK** command button (**Figure 5**).

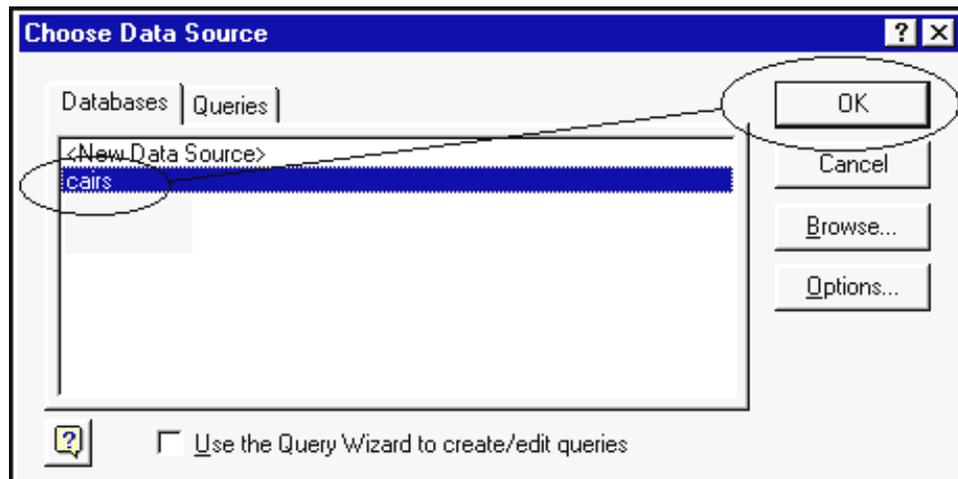


Figure 5 - The **CHOOSE DATA SOURCE** dialog box displaying CAIRS as a data source.



The **USE THE QUERY WIZARD TO CREATE/EDIT QUERIES** check box (see **Figure 5**) is checked by default. Uncheck this box, or you will activate the Wizard each time you select a data source.



TUTORIAL

Linking to the CAIRS Database with Microsoft Query

These steps are performed only after CAIRS has been created as a data source.

1. Double-click on the **MICROSOFT QUERY SHORTCUT** on your Windows desktop.
2. Select **New** from the **FILE** menu.
3. Select the CAIRS data source from the **DATABASES** selection area of the **CHOOSE DATA SOURCE** dialog box and then click on the **OK** command button.
4. Enter your user ID into the **USER NAME** edit box and your password into the **PASSWORD** edit box and then click on the **OK** command button. Microsoft Query will connect to CAIRS.



Creating CAIRS as a Data Source with Office 95

1. Select **New Query** from the **FILE** menu on the **MICROSOFT QUERY** window.
2. Click on the **OTHER** command button from the **SELECT DATA SOURCE** dialog box. The **ODBC DATA SOURCE** dialog box will appear.
3. Double-click on **CAIRS1998**. Enter your user ID in the **USER NAME** edit box, your password in the **PASSWORD** edit box, and **CAIRS1998** in the **SERVER** edit box. Then click on the **OK** command button. CAIRS is now listed as a data source in the **AVAILABLE DATA SOURCES** list.

The Microsoft Query Window

The Microsoft Query Window Panes

Once you have established a link to CAIRS, a segmented query window is displayed (**Figure 6**). The first segment, called the **TABLE PANE**, contains information about tables in the query. Because you set the **GICSVIEW** table as the default table when you created CAIRS as a new data source, it will be displayed in the **TABLE PANE** every time you establish a link to CAIRS. The second segment, called the **CRITERIA PANE**, contains information about the criteria you have selected to limit the records in the result set, and the third segment, called the **DATA PANE**, contains the result set.

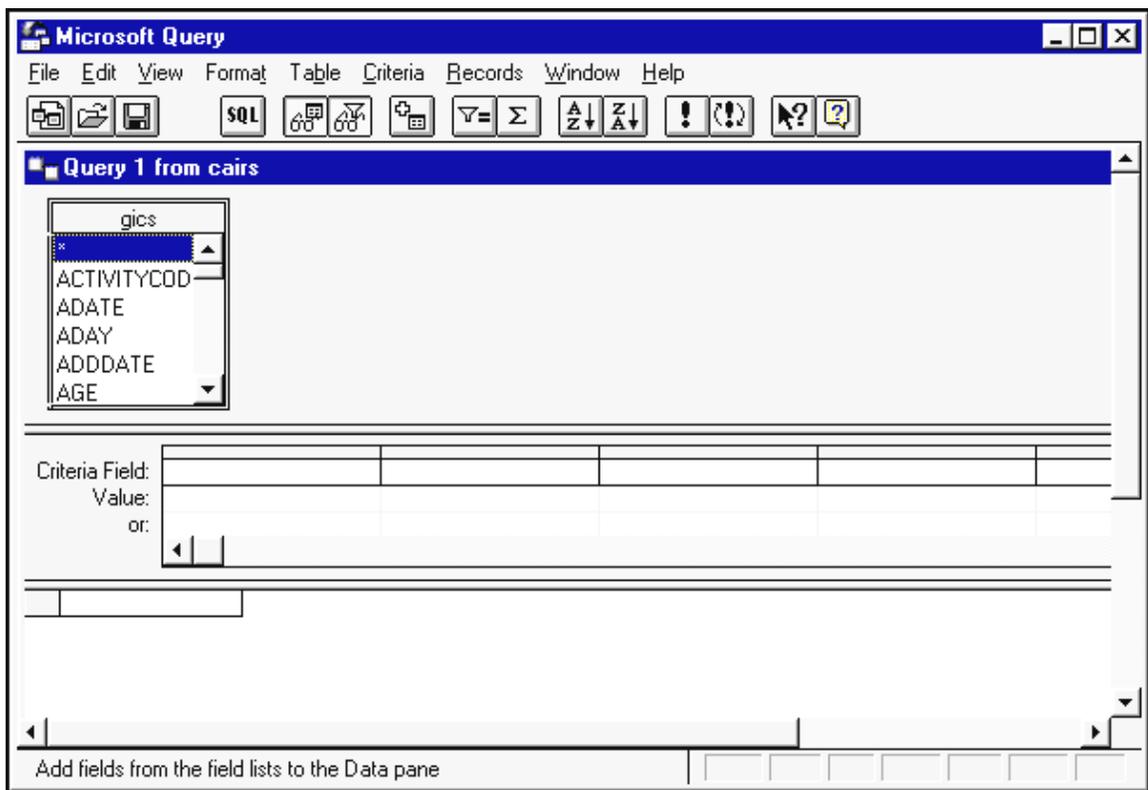


Figure 6 - The Microsoft Query window panes.

The Microsoft Query Menu Bar

The menu bar at the top of the window (See **Figure 6**) contains the following options:

FILE	This menu contains the operations that establish links to a database and that open queries, close queries, and save queries to disk.
EDIT	This menu contains options for copying text, undoing the last action performed, limiting the number of records returned to the query window, and configuring connection time to the database.
VIEW	This menu contains options for viewing the window panes and setting some query properties. In addition, there are options for viewing and editing user-entered parameters and SQL SELECT statements.
FORMAT	This menu contains options for formatting the DATA PANE .
TABLE	This menu contains options for adding, removing, or joining tables.
CRITERIA	This menu contains options for adding or removing criteria that records must meet to be included in the result set.
RECORDS	This menu contains options for adding or removing columns in the DATA PANE . It also contains options for sorting or refreshing the result set.
WINDOW	This menu contains options for how the query window will be displayed on the screen.
HELP	This menu contains information about Microsoft Query and options for activating the help utilities.

The Microsoft Query Tool Bar

The tool bar provides various *shortcuts* to menu options. Each button is described below.



The **NEW QUERY** button initiates a new query. It displays the **CHOOSE DATA SOURCE** dialog box, where you either choose the data source you want for the new query or select an existing query to change.



The **OPEN QUERY** button opens an existing query. It displays the **OPEN QUERY** dialog box, from which you open an existing query for viewing or editing.



The **SAVE FILE** button saves a query, replacing any previous version with the current version. If you haven't saved the query before, Microsoft Query displays the **SAVE AS** dialog box.



The **RETURN DATA** button returns data to another application. If you start Microsoft Query from another program, click on this button to return data from Microsoft Query to the original program.



The **VIEW SQL** button displays the corresponding SQL SELECT statement for the query in the query window, so you can view or edit the statement.



The **SHOW/HIDE TABLES** button displays or hides the **TABLE PANE**.



The **SHOW/HIDE CRITERIA** button displays or hides the **CRITERIA PANE**. Microsoft Query doesn't display the **CRITERIA PANE** automatically when you create a query. The **CRITERIA PANE** appears the first time you specify criteria by clicking on **Add Criteria** on the **CRITERIA** menu or by clicking the **SHOW/HIDE CRITERIA** button.



The **ADD TABLE(S)** button displays the **ADD TABLES** dialog box, which you can use to add one or more tables to the **TABLE PANE** in your query.



The **CRITERIA EQUALS** button specifies a criterion that selects only those records containing the same value as the active cell. The cell you select in the result set becomes the value for the criterion in the **CRITERIA PANE**.



The **CYCLE THROUGH TOTALS** button allows you to select totals for a column, depending on the type of data stored in the field.



The **SORT ASCENDING** button sorts records in ascending order from the beginning of the alphabet, the lowest number, or the earliest date by using the selected field.



The **SORT DESCENDING** button sorts records in ascending order from the end of the alphabet, the highest number, or the latest date by using the selected field.



The **QUERY NOW** button runs the query and displays the most current result set in the **DATA PANE**. When **Automatic Query** is on, Microsoft Query runs the query each time you add a field, specify criteria, or otherwise change the query design. When **Automatic Query** is off, you can only run the query by clicking on the **QUERY NOW** button.



The **AUTOMATIC QUERY** button turns **Automatic Query** on or off. This button controls whether the query is run every time you change it. By default, **Automatic Query** is on.



Click on the **HELP** button and then click on menu command, tool, or screen region to display context sensitive help for that item.



The **OFFICE ASSISTANT** button opens the **Microsoft Office 97 Assistant**. The assistant provides Help topics and tips to help you accomplish your tasks.

Designing a Query Specifying the Criteria



TERMINOLOGY

Criteria

Conditions you specify to limit which records are included in the result set of a query.

Criteria are added to a new query from the **ADD CRITERIA** dialog box. Criteria can only be specified for database tables displayed on the **TABLE PANE**. To open the dialog box (Shown in **Figure 7**), select **Add Criteria** from the **CRITERIA** menu.

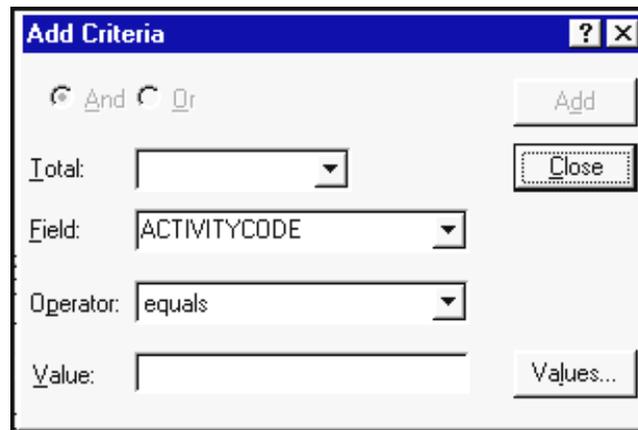


Figure 7 - The **ADD CRITERIA** dialog box.

The elements in the dialog box represent the components of the **SELECT** statement predicate. In the *Structured Query Language* section of this guide you learned that the general structure of a predicate is:

<column name> <operator> <value>

From the **ADD CRITERIA** dialog box, you specify *<column name>* with the **FIELD** selection box, *<operator>* with the **OPERATOR** selection box, and *<value>* with the **VALUE** edit box. The first field in the **GICSVIEW** table is the default selection in the **FIELD** selection box, and the “**equals**” operator is the default selection in the **OPERATOR** selection box. The **ADD** command button is used to add a predicate to your select statement. The **ADD** command button is disabled until you specify all information required to construct the syntax for a predicate within a **SELECT** statement. The **AND** and **OR** radio buttons are

used to specify the Boolean logic between predicates. These radio buttons are disabled until the first predicate has been added to the SELECT statement.

For example, to retrieve the latest version of all reports for the DOE Central Training Academy, you must define two predicates or criterium. To define the first predicate, select **ORG** from the **FIELD** selection box and **equals** from the **OPERATOR** selection box.



For **FIELD** selection boxes containing numerous items, open the selection area by clicking on the down arrow (), click in the box and then begin typing the characters of the desired item. You will be taken to the item that most closely matches the characters you have typed.

Next, click on the **VALUES** command button to open the **SELECTED VALUE(S)** dialog box (**Figure 8**). All possible values for the **ORG** field are displayed in the **VALUES** selection box. Select **0502209** from the **VALUES** selection box and then click on the **OK** command button. **0502209** is automatically entered into the **VALUE** edit box and the **ADD** command button is enabled.

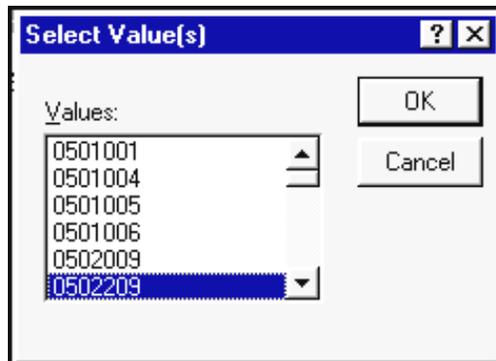


Figure 8 - The **SELECTED VALUE(S)** dialog box.



You can select multiple values from the dialog box by clicking on a value with the left mouse button. Values are deselected in the same way.

Finally, click on the **ADD** command button. The criterium is added to the **CRITERIA PANE** (**Figure 9**).

Criteria Field:	ORG			
Value:	'0502209'			
or:				

Figure 9 - The **CRITERIA PANE** displaying the predicate *ORG = '0502209'*.

To define the second predicate, return to the **ADD CRITERIA** dialog box. (Notice that the **AND** and **OR** radio buttons are enabled, **Figure 10**.)

Figure 10 - The **ADD CRITERIA** dialog box with the **AND** and **OR** radio buttons enabled.

Select **CURRENTFLAG** from the **FIELD** selection box, select **equals** from the **OPERATOR** selection box, and type **Y** in the **VALUE** edit box.



If you know the permitted values for a field, you can enter the value directly into the **VALUE** edit box. The values are not case sensitive. Multiple values are separated with commas.

Finally, click on the **ADD** command button. The criterium is added to the **CRITERIA PANE** (**Figure 11**). To close the **ADD CRITERIA** dialog box, click on the **CLOSE** command button.

Criteria Field:	ORG	CurrentFlag			
Value:	'0502209'	'Y'			
or:					

Figure 11 - The **CRITERIA PANE** displaying the predicate *ORG* = '0502209' and the predicate *CURRENTFLAG* = 'Y'.



Multiple versions of an occurrence report are stored in the **GICSVIEW** table. The **CURRENTFLAG** field is used to designate the latest version of a report. Therefore, to retrieve the latest version of a report, specify the criteria, "*CURRENTFLAG* = 'Y'".

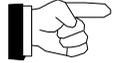


TUTORIAL

Adding Criteria to a Query

1. Select **Add Criteria** from the **CRITERIA** menu to display the **ADD CRITERIA** dialog box.
2. Select a field (table column) from the **FIELD** selection box.
3. Select an operator from the **OPERATOR** selection box.
4. Enter a value in the **VALUE** edit box.

NOTE



If you select either **is Null** or **is Not Null** as the operator, the **VALUE** edit box will be disabled. Values are not used with these operators.

5. Click on the **ADD** command button.

Adding Report Columns

Columns that you want to include in your report are added from the **ADD COLUMN** dialog box. These columns are displayed on the **DATA PANE**. To open the dialog box (Shown in **Figure 12**), select **Add Column** from the **RECORDS** menu.

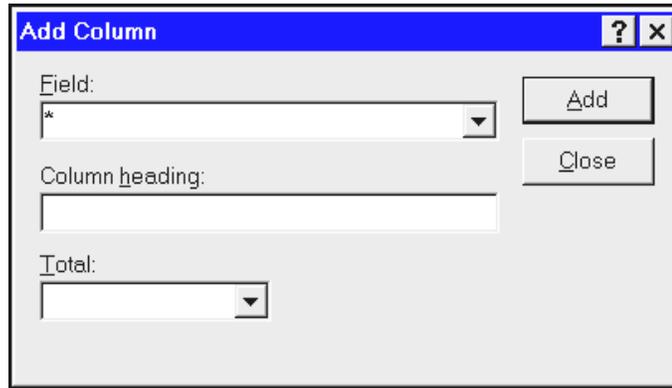


Figure 12 - The **ADD COLUMN** dialog box.

Table columns are selected from the **FIELD** selection box. The asterisk (*) wildcard operator, which selects all columns from the table(s) displayed on the **TABLE PANE**, is the default selection. The **COLUMN HEADING** edit box allows you to name the columns on your report. If you do not specify a name, the field name is used as the column name.

For example, to display the accident date and one-line description as report columns, click on the down arrow () in the **FIELD** selection box and select **ADATE**, type **Accident Date** in the **COLUMN HEADING** edit box, and click on the **ADD** command button. **Accident Date** is added as the first column heading and the result set is displayed in the **DATA PANE**.



HINT

Microsoft Query automatically runs a query and refreshes the data in the result set each time you add a field to the **DATA PANE** or add or change criteria in the **CRITERIA PANE**. If your query contains a large number of records, you might want to wait to run the query until you've finished making changes to the query. To turn off the automatic query feature, click on the **AUTOMATIC QUERY** tool bar button.

Next, click on the down arrow in the **FIELD** selection box and select **Descript**, type **One-Line Description** in the **COLUMN HEADING** edit box, and click on the **ADD** command button. **One-Line Description** is added as the second column heading and the result set is displayed in the **DATA PANE** (See **Figure 13**). Click on the **CLOSE** command button to close the dialog box.

Accident Date	One-Line Description
19860520	SECURITY GUARD experienced MULTIPLE INJURIES, INTERNAL AND EXTER
19880310	SECURITY GUARD experienced FRACTURE to his/her ARM(S) involving a/an 5
19880813	0 deaths, 0 injuries, and \$742 property loss in Highway vehicle, unspecified accide
19881215	SECURITY GUARD experienced INJURY to his/her FINGER(S) involving a/an 01
19890118	OTHER TECHNICIAN experienced INJURY to his/her MULTIPLE BODY PARTS
19890518	OTHER TECHNICIAN experienced INJURY to his/her MUSCLE/TENDON(S) invc
19890622	OTHER TECHNICIAN experienced INJURY to his/her GROIN involving a/an BO>
19890907	SECURITY GUARD experienced PUNCTURE to his/her FOOT/FEET involving a
19900131	SECURITY GUARD experienced MULTIPLE INJURIES, INTERNAL AND EXTER
19900201	SECURITY GUARD experienced MULTIPLE INJURIES, INTERNAL AND EXTER
19900206	HANDLER/LABORER/HELPER experienced CONTUSIONS to his/her FACE invc
19900626	SECURITY GUARD experienced STRAIN to his/her MUSCLE/TENDON(S) invol
19900629	0 deaths, 0 injuries, and \$3587 property loss in Highway vehicle, unspecified acci
19910130	ADMIN. SUPPORT/CLERICAL EMPLOYEE experienced STRAIN to his/her LOW

Figure 13 - The DATA PANE displaying a result set.



A quick way to add a column is to double-click on the field you want to add in the TABLE PANE.



Step-by-Step
TUTORIAL

Adding Report Columns

1. Select **Add Column** from the **RECORDS** menu to display the **ADD COLUMN** dialog box.
2. Select a field (table column) from the **FIELD** selection box.
3. Type the column heading into the **COLUMN HEADING** edit box. (This step is optional.)
4. Click on the **ADD** command button.

Saving the Query

A new query is saved as a .dqy file from the **SAVE AS** dialog box. To open the dialog box (Shown in **Figure 14**), select **Save** from the **FILE** menu. To save a new query file in the default location, type the file name in the **FILE NAME** edit box and click on the **SAVE** command button. An advantage to saving the query in the default location is that the file name of the query appears on the **QUERIES** tab when you open a new query. To save a new query file in a location other than the default, specify the directory in the **SAVE IN** selection box, type the file name in the **FILE NAME** edit box and click on the **SAVE** command button.

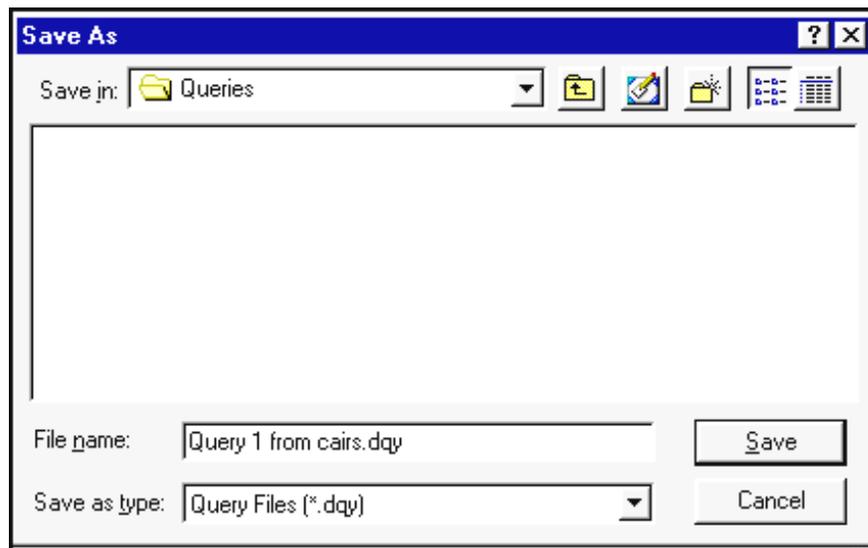


Figure 14 - The **SAVE AS** dialog box.



When you save a query file, Microsoft Query does not save the current result set displayed in the **DATA PANE**. It saves the query (i.e., the SQL **SELECT** statement). In fact, the result set is automatically refreshed, before the **SAVE AS** dialog box is displayed, even when the **AUTO QUERY** tool bar button is not activated.



Another way to save a new file is to click on the **SAVE FILE** tool bar button.



With Office 95, your queries are saved as .qry files. Office 97 displays an option to save queries as .qry files; however, doing this will result in a Microsoft Query connection error when you try to retrieve the file.

Exercise 5

As you work, write down the steps you take. These notes will act as a reminder when you take similar actions in the future.

1. Specify criteria to select the latest version of all 1997-1998 reports for Lawrence Berkeley Laboratory (8003003) and Lawrence Livermore National Laboratory (8004003).
2. Display the accident year and occupation code as report columns.
3. Save the query.

Viewing the SQL SELECT Statement

The SQL SELECT statement for your query is displayed in the **SQL** dialog box (Figure 15). To open the dialog box, click on the **VIEW SQL** tool bar button. From here you can view or modify the instructions that are sent to the server.

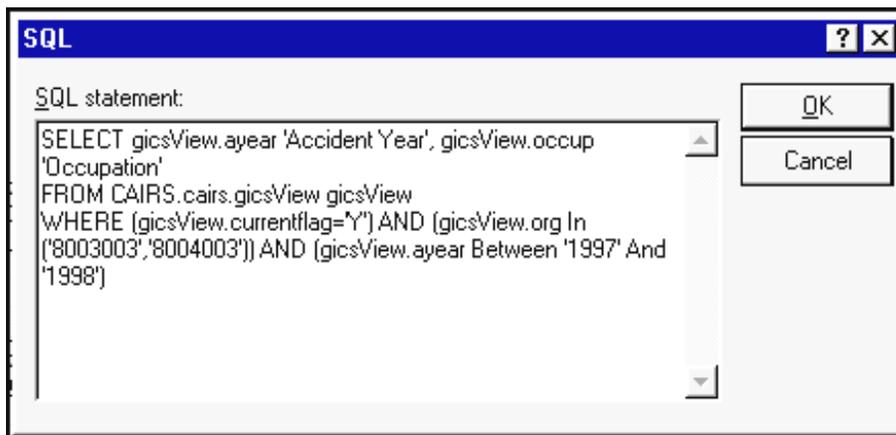


Figure 15 - The SQL dialog box.

The SELECT statement in **Figure 15** displays the statement that was formulated by Microsoft Query in **Exercise 5**. The clause, “*SELECT gicsView.AYEAR 'Accident Year', gicsView.OCCUP 'Occupation'*” specifies the table columns and headings to be included in the report. You populate this clause when you make selections within the **ADD COLUMN** dialog box. The clause, “*FROM CAIRS.cairs.gicsView gicsView*” specifies the table(s) to be included in the query. All tables displayed on the **TABLE PANE** are included in the query. The clause, “*WHERE (gicsView.currentflag='Y') AND (gicsView.org In ('8003003','8004003')) AND (gicsView.ayear Between '1997' And '1998')*” specifies the criteria for the individual records. You populate this clause when you make selections within the **ADD CRITERIA** dialog box.



Both the WHERE clause and the HAVING clause are populated when you make selections within the **ADD CRITERIA** dialog box. Microsoft Query makes it’s “best guess” when adding predicates to the SELECT statement. However, there are occasions when Microsoft Query will attach a predicate to a HAVING clause, which restricts groups, when what you really want is the predicate attached to a WHERE clause to restrict the records (or vice versa). When this occurs, you will need to manually change the SELECT statement in the **SQL** dialog box.

Setting Query Properties

Two query properties are set from the **QUERY PROPERTIES** dialog box (**Figure 16**). To open the dialog box, select **Query Properties** from the **VIEW** menu.



Figure 16 - The **QUERY PROPERTIES** dialog box.

The **Unique Values Only** property hides duplicate values in your query’s result set by adding the **DISTINCT** keyword to your **SELECT** clause (**Figure 17**). To activate this property, you just check the **UNIQUE VALUES ONLY** check box and click on the **OK** command button. The virtual table in the **DATA PANE** will be recreated to display only the unique values (**Figure 18**).

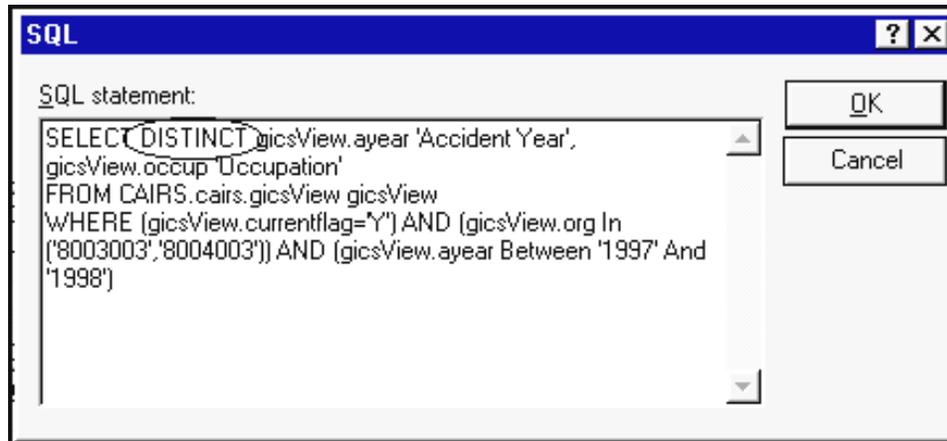


Figure 17 - Example of a SELECT statement using the DISTINCT keyword.

Accident Year	Occupation
1998	1109
1997	5303
1997	0209
1997	1301
1997	1101
1997	1109
1997	1104
1997	1301
1997	0101
1997	0101
1997	1301
1997	1109
1997	1109
1997	1301
1997	0101
1998	1301
1997	1301
1997	1104
1997	1109
1997	1301

Record: 663

Before Distinct

Accident Year	Occupation
1997	0101
1997	0202
1997	0203
1997	0204
1997	0209
1997	1101
1997	1102
1997	1104
1997	1109
1997	1301
1997	2101
1997	2209
1997	4101
1997	4201
1997	4401
1997	5302
1997	5303
1997	5304
1997	5309
1997	5401

Record: 47

After Distinct

Figure 18 - Example of an application of the DISTINCT keyword or the GROUP BY clause.

The **Group Records** property allows you to organize your result set into groups by adding a GROUP BY clause to your SELECT statement. To activate this property, you just check the **GROUP RECORDS** check box and click on the **OK** command button. The virtual table in the **DATA PANE** will be recreated to display the grouping (**Figure 19**).

Accident Year	Occupation
1998	0204
1998	1101
1998	1101
1998	1301
1998	1109
1998	1109
1998	1301
1998	1109
1998	1301
1998	1109
1998	1109
1998	1301
1998	1109
1998	1101
1998	1301

Before Group By

Accident Year	Occupation
1997	0204
1997	0209
1997	0209
1997	1101
1997	1101
1997	1102
1997	1104
1997	1104
1997	1109
1997	1109
1997	1301
1997	1301
1997	2101
1997	2209

After Group By

Figure 19 - Example of an application of the GROUP BY clause.

To know exactly how the records are being grouped, you will need to view the SQL statement (**Figure 20**). In this example, the records are first grouped by all columns cited in the SELECT and the HAVING clause, then the groups are restricted by the predicates in the HAVING clause.

```

SQL
SQL statement:
SELECT gicsView.ayear 'Accident Year', gicsView.occup
'Occupation'
FROM CAIRS.cairs.gicsView gicsView
GROUP BY gicsView.ayear, gicsView.occup,
gicsView.currentflag, gicsView.org
HAVING (gicsView.currentflag='Y') AND (gicsView.org In
('8003003','8004003')) AND (gicsView.ayear Between '1997' And
'1998')
  
```

Figure 20 - Example of a SELECT statement with a GROUP BY clause and a HAVING clause.

This was Microsoft Queries best guess on how to group the records. However, if your intent was to form a group of only the columns cited in the SELECT clause, which is the minimum requirement for columns in a GROUP BY clause, and to restrict the records rather than the groups, you will need to manually change the select statement in the SQL dialog box (**Figure 21**).

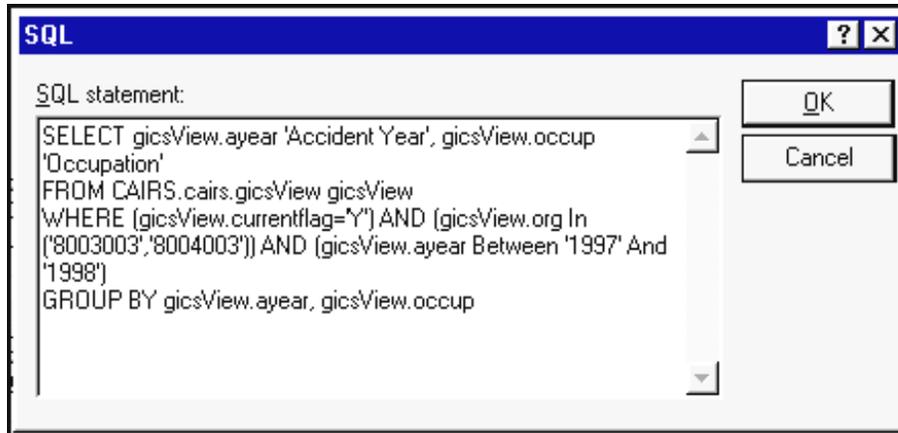


Figure 21 - Example of a SELECT statement with a GROUP BY clause and a WHERE clause.

Sorting Records

After you have added one or more columns to the **DATA PANE**, the Microsoft Query sort option is enabled. This option allows you to sort the result set by values in a column by adding an ORDER BY clause to your SELECT statement (**Figure 22**).

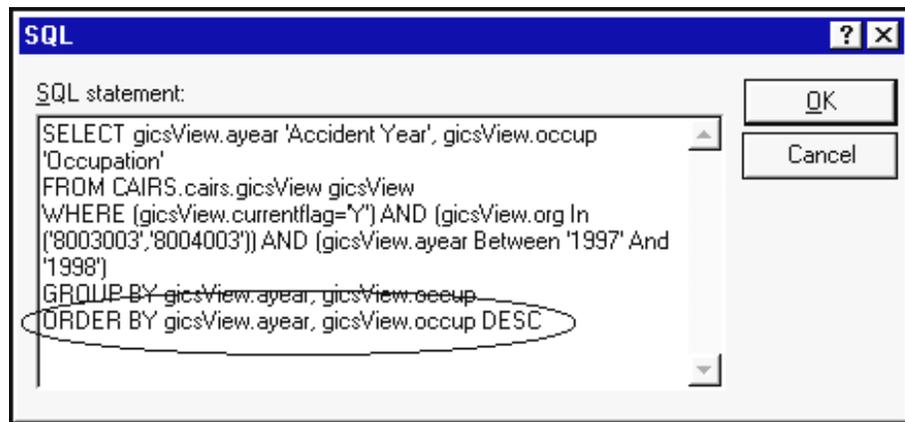


Figure 22 - Example of a SELECT statement using the ORDER BY clause.

Sorting options are specified from the **SORT** dialog box. To open the dialog box (shown in **Figure 23**), select **Sort** from the **RECORDS** menu.

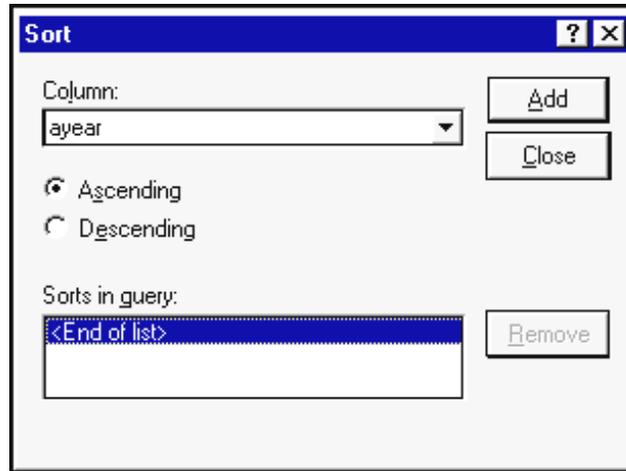
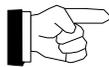


Figure 23 - The **Sort** dialog box.

The sort column is selected from the **COLUMN** selection box. You can sort records in an ascending order by selecting the **ASCENDING** radio button or a descending order by selecting the **DESCENDING** radio button. The **SORTS IN QUERY** list box displays the sorts that are defined for the query.

NOTE



An ascending sort sorts records from the beginning of the alphabet, the lowest number, or the earliest date. A descending sort sorts records from the end of the alphabet, the highest number, or the latest date.

Microsoft Query permits you to sort on multiple fields. For example, to specify a primary ascending sort by Accident Year and then a secondary descending sort by Occupation, select **ayear** from the **COLUMN** selection box, select the **ASCENDING** radio button, and click on the **ADD** command button. “(Asc)ayear” has been added to the **SORTS IN QUERY** list box and the **DATA PANE** is recreated to reflect the primary sort (**Figure 24**). Next, select **occup** from the **COLUMN** selection box, select the **DESCENDING** radio button, and click on the **ADD** command button. “(Desc)occup” has been added to the **SORTS IN QUERY** list box and the **DATA PANE** is recreated to reflect the secondary sort (**Figure 25**). Click on the **CLOSE** command button to close the **Sort** dialog box.

Accident Year	Occupation
1997	0101
1997	0202
1997	0203
1997	0204
1997	0209
1997	1101
1997	1102
1997	1104
1997	1109
1997	1301
1997	2101
1997	2209
1997	4101
1997	4201

Record: 47

Figure 24 - The **DATA PANE** displaying a primary ascending sort by the **ACCIDENT YEAR** column.

Accident Year	Occupation
1997	7102
1997	7101
1997	5401
1997	5309
1997	5304
1997	5303
1997	5302
1997	4401
1997	4201
1997	4101
1997	2209
1997	2101
1997	1301
1997	1109

Record: 47

Figure 25 - The **DATA PANE** displaying a primary ascending sort by the **ACCIDENT YEAR** column and a secondary descending sort by the **OCCUPATION** column.



A quick way to perform sorts is to use the **SORT ASCENDING** and **SORT DESCENDING** tool bar buttons. In the **DATA PANE**, click on the column you want to sort on and then click on a sort button. To sort on multiple columns, click on the next column you want to sort on, hold down the **[CTRL]** key and click on a sort button.

Exercise 6

As you work, write down the steps you take. These notes will act as a reminder when you take similar actions in the future.

1. Specify criteria to select the latest version of all injury/illness cases for the Richland Field Office.
2. Display the organization code and accident year as report columns.
3. Do a primary ascending sort for organization code and a secondary ascending sort for accident year.
4. Organize your result set into groups by organization and accident year.
5. Save the query.

Adding Calculated Columns

Microsoft Query allows you to add calculated columns to your report. Calculated columns are added from the **ADD COLUMN** dialog box. To open the dialog box, select **Add Column** from the **RECORDS** menu. In the **TOTAL** selection box, functions are listed that will return values for the field specified in the **FIELD** selection box based on the results of a calculation or other operation. You can select from the functions described in the table below.

Function	Description
AVG	Returns the average of a set of values in a field.
COUNT	Returns the total number of records.
MAX	Returns the maximum set of values in a field.
MIN	Returns the minimum set of values in a field.
SUM	Returns the sum of a set of values in a field.



Only the COUNT, MAX, and MIN functions can be applied to character data type fields. The number data type field will support all of the functions described in the table above.

As an example, the result set from *Exercise 6* would be more meaningful if the total number of occurrence reports is included as a column. To add this calculated column, select **gic** from the **FIELD** selection box, type **Total Cases** in the **COLUMN HEADING** edit box, and select **COUNT** from the **TOTAL** selection box (see **Figure 26**). **Total Cases** is added as a column heading and the result set is recreated and displayed in the **DATA PANE** (**Figure 27**).

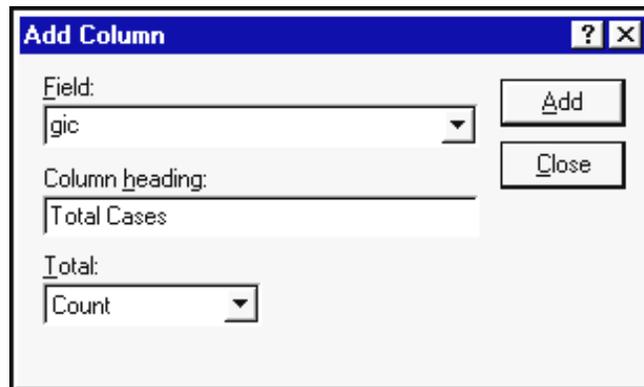


Figure 26 - The **ADD COLUMN** dialog box used for adding calculated fields.

Organization Code	Accident Year	
7500503	1981	
7500503	1982	
7500503	1983	
7500503	1984	
7500503	1985	
7500503	1986	
7500503	1987	
7500503	1988	
7500503	1989	
7500503	1990	
7500503	1991	
7500503	1992	
7500503	1993	
7500503	1994	

Before the COUNT function

Organization Code	Accident Year	Total Cases
7500503	1981	21
7500503	1982	21
7500503	1983	36
7500503	1984	26
7500503	1985	23
7500503	1986	17
7500503	1987	22
7500503	1988	11
7500503	1989	17
7500503	1990	35
7500503	1991	90
7500503	1992	101
7500503	1993	130
7500503	1994	130

After the COUNT function

Figure 27 - The result set both before and after the addition of a calculated column.

Exercise 7

As you work, write down the steps you take. These notes will act as a reminder when you take similar actions in the future.

1. Specify criteria to select the latest version of all injury/illness cases that were reported for 1997.
2. Display by field office the total number of cases, the total number of lost workdays (LWD), and the average number of lost workdays.

HINT



Total number of cases is calculated with the COUNT function, the total number of lost workdays is calculated with the SUM function, and the average number of lost workdays is calculated with the AVG function.

3. Save the query.

Adding and Joining Tables

There may be times when you would like the result set to contain information from multiple tables. This is a two-step process. First you add the additional table to the **TABLE PANE** and then you join the tables. A join shows how data is related between two tables and determines which records are included in the result set. Tables are added to the **TABLE PANE** from the **ADD TABLES** dialog box (**Figure 28**), and joins are created directly from the **TABLE PANE**.

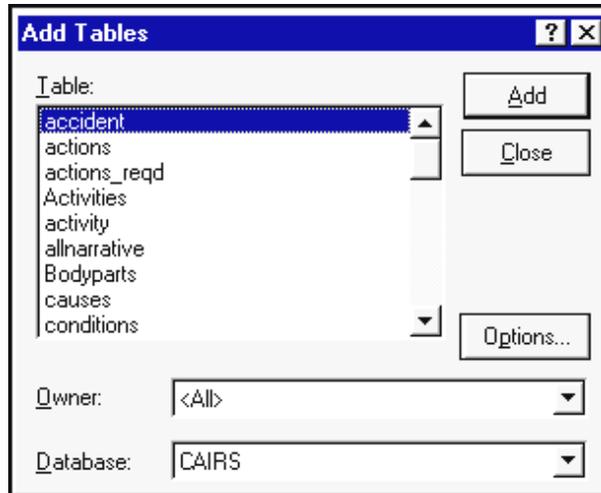


Figure 28 - The **ADD TABLES** dialog box.

For example, to create a distribution by activity description for the Idaho Field Office for the latest versions of occurrences reported in 1997, you select **Add Tables** from the **TABLE** menu to open the **ADD TABLES** dialog box. Select **Activities** from the **TABLE** selection box and then click on the **ADD** command button. Click on the **CLOSE** command to close the dialog box. The **ACTIVITIES** table is displayed on the **TABLE PANE** next to the **GICSVIEW** table. Adding this additional table allows you to create a query from the values in both tables.

To join the tables you simply drag the **ACTIVITYCODE** field on the **GICSVIEW** table to the **CODE** field on the **ACTIVITIES** table. You may get a warning message that says “Columns activitycode and Code that you are about to join are of different types. Create the join anyway?” (**Figure 29**). In this case, you get the warning message because Code is of type char, whereas activitycode is of type varchar. Because the values in both fields are characters of exactly four characters in length the join is valid; Therefore, click on the **YES** command button. A join line between the two tables is created (**Figure 30**). Now Microsoft Query will select only those records that have the same value in the joined fields in both tables (**Figure 31**).

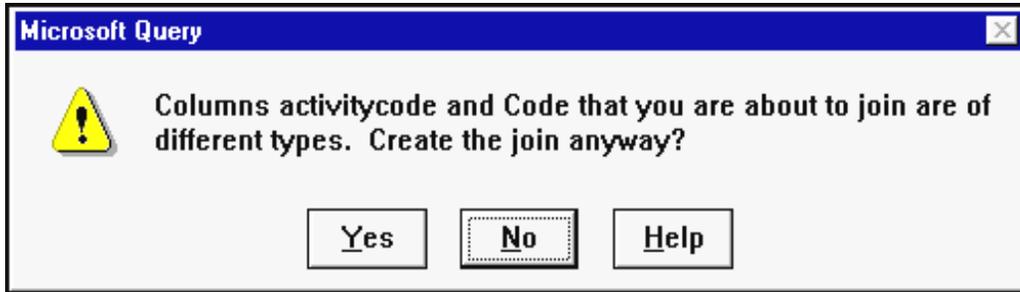


Figure 29 - Microsoft Query warning box.

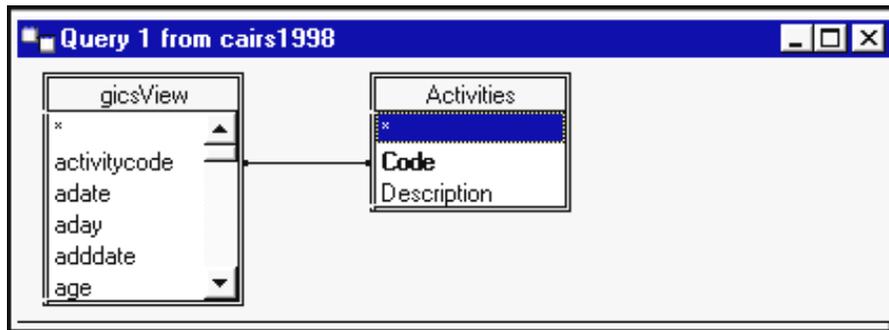


Figure 30 - The TABLE PANE displaying a table join between the **ACTIVITYCODE** and **CODE** fields.

Query 1 from cairs1998

gicsView: activitycode, adate, aday, adddate, age

Activities: Code, Description

Criteria Field: fo, currentflag, ayear
 Value: '30', 'Y', '1997'

Description	Number	LWD	Average LWD
ACTIVITY	2	79	39.500000
BUILDING/EQUIP MAINT/REPAIR ACTIVITY	9	48	5.333333
CONSTRUCTION ACTIVITY	3	32	10.666666
EMERGENCY RESPONSE ACTIVITY	2	3	1.500000
EQUIPMENT INSTALLATION ACTIVITY	1	0	.000000
FOOD SERVICE ACTIVITY	3	17	5.666666
GROUNDS MAINTENANCE ACTIVITY	2	0	.000000
JANITORIAL/HOUSEKEEPING ACTIVITY	14	81	5.785714
LONG-TERM SURVEILLANCE & MAINTENANCE	1	0	.000000
MATERIAL HANDLING ACTIVITY	24	423	17.625000
OFFICE ACTIVITY	38	46	1.210526
OTHER NON-TASK ACTIVITY	3	0	.000000
PHYSICAL FITNESS TRAINING ACTIVITY	1	0	.000000
PRODUCTION/OPERATION ACTIVITY	95	603	6.347368

Record: 20

Figure 31 - Example of a query with a “join” between two tables.

Sometimes tables joins are more complex than the example above. For example, the result set from *Exercise 7* would be more meaningful if the Field Office names were included as a column in the result set. In this example you would need to use the lookup table for Field Office codes (i.e., table **OFFICE**). You add the **OFFICE** table to the **TABLE PANE** by selecting **Add Tables** from the **TABLE** menu to open the **ADD TABLES** dialog box. Select **office** from the **TABLE** selection box and then click on the **ADD** command button. Click on the **CLOSE** command to close the dialog box.

The fields that we need to join are **FO** from the **GICSVIEW** table and **ORG** from the **OFFICE** table. Both are of type varchar and so Microsoft Query will allow us to create a join without issuing an warning message; however, no records are returned in the result set (**Figure 32**). The result set is empty because the **FO** contains a two character code, whereas **ORG** contains a seven character code (the first two characters being the field office code followed by five zeros).

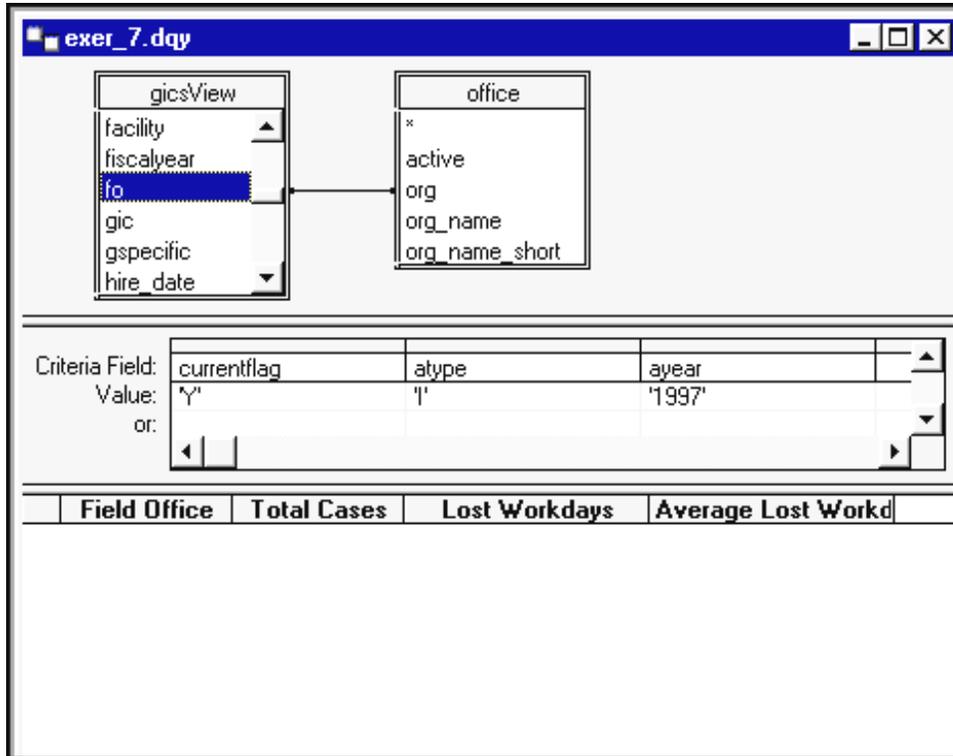


Figure 32 - Example of a query with a join between two tables that returns no records in the result set.

You can join fields that do not match exactly by modifying the SELECT statement. Select **SQL** from the **VIEW** menu to open the **SQL** dialog box. You will need to modify the join predicate in the WHERE clause from *gicsView.fo=office.org* to *(gicsView.fo=substring(office.org,1,2))* (**Figure 33**). This join predicate compares **FO** to the first two characters of **ORG**. Click on the **OK** command button to query with the modified SELECT statement. Microsoft Query graphically displays this modification by deleting the join line and adding an additional criteria to the **CRITERIA** pane (**Figure 34**). To display the Field Office names, you highlight the Field Office code column and press the **[Delete]** key. Select **Insert Column** from the records menu display the **INSERT COLUMN** dialog box. Select **office.org_name** from the **FIELD** selection box, type **Field Office** in the **COLUMN HEADING** edit box, and click on the **INSERT** command button (**Figure 34**).

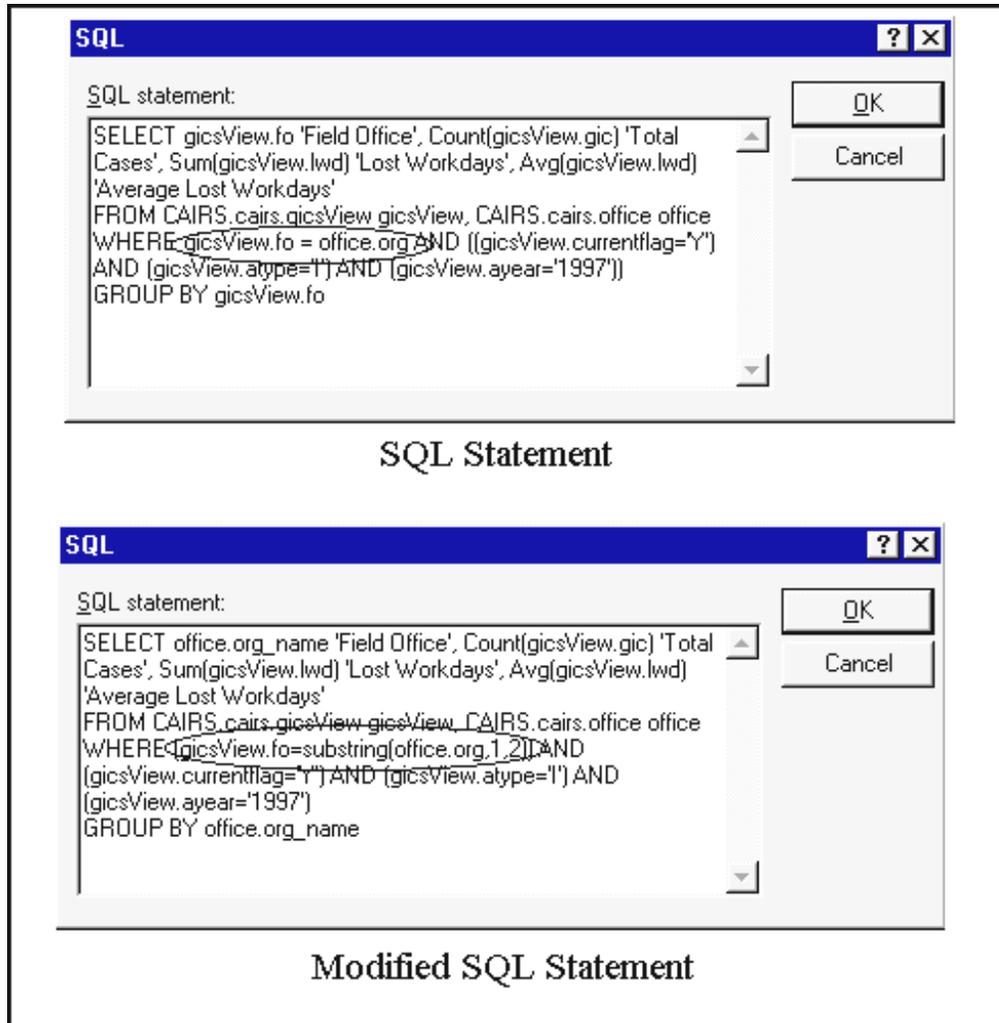


Figure 33 - Example of the modification of a join predicate in the WHERE clause of a SELECT statement.

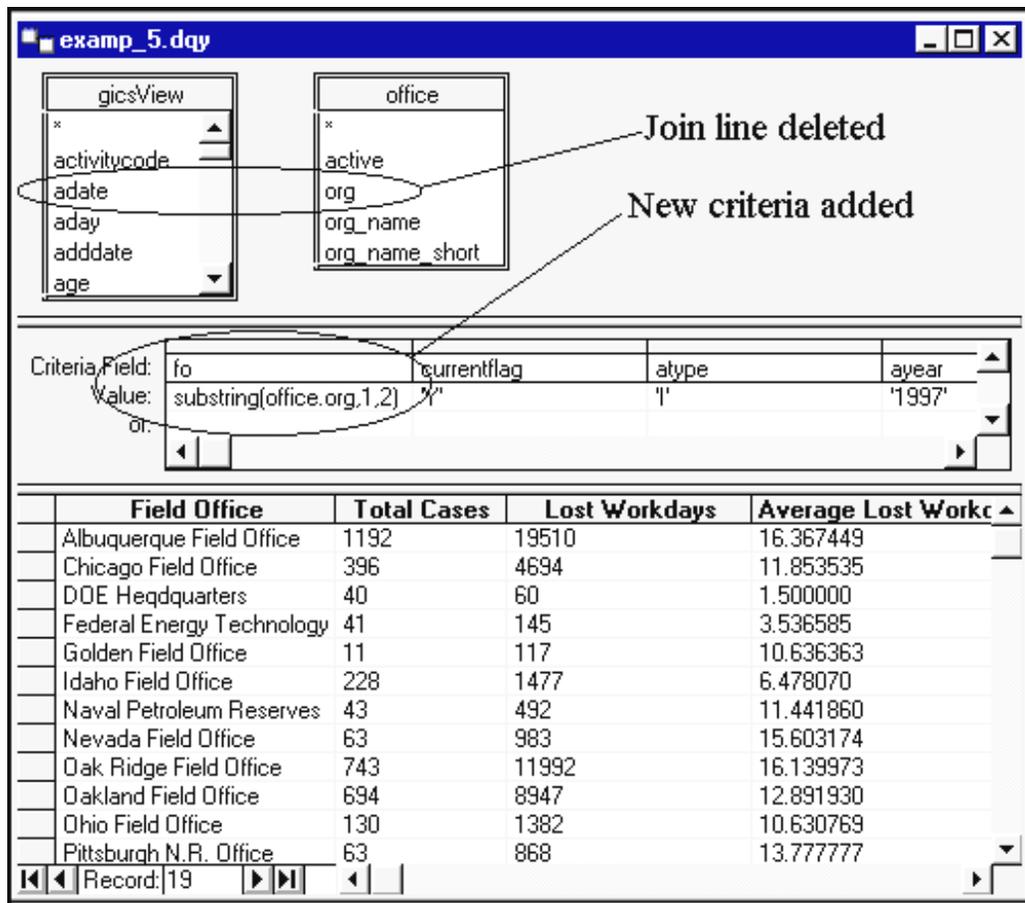


Figure 34 - Example of how Microsoft Query graphically displays modifications to a join predicate.

In most cases you will be joining the **GICSVIEW** table to another database table. The table below shows the fields that should be joined between the **GICSVIEW** table and another table to ensure optimum database performance and correct record retrieval.

Table Name	Recommended Joins to the GICSVIEW table
accident	Join gicsView.gic to accident.GIC and Join gicsView.timestamp to accident.TIMESTAMP
actions	Join gicsView.gic to actions.GIC and Join gicsView.timestamp to actions.TIMESTAMP
actions_reqd	Join gicsView.gic to actions_reqd.GIC and Join gicsView.timestamp to actions_reqd.TIMESTAMP
Activities	Join gicsView.activitycode to activities.Code
activity	Join gicsView.gic to activity.GIC and Join gicsView.timestamp to activity.TIMESTAMP

Table Name	Recommended Joins to the GICSVIEW table
allnarrative	Join gicsView.gic to allnarrative.GIC and Join gicsView.timestamp to allnarrative.TIMESTAMP
Bodyparts	Join gicsView.bodypartii to Bodyparts.Code
causes	Join gicsView.gic to causes.GIC and Join gicsView.timestamp to causes.TIMESTAMP
conditions	Join gicsView.gic to conditions.GIC and Join gicsView.timestamp to conditions.TIMESTAMP
events	Join gicsView.gic to events.GIC and Join gicsView.timestamp to events.TIMESTAMP
factors	Join gicsView.gic to factors.GIC and Join gicsView.timestamp to factors.TIMESTAMP
IIType	Join gicsView.iitype to IIType.Code
LossEvents	Join gicsView.losseventcode to LossEvents.Code
materials	Join gicsView.gic to materials.GIC and Join gicsView.timestamp to materials.TIMESTAMP
Objects	Join gicsView.eq1_g to Objects.Code or Join gicsView.eq2_g to Objects.Code or Join gicsView.dmg_target to Objects.Code
Occupations	Join gicsView.occup to Occupations.Code
office	In the WHERE clause of the SELECT statement include: (gicsView.fo=substring(office.org,1,2))
Opertype	Join gicsView.optype to Opertype.Code
orgd	Join gicsView.org to orgd.ORG
PropLossType	Join gicsView.pcode to PropLossType.Code
ProtectionEq	Join gicsView.ppecodes to ProtectionEq.Code
VehType	Join gicsView.vcode to VehType.Code

Exercise 8

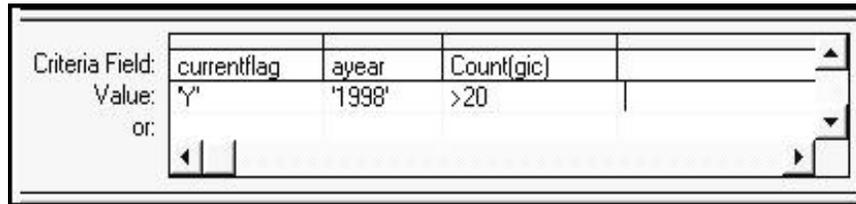
As you work, write down the steps you take. These notes will act as a reminder when you take similar actions in the future.

1. Specify criteria to select the latest version of all 1998 injury/illness.
2. Display the loss producing event description, loss producing event code, and number of cases, workdays away (WDL), and workdays restricted (WDLR) for each loss-producing event. Finally, display a column that calculates the average lost workday (LWD).
3. Order the result set, in ascending order, by the loss-producing event code.
4. Save the query.

Placing Restrictions on Groups

CAIRS is rich with quantitative data that characterizes both the frequency and severity of injury/illness experience. Microsoft Query allows you to take full advantage of this data by allowing you to set frequency and severity limits. You set the limits by placing restrictions on groups within the HAVING clause. For example, you could restrict the groupings in *Exercise 8* to only those groups that account for more than 20 cases and 100 days away from work.

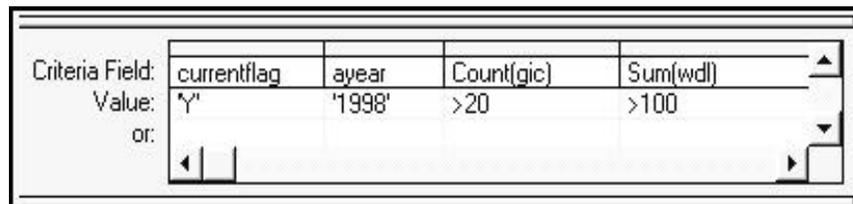
To specify the frequency restriction, select **Add Criteria** from the **CRITERIA** menu to open the **ADD CRITERIA** dialog box. Select **Count** from the **TOTAL** selection box, **gicsView.gic** from the **FIELD** selection box, and **is greater than** from the **OPERATOR** selection box. You define the frequency limit from the **VALUE** edit box by typing **20**. Click on the **ADD** command button to add this criteria to your query (**Figure 35**).



Criteria Field:	currentflag	ayear	Count(gic)	
Value:	'Y'	'1998'	>20	
or:				

Figure 35 - Example of criteria that restricts groups based on frequency.

Similarly, to specify the severity restriction, select **Sum** from the **TOTAL** selection box, **gicsView.wdl** from the **FIELD** selection box, and **is greater than** from the **OPERATOR** selection box. You define the severity limit from the **VALUE** edit box by typing **100**. Click on the **ADD** command button to add this criteria to your query (**Figure 36**).



Criteria Field:	currentflag	ayear	Count(gic)	Sum(wdl)
Value:	'Y'	'1998'	>20	>100
or:				

Figure 36 - Example of criteria that restricts groups based on both frequency and severity.

Now your result set includes only the loss-producing event categories that account for more than 20 cases and 100 days away from work (**Figure 37**).

	Events Description	Events Code	Number	Workdays Away	Workdays Restricted	Avg LWD
▶	Struck against stationary object	0103	200	197	204	2.005000
	Struck by falling object	0202	113	146	364	4.513274
	Compressed or pinched by rolling, slidin	0303	107	233	683	8.560747
	Fall down stairs or steps	1102	32	136	153	9.031250
	Fall to floor, walkway, or other surface	1302	198	716	1209	9.722222
	Fall onto or against objects	1303	48	245	255	10.416666
	Bending, climbing, crawling, reaching, t	2102	235	1050	2592	15.497872
	Running--without other incident	2104	37	151	469	16.756756
	Slip, trip, loss of balance--without fall	2106	121	342	907	10.322314
	Bodily reaction, nec	2109	22	187	382	25.863636
	Overexertion in lifting	2202	289	971	2590	12.321799
	Overexertion in pulling or pushing objec	2203	126	536	1287	14.468253
	Overexertion in holding, carrying, turnin	2204	150	486	1800	15.240000
	Typing or key entry	2302	388	239	1403	4.231958
	Repetitive use of tools	2303	48	194	419	12.770833
	Repetitive placing, grasping, or moving	2304	42	206	493	16.642857
	Repetitive motion, nec	2309	46	149	239	8.434782
	Inhalation of substance, unspecified	3411	29	129	421	18.965517
	Transportation accident, unspecified	4001	22	115	80	8.863636

Figure 37 - Example of a result set where groupings have been restricted by both frequency and severity.

Exercise 9

As you work, write down the steps you take. These notes will act as a reminder when you take similar actions in the future.

1. Specify criteria to select the latest version of all 1998 injury/illness.
2. Display the injury type description, body part injured description, and number of cases and average lost workdays (LWD) per injury type/body part injured grouping.

HINT



Occasionally Microsoft Query will automatically create a join line between tables that is not necessary. To delete this line, click on the line and press the *[Delete]* key.

3. Place a frequency limit on the groups of 40 or more cases and 5 or more average lost workdays. (You may want to change the limit values a few times and view the results.)
4. Save the query.

Setting Query Parameters

Microsoft Query allows you to create queries that accept user-defined values when the query is executed. These user-defined values are called query parameters. We can use the saved query from *Exercise 5* to demonstrate the process. You retrieve the file by selecting **Open** from the **FILE** menu and double-clicking on the file name (e.g., exer_5.dqy). The query is displayed in the query window (**Figure 38**).

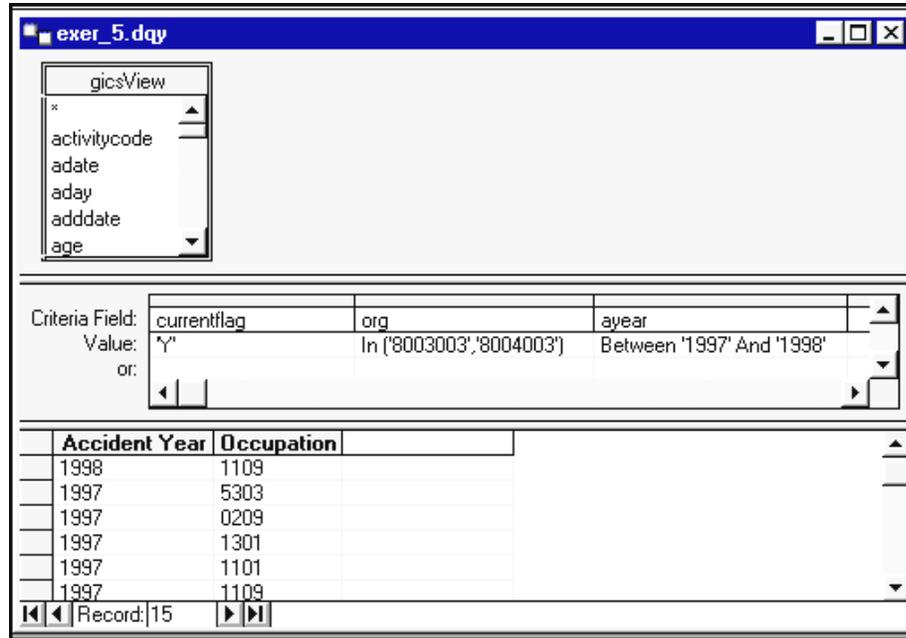


Figure 38 - The query window before setting a user-defined parameter.

The automatic query feature must be disabled before you set a parameter. Disable automatic query by clicking on the **AUTO QUERY** tool bar button so that it is no longer pressed in. You set parameters from the **CRITERIA PANE**. Click on the value cell below the **AYEAR** criteria field cell. Make sure the value statement “Between ‘1997’ and ‘1998’” is highlighted and press the **[Delete]** key. Type **[Enter a Year]** in the cell (**Figure 39**). The left and right square brackets indicate that this is a parameter field.

NOTE



The text of the prompt (i.e., the words you type between the square brackets) must be different from the field name, however it can include the field name.

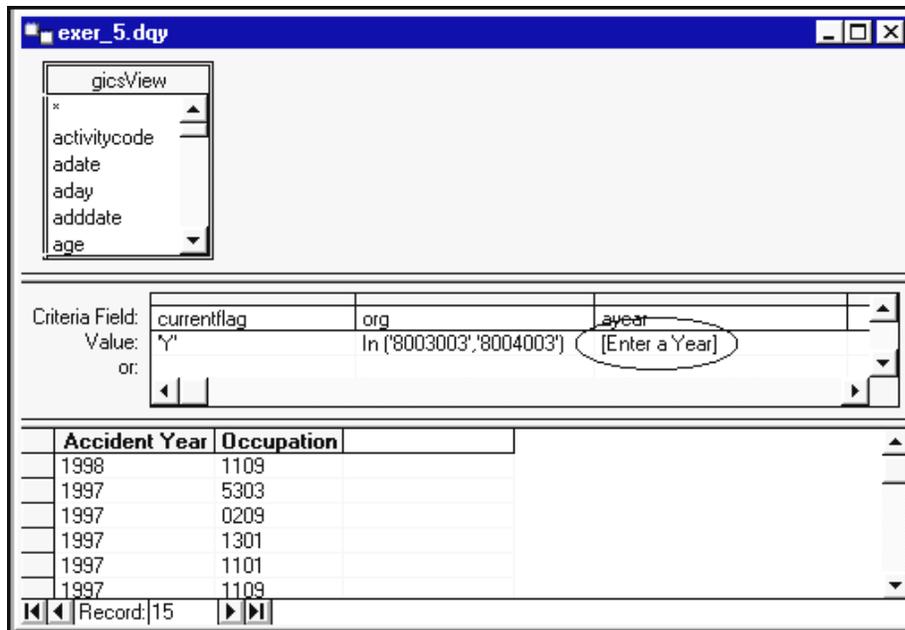


Figure 39 - The query window after setting a user-defined parameter.

To run the query, click on the **QUERY NOW** tool bar button to display the **ENTER PARAMETER VALUE** dialog box (**Figure 40**). Click in the **ENTER A YEAR** text box and type **1996**. Click on the **OK** command button and the **DATA PANE** is recreated with new result set.

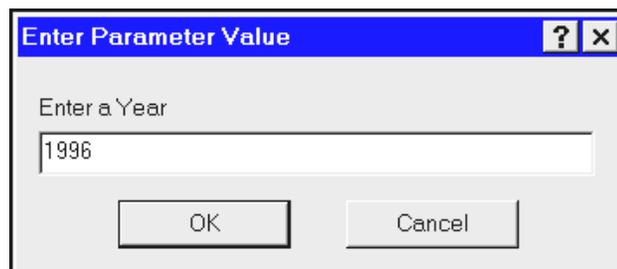


Figure 40 - The **ENTER PARAMETER VALUE** window.



Microsoft Query 2.0 under Office 95, does not support query parameters.



Step-by-Step

Creating a User-Defined Parameter

TUTORIAL

1. Click on the **AUTO QUERY** tool bar button so it is no longer pressed in.
2. If needed, display the **CRITERIA PANE** by clicking on the **SHOW/HIDE CRITERIA** tool bar button.
3. Click on the first empty cell in the **CRITERIA FIELD** row, and then click the arrow in the cell to select a field from the list.
4. Click on the empty cell below the field you selected in the **VALUE** row.
5. Type [(a left square bracket) followed by the text for the prompt you want Microsoft Query to display when the query is run, and then type] (a right square bracket).
6. To run the query, click on the **QUERY NOW** tool bar button.

Exercise 10

As you work, write down the steps you take. These notes will act as a reminder when you take similar actions in the future.

1. Specify criteria to select the latest version of all cases for a user-defined field office, accident type, and accident year.
2. Display the key, accident date, reporting organization, and one-line description.
3. Run the query.
4. Save the query file.

Modifying a Query's Design

NOTE



The information below describes how to modify a query using the Microsoft Query **TABLE PANE**, **CRITERIA PANE**, and **DATA PANE**. You can also modify a query from the **SQL** dialog box by making changes to the **SQL SELECT** statement.

Opening a Query

A saved query is opened from the **OPEN QUERY** dialog box (**Figure 41**). To open the dialog box, select **Open** from the **FILE** menu. To open a query, click on the query you want (it will be a .dqy or a .qry file) and then click on the **OPEN** command button.

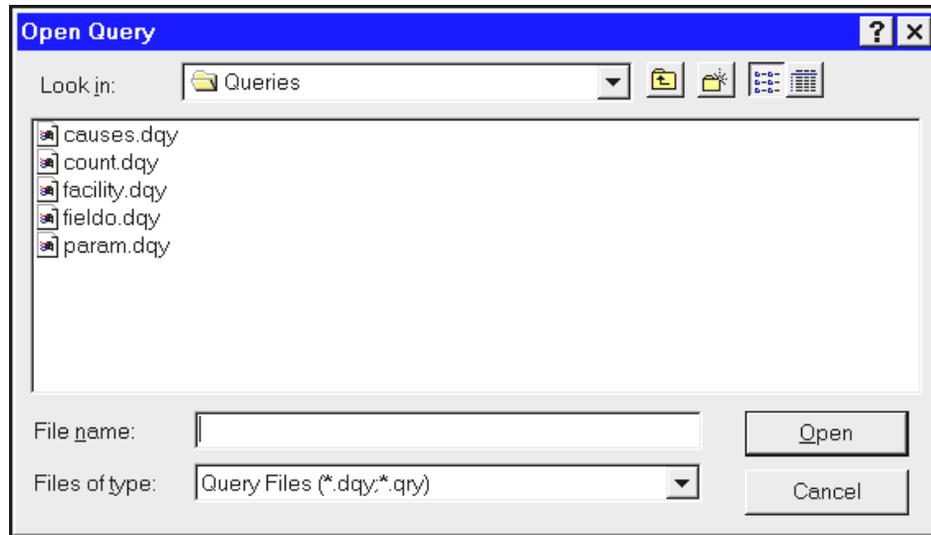


Figure 41 - The **OPEN QUERY** dialog box.

If you have not already established a connection to CAIRS, the **SQL SERVER LOGIN** dialog box will appear (**Figure 42**). Enter your logon name and password and click on the **OK** command button. The query file is retrieved by Microsoft Query and displayed in the query window.

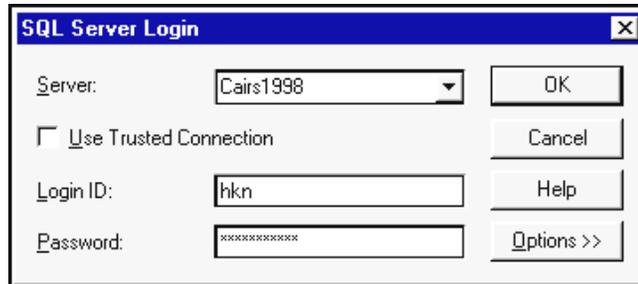
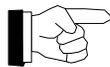


Figure 42 - The **SQL SERVER LOGIN** dialog box.

NOTE



If the query contains user-defined parameters, you will be prompted for the parameter values.

HINT



A quick way to open the **OPEN QUERY** dialog box is to click on the **OPEN QUERY** tool bar button.

Step-by-Step

TUTORIAL

Opening a Saved Query File

1. Select **Open** from the **FILE** menu.
2. Locate and select the query you want to open and click on the **OPEN** command button.
3. If the **SQL SERVER LOGIN** dialog box appears, enter your user name and password, then click on the **OK** command button.
4. If prompted, enter user-defined parameters.



You cannot open .qry files saved within the Office 95 or Office 97 environment into Microsoft Query 97.



Opening a Saved Query File with Office 95

1. Select **New Query** from the **FILE** menu to open the **SELECT DATA SOURCE** dialog box.
2. Select **CAIRS1998** from the **AVAILABLE DATA SOURCES** list and click on the **USE** command button to open the **SQL SERVER LOGIN** dialog box.
3. Enter your user ID in the **USER NAME** edit box and your password in the **PASSWORD** edit box. Click on the **OK** command button to link to the **CAIRS** database.
4. Select **Open Query** from the **FILE** menu.
5. Locate and select the query you want to open and click on the **OK** command button.



You cannot open .qry files saved within the Office 97 environment with Microsoft Query 95

Modifying Table Information

Table information is modified within the **TABLE PANE**. You can add tables, add table joins, delete tables, and delete table joins. The steps for adding tables or joining tables for a previously saved query are the same as those described in the *Adding and Joining Tables* section of this guide.

To delete a table, click on the table you want to delete and select **Remove Table** from the **TABLE** menu. The table is deleted from the **TABLE PANE**. This action automatically deletes any joins associated with that table. In addition, any criteria on the **CRITERIA PANE** or column on the **DATA PANE** which came from the delete table are automatically removed (**Figure 43**).

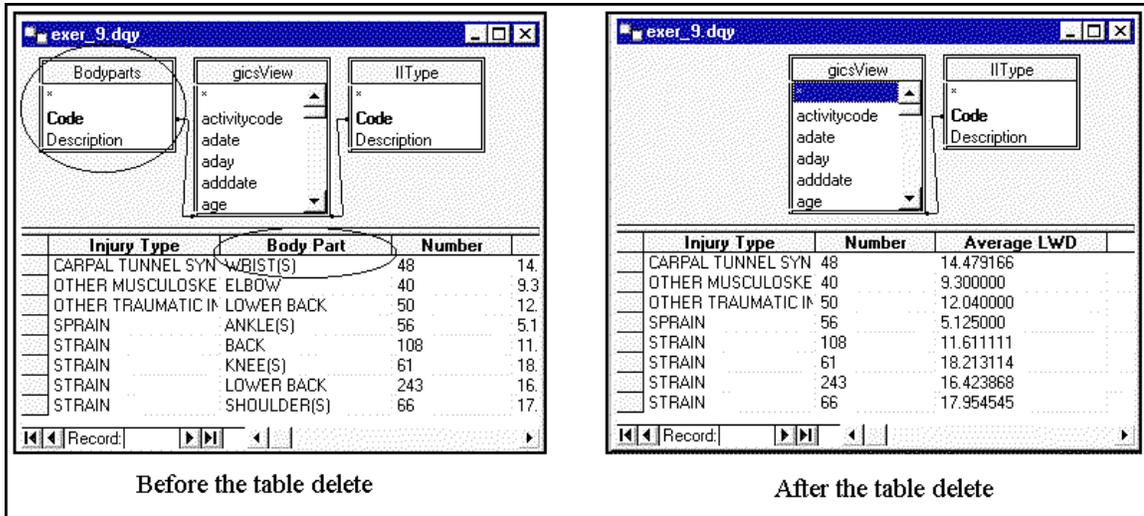


Figure 43 - The TABLE PANE and the DATA PANE before and after the deletion of the BODYPARTS table.



An easy way to delete a table is to click on the table and then press the *[Delete]* key.

You can also delete a table join without deleting a table. Single-click on the join line to display the line as a heavy line (**Figure 44**), then press the *[Delete]* key.

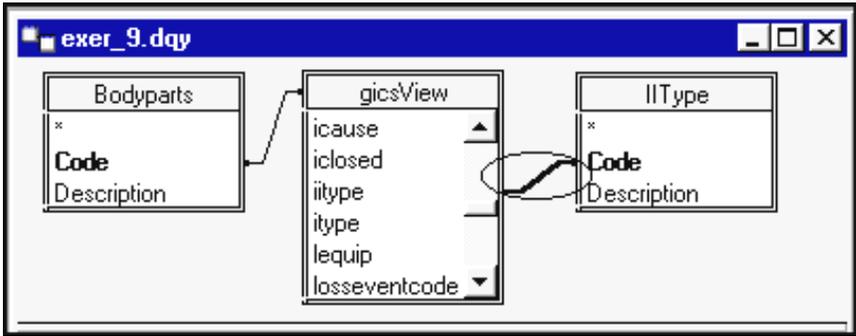


Figure 44 - A table join that has been selected for deletion.



If you delete a table join and then execute the query before you create another join line or delete the table, the Cartesian product of the unjoined table and the other tables displayed in the TABLE PANE will be returned to the DATA PANE.

Modifying Criteria Information

You modify the criteria for the query from the **CRITERIA PANE**. You can add criteria, delete criteria, and change criteria fields or values. The steps for adding criteria to a previously saved query are the same as those described in the *Specify the Criteria* section of this guide.

To delete all criteria, select **Remove All Criteria** from the **CRITERIA** menu. All criteria is removed from the **CRITERIA PANE**. To delete a single criterium, click at the top of a criterium column to highlight the entire column (**Figure 45**), then press the *[Delete]* key. The highlighted criterium column is removed from the **CRITERIA PANE** (**Figure 45**).

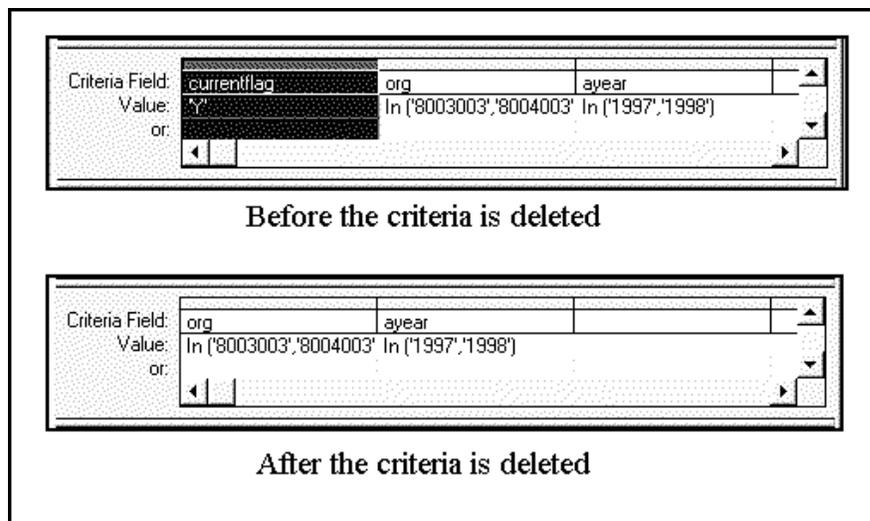


Figure 45 - The **CRITERIA PANE** before and after the deletion of the **CURRENTFLAG** criterium.

To change a criteria field, double-click on the field you want changed in the **CRITERIA FIELD** row. An **EDIT CRITERIA** dialog box appears and displays the current field and field total selections (**Figure 46**). You select a replacement field using the **FIELD** selection box. You can also redefine the field total using the **TOTAL** selection box. Click on the **OK** command button to replace the current selection.



Figure 46 - The **EDIT CRITERIA** dialog box used to change fields.



Another way to change a field selection is to single-click on the field you want changed in the **FIELD CRITERIA** row. The field name is highlighted and a down arrow () appears in the cell. Click on the down arrow to display a field selection box, then click on the field you want as the criteria field. The new field name replaces the previous selection.

To change a field value, double-click on the value you want changed in the **VALUE** row. An **EDIT CRITERIA** dialog box appears and displays the current field, field total selections, operator, and value (**Figure 47**). (Notice that only the operator and value can be changed.) You enter a replacement value into the **VALUE** selection box. Alternately, you could click on the **VALUES** command button and select a replacement value. You can also redefine the operator using the **OPERATOR** selection box. Click on the **OK** command button to replace the current selection.

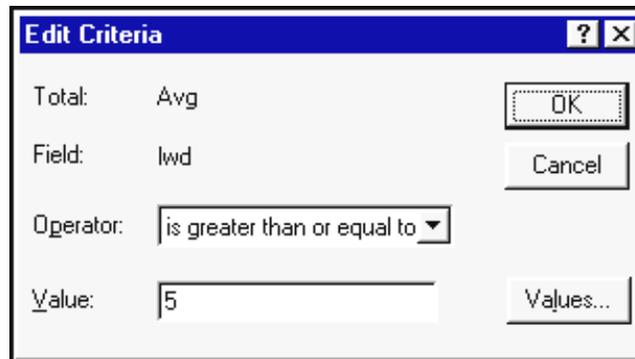


Figure 47 - The **EDIT CRITERIA** dialog box used to change field values.



HINT

If you know the correct syntax and value(s), you can type the operator and value(s) directly into the value cell.

Modifying Report Columns

You modify the columns displayed in your report from the **DATA PANE**. You can remove, insert, edit, or move columns. To remove a column, click anywhere on the column you want to remove and then select **Remove Column** from the **RECORDS** menu. The column is deleted from the **DATA PANE**. You can also remove a column by single-clicking on the column heading to highlight the entire column and then press the **[Delete]** key.

To insert a column, single-click on the column heading of the column to the right of where you want the new column inserted. This will highlight the entire column. Select **Insert Column** from the **RECORDS** menu to open the **INSERT COLUMN** dialog box (**Figure 48**). Specify the field, column headings (optional), and totals (optional) and then click on the **INSERT** command button. The new column is inserted to the left of the column that you highlighted (**Figure 49**).

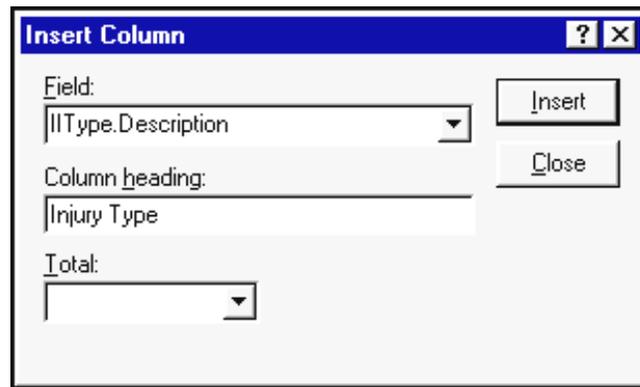


Figure 48 - The **INSERT COLUMN** dialog box.

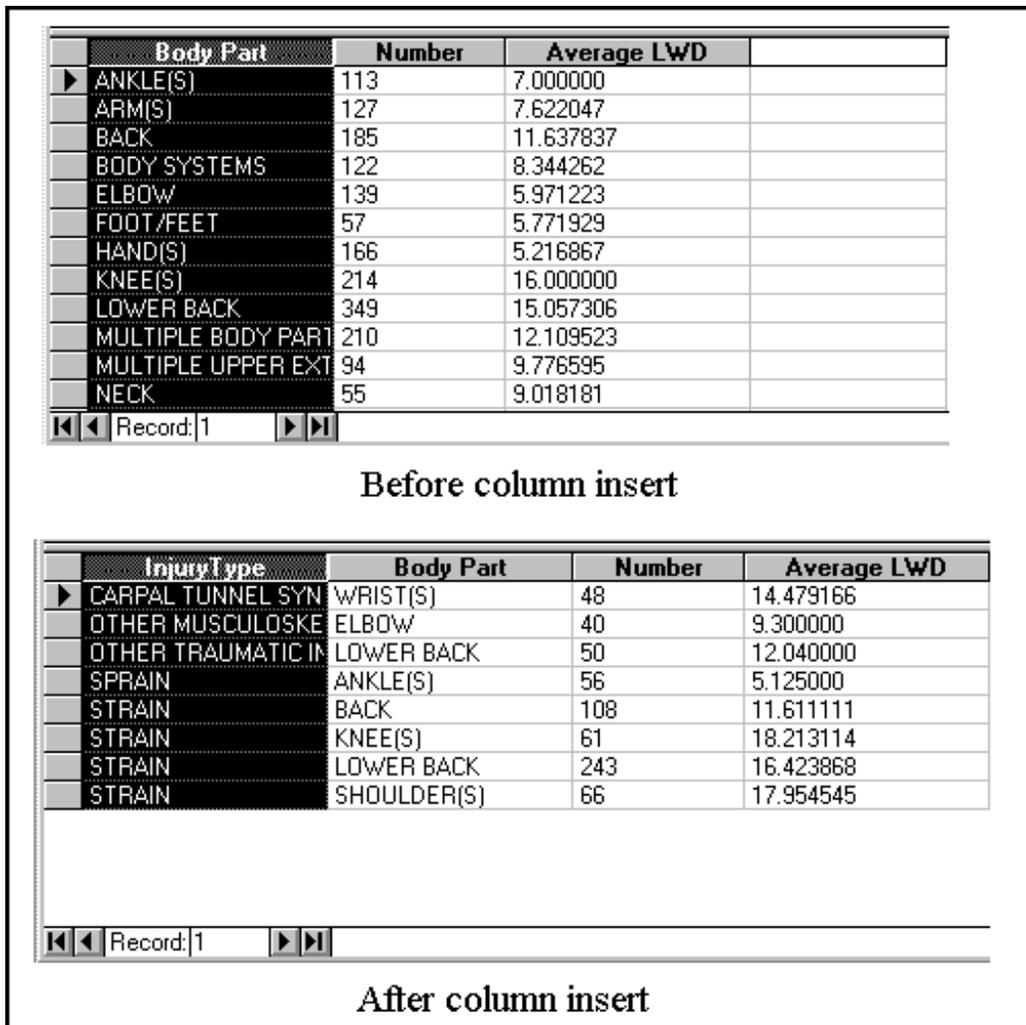


Figure 49 - The DATA PANE before and after the insertion of the direct cause description column.

To edit a report column, click anywhere on the column you want to edit and then select **Edit Column** from the **RECORDS** menu. The **EDIT COLUMN** dialog box appears and displays the current field, column heading, and total selections (**Figure 50**). You can edit the column by changing the field in the **FIELD** selection box, the heading in the **COLUMN HEADING** edit box, or the total in the **TOTAL** selection box. Click on the **OK** command button to save and display the changes in the **DATA PANE** (**Figure 51**).



A quick way to launch the **EDIT COLUMN** dialog box is to double-click on the column heading of the column you want to edit.

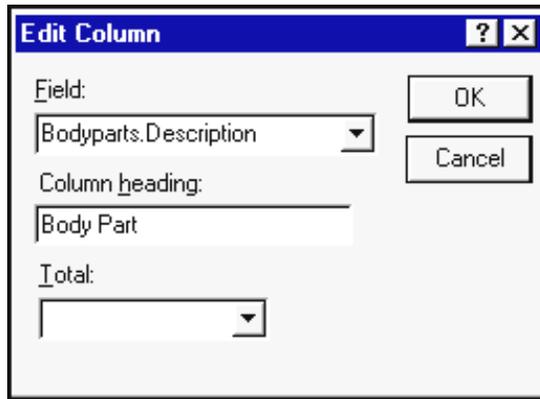


Figure 50 - The EDIT COLUMN dialog box.

	InjuryType	Description	Number
▶	CARPAL TUNNEL SYN	WRIST(S)	48
	OTHER MUSCULOSKE	ELBOW	40
	OTHER TRAUMATIC IN	LOWER BACK	50
	SPRAIN	ANKLE(S)	56
	STRAIN	BACK	108
	STRAIN	KNEE(S)	61
	STRAIN	LOWER BACK	243
	STRAIN	SHOULDER(S)	66

Record: 1

Before column edit

	InjuryType	Body Part	Number
▶	CARPAL TUNNEL SYN	WRIST(S)	48
	OTHER MUSCULOSKE	ELBOW	40
	OTHER TRAUMATIC IN	LOWER BACK	50
	SPRAIN	ANKLE(S)	56
	STRAIN	BACK	108
	STRAIN	KNEE(S)	61
	STRAIN	LOWER BACK	243
	STRAIN	SHOULDER(S)	66

Record: 1

After column edit

Figure 51 - The DATA PANE before and after editing the description column.

To move a report column, single-click on the column heading of the column you want to move. Move your mouse pointer over the column heading, and then drag the column to the place you want.

Exercise 11

As you work, write down the steps you take. These notes will act as a reminder when you take similar actions in the future.

1. Open the file saved in **Exercise 5**.
2. Modify the criteria to accept a user-defined value for organization.
3. Change the **OCCUPATION** column to display the description of occupation.



HINT

You will need to modify the **TABLE PANE** to include the **OCCUPATIONS** table.

4. Add a report column that displays the number of reports per occupation.

Displaying and Formatting the Result Set

Displaying Report Columns

Microsoft Query allows you to determine which columns from the result set are displayed on the **DATA PANE**. To set the column display property, select **Show Columns** from the **FORMAT** menu to open the **SHOW COLUMNS** dialog box (**Figure 52**). All columns in the result set are listed in the **COLUMNS** list box. If a column name is checked, it is displayed on the **DATA PANE**; Similarly, if a column name is not checked, it is not displayed on the **DATA PANE**. To remove a check from a column name (hide the column), select the column and then click on the **HIDE** command button. To add a check to a column name (show a column), select the column and then click on the **SHOW** command button. After you set the display property for the columns in the result set, you close the dialog box by clicking on the **CLOSE** command button.



A quick way to hide columns is to click on the column you want to hide and then select **Hide Columns** from the **FORMAT** menu.

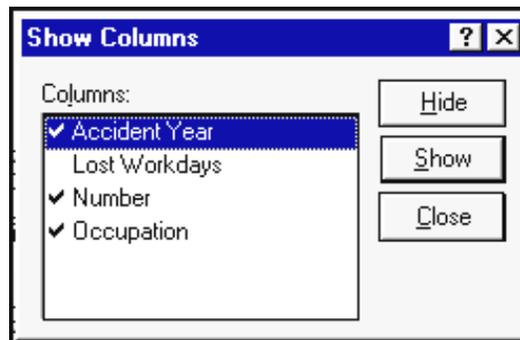


Figure 52 - The **SHOW COLUMNS** dialog box.

Formatting the Report

Microsoft Query allows you to customize the font, row height, and column width for the **DATA PANE**. The report font is set from the **FONT** dialog box (**Figure 53**). To open the dialog box, select **Font** from the **FORMAT** menu. After you select the font settings for your report, click on the **OK** command button to activate the settings. Clicking on the **CANCEL** command button will cancel any selections that have been changed.

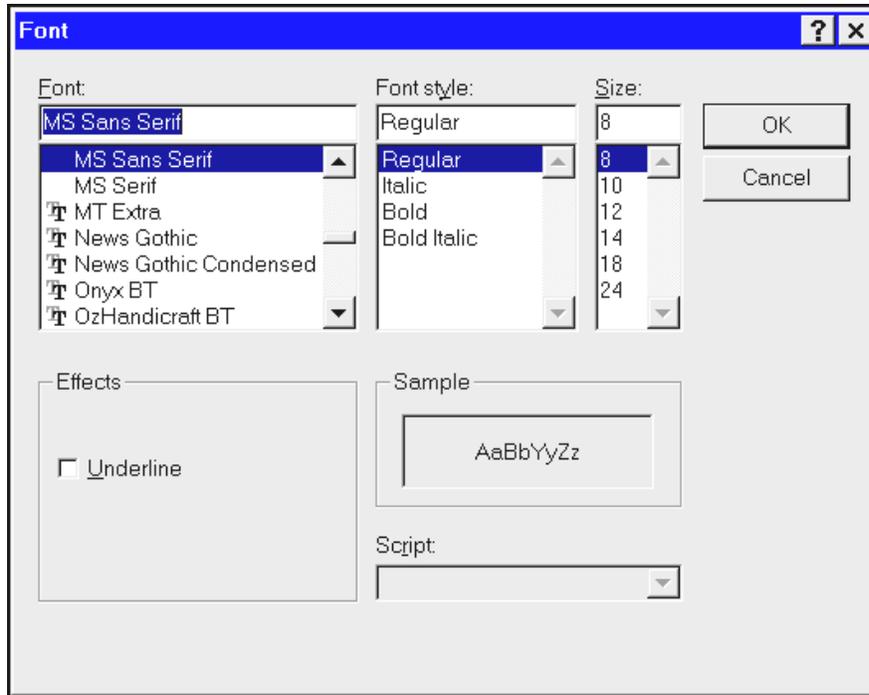


Figure 53 - The FONT dialog box.

Row height is set from the **ROW HEIGHT** dialog box (**Figure 54**). To open the dialog box, select **Row Height** from the **FORMAT** menu. Enter the height you want into the **ROW HEIGHT** edit box and then click on the **OK** command button to activate this setting. Clicking on the **CANCEL** command button will cancel any setting that has been changed.

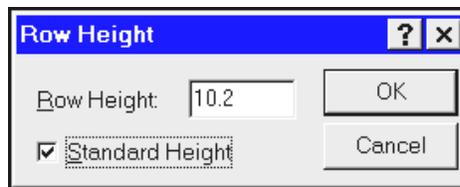


Figure 54 - The ROW HEIGHT dialog box.



If the **STANDARD HEIGHT** check box is checked, Microsoft Query will apply the default height. The default height varies, depending on the font used in the **DATA PANE**.

Column width is set from the **COLUMN WIDTH** dialog box (**Figure 55**). To open the dialog box, select **Column Width** from the **FORMAT** menu. Enter the width you want into the **COLUMN WIDTH** edit box and then click on the **OK** command button to activate this setting. Clicking on the **CANCEL** command button will cancel any setting that has been changed.

NOTE



If the **STANDARD WIDTH** check box is checked, Microsoft Query will apply the default width. The default width varies, depending on the font used in the **DATA PANE**.

NOTE



If you click on the **BEST FIT** command button, the column width of the selected column will be adjusted automatically to the largest value displayed on the **DATA PANE**.

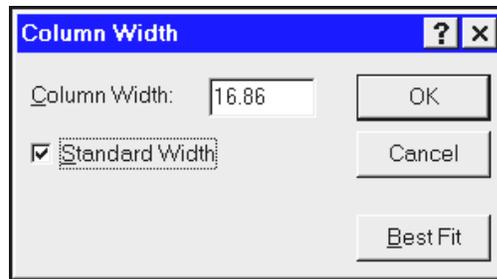


Figure 55 - The **COLUMN WIDTH** dialog box.

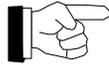


Formatting properties are not saved with a query. Therefore, you will need to reset any customized formatting every time a query is opened.

CAIRS Direct Access Templates

DOE is developing a number of templates that will assist you in creating your own queries of the CAIRS database. You can run these templates directly from Microsoft Query or you can execute them from within Microsoft Excel or another application that provides additional computational and formatting capabilities. In addition, you can modify them to add other selection criteria or fields.

NOTE



The templates that are presently being developed will run under Microsoft Office 97 (.dqy). In the future, these will be converted to run under Office 95 (.qry) as well. In the meantime, you can copy the SQL statements into the query window in Query 95 to create a new saved query.

In general, the templates demonstrate some of the more advanced queries of the CAIRS database. You should be aware that, although the queries will run under Microsoft Query, many elements of the SQL statement cannot be created directly by using the selection processes in the Query. The SQL statements were generally typed directly into the query window. However, if the additional criteria or columns that you wish to add can be directly entered, you can use the selection processes to modify the query without losing the special query features.

The Templates

Templates can be downloaded from the Internet at this URL:

XXXXXXXXXXXXXXXXXXXXXXX

Templates that are currently available are described in the table below. Example reports are shown in **Appendix C**.

File Name	Query Description

Executing a Template from Microsoft Excel

To execute a template from Microsoft Excel, start Excel and open a new worksheet. From the **DATA** menu select **Get External Data** and then select **Run Database Query** as shown in **Figure 56**.

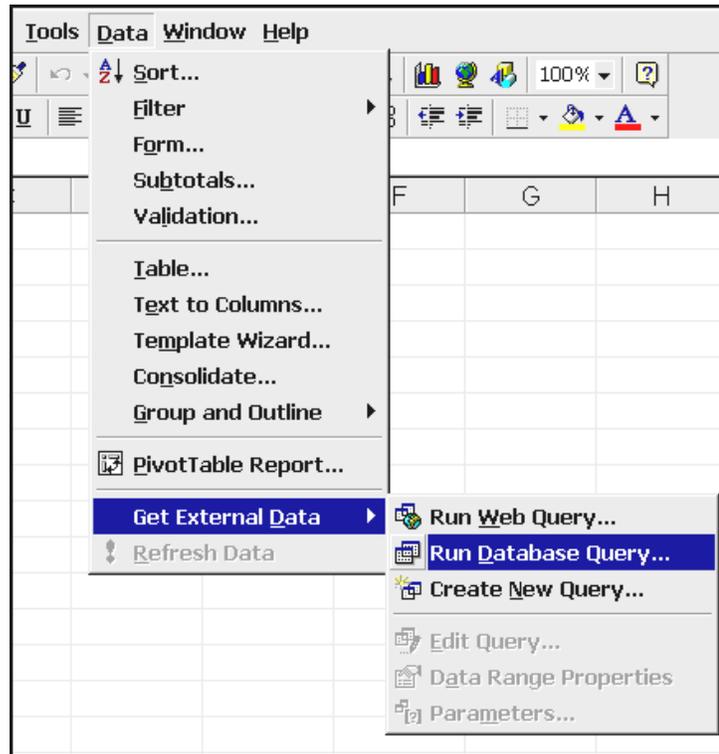


Figure 56 - The Microsoft Excel **DATA** menu.

This will open the **RUN QUERY** dialog box as shown in **Figure 57**. If your saved queries are located in the default directory, they will appear in the dialog box. If they were saved elsewhere, you must locate them using the **LOOK IN** selection box. Click on the query you wish to use to highlight it and then click on the **GET DATA** command button.

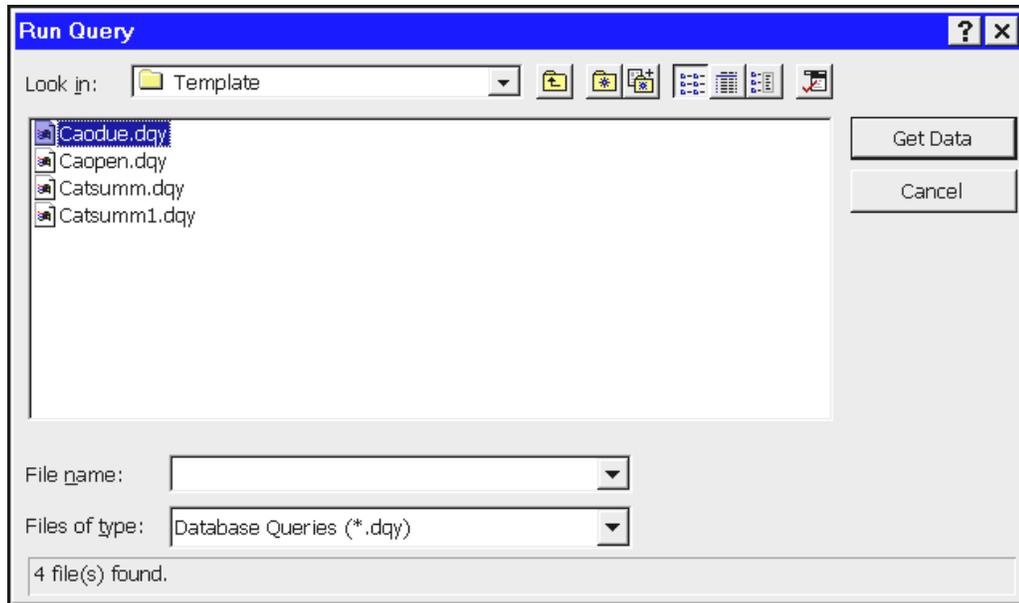


Figure 57 - The **RUN QUERY** dialog box.

The **RETURNING EXTERNAL DATA** dialog box will next be displayed as shown in **Figure 58**. At this point, you could simply accept the default settings and return the data to Excel by clicking on the **OK** command button. You will be prompted to enter values for any parameters saved with the query and the data will be returned directly into Excel.

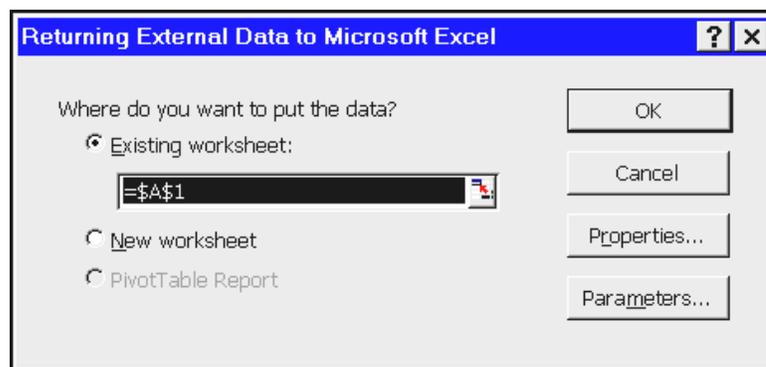


Figure 58 - The **RETURNING EXTERNAL DATA** dialog box.

There are a number of options that you can use to further specify the manner in which the data is returned. By default, the data will be located with the upper left corner of the data table positioned at the current location of your cursor. For a new Worksheet, this will normally be cell \$A\$1. If you wish to locate the data in a different position, simply enter the cell location in the dialog box (or remember to position your cursor to the desired location before you initiate the request for external data).

Additional options can also be selected from this dialog box. Clicking on the **PROPERTIES** command button brings up the **EXTERNAL DATA RANGE PROPERTIES** dialog box shown in **Figure 59**. One option of particular interest is the **INCLUDE FIELD NAMES** option. By default, this box is checked and field names will be returned from Microsoft Query. However, the name that is returned is the raw field name, not the column name defined in Microsoft Query. If you are going to be repeatedly importing the same query into Excel, you may wish to save an Excel worksheet with nicely formatted header information and return the data just below the headers. In this case, simply click on the check box to remove the mark, and the data will be returned without the field names.

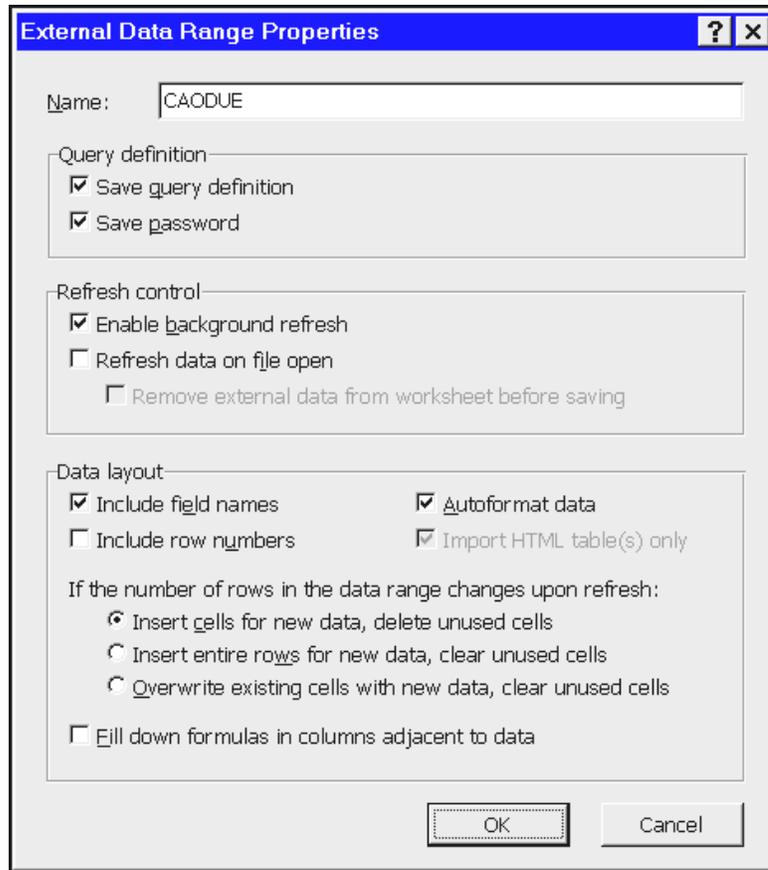


Figure 59 - The **EXTERNAL DATA RANGE PROPERTIES** dialog box.

Clicking on the **PARAMETERS** command button on the **RETURNING EXTERNAL DATA** dialog box opens the **PARAMETERS** dialog box shown in **Figure 60**. This box allows you to prespecify the parameters that will be used for your query. By default, you will be prompted to enter values when you actually execute the query. Alternately, you can enter the values in the dialog box and they will be used when the query is executed.

A third option is also available. Continuing with the previous example where you have saved header information for a report, the saved worksheet could also contain the values for the parameters. If you select the third option and specify the cells in which values are located, the stored worksheet values will be used to execute the query.

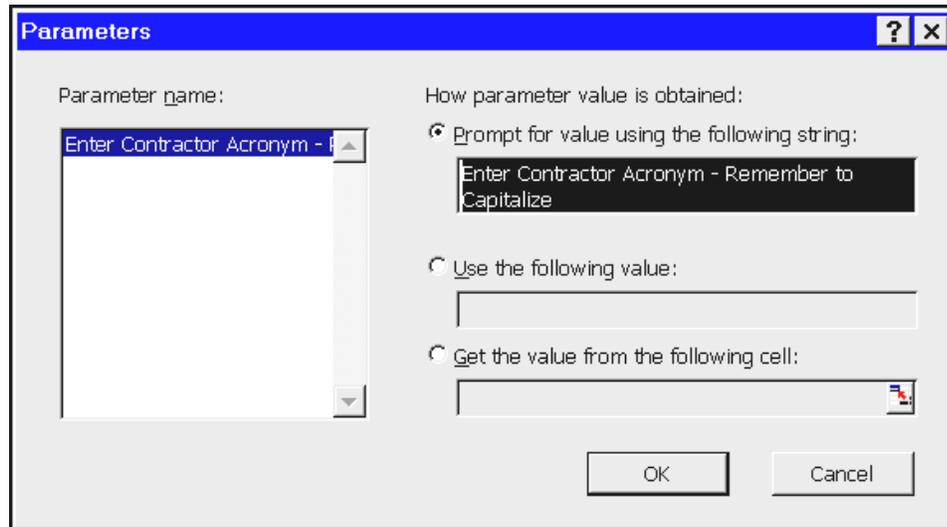


Figure 60 - The EXTERNAL DATA PARAMETERS dialog box.



Step-by-Step

Executing a Database Query from Microsoft Excel 97

TUTORIAL

1. Start Microsoft Excel and open a new worksheet.
2. Select **Get External Data** from the **DATA** menu.
3. Select **Run Database Query** to open the **RUN QUERY** dialog box.
4. Locate and select your query and then click on the **GET DATA** command button to open the **RETURNING EXTERNAL DATA** dialog box.
5. Change (optional) the parameter, properties, or other options displayed on the dialog box.
6. Click on the **OK** command button to open the **SQL SERVER LOGIN** dialog box.
7. Type your user ID in the **USER NAME** edit box and your password in the **PASSWORD** edit box.
8. Click on the **OK** command button to return the result set to Microsoft Excel.



Converting Query Files to Office 95

1. Open the file queries.txt in a text editor such as Notepad.
2. Start Query 95 and select **New Query** from the **FILE** menu to open the **SELECT DATA SOURCE** dialog box.
3. Select **CAIRS1998** from the **AVAILABLE DATA SOURCES** list and click on the **USE** command button to open the **SQL SERVER LOGIN** dialog box.
4. Enter your user ID in the **USER NAME** edit box and your password in the **PASSWORD** edit box. Click on the **OK** command button to link to the ORPS database.
5. Click on the **SQL** button on the toolbar to open the query window.
6. Select the desired SQL statement from queries.txt and copy it to the Clipboard.
7. Paste the SQL statement into the query window.
8. Click on the **OK** button to close the query window and execute the query.
9. Edit the selection parameters to the desired values and save the query as a .qry file.



Executing a Database Query from Microsoft Excel 95

1. Start Microsoft Excel and open a new worksheet.
2. Select **Get External Data** from the **DATA** menu to launch Microsoft Query.
3. Select **CAIRS1998** as your data source and click on the **USE** command button to open the **SQL SERVER LOGIN** dialog box.
4. Enter your user ID in the **USER NAME** edit box and your password in the **PASSWORD** edit box. Click on the **OK** command button.
5. Click the **CLOSE** command button on the **ADD TABLES** dialog box.
6. Select **Open Query** from the **FILE** menu.
7. Locate and select the query you want to execute and click on the **OK** command button.
8. Click on the **RETURN DATA** tool bar button.
9. Set the options displayed in the **GET EXTERNAL DATA** dialog box (optional) and click on the **OK** command button to return the result set to Microsoft Excel.

Appendix A - TEST Database

ACCIDENT

NUMBER	DATE (yyyymmdd)	ORG	INJURY_ TYPE	BODY_ PART	WDLR	WDL
1	19980112	0502009	0012	3101	20	10
2	19980115	0502009	0052	3101	0	0
3	19980131	3003003	0072	4401	0	4
4	19980202	7503409	0012	4102	8	20
5	19980202	7503409	0035	3301	5	0
6	19980217	3003003	0052	0305	0	14
7	19980301	3003003	0012	2301	68	126
8	19980321	3003003	0083	3101	0	0
9	19980322	3003003	0052	3101	0	2
10	19980409	0502009	0035	4102	2	2
11	19980409	7503409	0035	0305	0	1
12	19980505	7503409	0052	3301	0	0
13	19980506	0502009	0062	0101	0	256
14	19980526	3003003	0012	3101	5	5
15	19980611	3003003	0052	4102	10	34
16	19980619	0502009	0035	3101	0	0
17	19980704	0502009	0037	3101	0	0
18	19980707	3506502	1051	2503	67	45
19	19980807	3506502	0035	4102	0	0
20	19980814	3506502	0035	4102	0	6
21	19981010	3506502	0012	3301	45	2
22	19981030	0502009	0037	4401	0	1

ORGANIZATION

CODE	DESCRIPTION
0502009	Albuquerque
3003003	Idaho
3506502	Nevada
7503409	Richland

INJURY

CODE	DESCRIPTION
0012	Fracture
0035	Laceration
0037	Animal Bite
0052	Burn
0062	Concussion
0072	Frostbite
0083	Electric Shock
1051	Hernia

BODY

CODE	DESCRIPTION
0101	Brain
0305	Face
2301	Back
2503	Groin
3101	Arm
3301	Hand
4401	Toe
4102	Leg

Appendix B - Database Schema and Cross Reference

FIELD	DESCRIPTION	TYPE	CAN BE NULL	COMMENTS
Table: accident (report submittal, approval, and responsibility information table)				
ADDDATE	The date the report was first entered into CAIRS.	varchar(8)	YES	Format is yyyyymmdd.
CADATE	Implementation date for recommended corrective action	varchar(8)	YES	Format is yyyyymmdd.
CONTACT	Accident investigation contact	varchar(40)	YES	
CONTACTP	Phone of accident investigation contact	varchar(12)	YES	
GIC	Report number	varchar(8)	NO	
INVEST	Person who completed the form	varchar(40)	YES	
INVESTD	Date of signature of the person who completed the form	varchar(8)	YES	Format is yyyyymmdd.
INVESTP	Phone of person who completed the form	varchar(12)	YES	
INVESTT	Job title of person who completed the form	varchar(40)	YES	
MODDATE	The date the report was last modified.	varchar(8)	YES	Format is yyyyymmdd.
SUPER	Supervisor responsible for corrective actions	varchar(40)	YES	
SUPERD	Date of signature of supervisor responsible for corrective actions	varchar(8)	YES	Format is yyyyymmdd.
SUPERP	Phone of supervisor responsible for corrective actions	varchar(12)	YES	
TIMESTAMP	Record timestamp	varchar(14)	NO	Format is yyyyymmddhhmmss.

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: actions (corrective actions text table)				
CURRENTFLAG	Current record flag (Y/N)	char(1)	YES	
GIC	Report number	varchar(8)	NO	
LINENUM	Line identifier for each 255-character record	numeric(2)	NO	
TEXT	Corrective actions taken text	varchar(255)	NO	
TIMESTAMP	Record timestamp	varchar(14)	NO	
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: actions_reqd (corrective actions recommended text table)				
CurrentFlag	Current record flag (Y/N)	char(1)	YES	
GIC	Report number	varchar(8)	NO	
LINENUM	Line identifier for each 255-character record	numeric(2)	NO	
TEXT	Corrective actions recommended text	varchar(255)	NO	
TIMESTAMP	Record timestamp	varchar(14)	NO	
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: Activities (activity code lookup table)				
Code	Activity code	char(4)	NO	
Description	Activity description	varchar(255)	NO	
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: activity (activity text table)				
CURRENTFLAG	Current record flag (Y/N)	char(1)	YES	
GIC	Report number	varchar(8)	NO	
LINENUM	Line identifier for each 255-character record	numeric(2)	NO	
TEXT	Activity in progress at the time of the accident text	varchar(255)	NO	
TIMESTAMP	Record timestamp	varchar(14)	NO	

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: allnarrative (all narrative text table)				
CurrentFlag	Current record flag (Y/N)	char(1)	YES	
GIC	Report number	varchar(8)	NO	
LINENUM	Line identifier for each 255-character record	numeric(2)	NO	
TEXT	All narrative text	varchar(255)	NO	Includes the following text fields: actions, actions_reqd, activity, causes, conditions, descript, events, factors, materials.
TEXTNAME	Text field column names	varchar(16)	NO	
TIMESTAMP	Record timestamp	varchar(14)	NO	
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: Bodyparts (body part code lookup table)				
Code	Body part affected code	char(4)	NO	
Description	Body part affected description	varchar(255)	NO	
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: causes (employee action text table)				
CURRENTFLAG	Current record flag (Y/N)	char(1)	YES	
GIC	Report number	varchar(8)	NO	
LINENUM	Line identifier for each 255-character record	numeric(2)	NO	
TEXT	Employee actions that contributed to the accident text	varchar(255)	NO	
TIMESTAMP	Record timestamp	varchar(14)	NO	
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: conditions (conditions and causes text table)				
CURRENTFLAG	Current record flag (Y/N)	char(1)	YES	
GIC	Report number	varchar(8)	NO	
LINENUM	Line identifier for each 255-character record	numeric(2)	NO	
TEXT	Conditions and causes text	varchar(255)	NO	
TIMESTAMP	Record timestamp	varchar(14)	NO	

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: events (sequence of events text table)				
CURRENTFLAG	Current record flag (Y/N)	char(1)	YES	
GIC	Report number	varchar(8)	NO	
LINENUM	Line identifier for each 255-character record	numeric(2)	NO	
TEXT	Accident event sequence text	varchar(255)	NO	
TIMESTAMP	Record timestamp	varchar(14)	NO	
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: exposure (vehicle accident and injury/illness exposure table)				
CURRENTFLAG	Current record flag (Y/N)	char(1)	YES	
EXP_CODE	Exposure type	varchar(2)	NO	AA - Workhours AF - Fixed wing aircraft AR - Rotary wing aircraft BU - Bus GC - Government car MR - Marine craft RR - Rail TK - Truck (1 ton or more)
EXP_CONV	Number of conveyences	numeric(12)	YES	This field will always contain a zero for the exposure type AA (workhours).
EXP_IDENT	Unique record number	numeric(12)	NO	
EXP_KEY	Concatenation of org code, year, and quarter	varchar(14)	NO	
EXPOSURE	Total workhours, miles, or hours	numeric(12)	YES	
ID	(Currently not used - empty field)	varchar(8)	YES	Not in use
ORG	Organization code	varchar(8)	YES	
QTR	Quarter	varchar(2)	YES	
TIMESTAMP	Record timestamp	varchar(14)	NO	
YEAR	Year	varchar(4)	YES	

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: exposureroll (field office or operation type vehicle accident and injury/illness summary table)				
CODE	Exposure type	varchar(2)	NO	AA - Workhours AF - Fixed wing aircraft AR - Rotary wing aircraft BU - Bus GC - Government car MR - Marine craft RR - Rail TK - Truck (1 ton or more)
CONV	Number of conveyances	numeric(12)	NO	
CURRENTFLAG	Current record flag (Y/N)	varchar(1)	NO	
EXPKEY	Key for Field Office and Operation Type rollups	varchar(7)	NO	
EXPOSURE	Total workhours, miles, or hours	numeric(12)	NO	
FO	Field Office code	varchar(2)	NO	This field will be coded as 99 if the record is for an operation type.
OPTYPE	Operation type code	varchar(1)	NO	This field will be coded as 0 if the record is for a Field Office.
QTR	Quarter	varchar(1)	NO	
TIMESTAMP	Record timestamp	varchar(14)	NO	
YEAR	Year	varchar(4)	NO	
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: factors (influencing factors or underlying causes text table)				
CURRENTFLAG	Current record flag (Y/N)	char(1)	YES	
GIC	Report number	varchar(8)	NO	
LINENUM	Line identifier for each 255-character record	numeric(2)	NO	
TEXT	Influencing factors text	varchar(255)	NO	
TIMESTAMP	Record timestamp	varchar(14)	NO	

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: gicsView (injury/illness, vehicle accident, and property loss information)				
activitycode	Activity code	varchar(4)	YES	
adate	Accident date	varchar(8)	YES	Format is yyyyymmdd.
aday	Accident day	varchar(2)	YES	Format is dd.
adddate	The date the report was first entered into CAIRS	varchar(8)	YES	Format is yyyyymmdd.
age	Age of employee/operator	numeric(3)	YES	
akey	Concatenation of organization code and case number	varchar(14)	YES	
amonth	Accident month	varchar(2)	YES	Format is mm.
ao	Area Office	varchar(4)	YES	
aplace	Accident place code	varchar(2)	YES	
atime	Time of the accident in military hours	varchar(2)	YES	Format is hh.
atype	Accident type	varchar(2)	YES	
ayear	Accident year	varchar(4)	YES	Format is yyyy.
bodypartii	Body part affected code	varchar(4)	YES	
caseid	Case number	varchar(8)	YES	Format is yyyynnn.
claim	Damage to non-DOE vehicle/property	numeric(12)	YES	
claimp	Amount paid by DOE for damage to non-DOE vehicle/property	numeric(12)	YES	
currentflag	Current record flag (Y/N)	char(1)	YES	
dcause	Direct cause code	varchar(4)	YES	
ddate	Death date	varchar(8)	YES	Format is yyyyymmdd.
death	Death flag (Y/N)	varchar(2)	YES	
department	Department, division, or ID code	varchar(40)	YES	
descript	System generated one-line accident descriptions	varchar(255)	YES	
dloss	DOE loss	numeric(12)	YES	
dmg_target	Target of property damage code	varchar(14)	YES	This field was a new field beginning 1/26/1998. However, it has been populated back to 1995.
dmg_target_old	Obsolete	varchar(4)	YES	Field not in use.

dob	Date of birth of employee/operator	varchar(8)	YES	This field is a new field beginning 1/26/1998. Currently only year is being collected.
dspecific	Specific location of the accident	varchar(255)	YES	
epremise	Accident occurred on employer's premises	varchar(2)	YES	
eq_c	Equipment design or defect flag	varchar(2)	YES	
eq1_b	Obsolete	varchar(50)	YES	Field not in use.
eq1_g	Injury/illness source code	varchar(4)	YES	
eq1_id	Obsolete	varchar(10)	YES	Field not in use.
eq2_b	Obsolete	varchar(50)	YES	Field not in use.
eq2_g	Secondary injury/illness source code or property/vehicle source code	varchar(14)	YES	
eq2_id	Obsolete	varchar(10)	YES	Field not in use.
facility	Facility code	varchar(10)	YES	Field not in use.
fiscalyear	Fiscal Year (yyyy)	varchar(4)	YES	
fo	Field office code	varchar(2)	YES	
gic	Report number	varchar(8)	NO	
gspecific	Obsolete	varchar(4)	YES	Field not in use.
hire_date	Date of hire of employee/operator	varchar(8)	YES	This field is a new field beginning 1/26/1998. Currently only year is being collected.
hlth_provider	Name of health care provider	varchar(100)	YES	This field is a new field beginning 1/26/1998.
hosp_ovrnite	Hospitalized overnight flag (Y/N)	varchar(2)	YES	This field is a new field beginning 1/26/1998.
hospital	Name and address of treatment facility	varchar(100)	YES	This field is a new field beginning 1/26/1998.
icause	Indirect cause code(s)	varchar(8)	YES	
iclosed	Employee returned to work flag (Y/N)	varchar(2)	YES	
liitype	Injury/illness type code	varchar(4)	YES	
itype	Investigation type	varchar(2)	YES	
lequip	Experience on job or equipment code	varchar(2)	YES	

losseventcode	Loss producing event code	varchar(4)	YES	This field is a new field beginning 1/26/1998. Only records that have been added or modified since that time have a valid loss code. All other records show a loss code of 9999.
lwd	Lost workdays	numeric(6)	YES	
moddate	The date the report was last modified.	varchar(8)	YES	Format is yyyyymmdd.
mult	Multiple case number	varchar(8)	YES	
name	Name of employee/operator	varchar(30)	YES	This field is not available for general use.
occup	Occupation code	varchar(4)	YES	
occupd	Occupation description	varchar(100)	YES	
oldcairs	HP flag (Y/N)	char(1)	YES	Indicates if a record was present on the pre-GUI production platform.
olddescript	Obsolete	varchar(255)	YES	Field not in use.
oldeq1_g	Obsolete	varchar(4)	YES	Field not in use.
oldeq2_g	Obsolete	varchar(4)	YES	Field not in use.
oldoccup	Obsolete	varchar(4)	YES	Field not in use.
oloss	Non-DOE property or vehicle loss	numeric(12)	YES	
op_code	Operator of vehicle or equipment flag (Y/N)	varchar(2)	YES	This field is a new field beginning 1/26/1998. Only records that have been added since that time will have values.
optype	Operation type code	varchar(4)	YES	
org	Organization code	varchar(8)	YES	
osha	OSHA classification code	varchar(2)	YES	
oshatype	OSHA type code	varchar(1)	YES	
pcode	Property loss type code	varchar(4)	YES	
pempl	Length of employment code	varchar(8)	YES	
ppecodes	Personal protective equipment code	varchar(14)	YES	

pv_rec_inj	Property/vehicle accident resulted in injury flag (Y/N)	char(1)	YES	This field is a new field beginning 1/26/1998. Only records that have been added since that time will have values.
pvclosed	Dollar amounts final flag (Y/N)	varchar(2)	YES	
pvcode	Property/vehicle loss type code	varchar(4)	YES	
reallyclosed	Injury/illness case closed flag	varchar(1)	YES	
reimb	Dollar loss that should be reimbursable to DOE	numeric(12)	YES	
reimbr	Reimbursable amount actually paid to DOE	numeric(12)	YES	
seatb	Vehicle equipped with seat belts	char(1)	YES	
seatbw	Seat belt in use	char(1)	YES	
sex	Gender of employee/operator	char(1)	YES	
term	Termination flag (Y/N)	varchar(2)	YES	
timestamp	Record timestamp	varchar(14)	NO	Format is yyyyymmddhhmmss.
tloss	Total property/vehicle damage	numeric(12)	YES	
transfer	Transfer flag (Y/N)	varchar(2)	YES	
userid	User ID of person who entered or modified the record.	varchar(8)	YES	
vcode	Vehicle loss type code	varchar(4)	YES	
wdl	Workdays lost	numeric(6)	YES	
wdlr	Workdays restricted	numeric(6)	YES	
wtime	Time employee began work	varchar(2)	YES	Field not in use.
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: IType (injury/illness type code lookup table)				
Code	Injury/illness type code	char(4)	NO	
Description	Injury/illness type description	varchar(255)	NO	

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: illnessroll (quarterly organization summary table for illness data)				
currentflag	Current record flag (Y/N)	char(1)	YES	
DAFWC	Total days away from work cases	numeric(12)	NO	
DEATH	Total deaths	numeric(12)	NO	
FQTR	Fiscal quarter	varchar(1)	YES	
FYEAR	Fiscal year	varchar(4)	YES	Format is yyyy.
hours	Workhours	numeric(12)	NO	
ILLKEY	Concatenation of organization code, year, and quarter	varchar(12)	NO	
LWC	Total lost workday cases	numeric(12)	NO	
nfc	Total non-fatal cases without lost workdays	numeric(12)	NO	
ORG	Organization code	varchar(7)	NO	
QTR	Quarter	varchar(1)	NO	
terms	Total terminations and transfers	numeric(12)	NO	
timestamp	Record timestamp	varchar(14)	NO	Format is yyyyymmddhhmmss.
TRC	Total recordable cases	numeric(12)	NO	
WDL	Number of workdays lost	numeric(12)	NO	
wdlr	Number of workdays restricted	numeric(12)	NO	
YEAR	Year	varchar(4)	NO	Format is yyyy.

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: injuryillnessroll (quarterly organization summary table for combined injury and illness data)				
currentflag	Current record flag (Y/N)	char(1)	YES	
DAFWC	Total days away from work cases	numeric(12)	NO	
DEATH	Total deaths	numeric(12)	NO	
FQTR	Fiscal quarter	varchar(1)	YES	
FYEAR	Fiscal year	varchar(4)	YES	Format is yyyy.
hours	Workhours	numeric(12)	NO	
IC	Total number of illnesses	numeric(12)	NO	
INJKEY	Concatenation of organization code, year, and quarter	varchar(12)	NO	
LWC	Total lost workday cases	numeric(12)	NO	
nfc	Total non-fatal cases without lost workdays	numeric(12)	NO	
ORG	Organization code	varchar(7)	NO	
QTR	Quarter	varchar(1)	NO	
terms	Total terminations and transfers	numeric(12)	NO	
timestamp	Record timestamp	varchar(14)	NO	Format is yyyyymmddhhmmss.
TRC	Total recordable cases	numeric(12)	NO	
WDL	Number of workdays lost	numeric(12)	NO	
wdlr	Number of workdays restricted	numeric(12)	NO	
YEAR	Year	varchar(4)	NO	Format is yyyy

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: injuryorg (quarterly organization summary table for individual OSHA classifications)				
currentflag	Current record flag (Y/N)	char(1)	YES	
DAFWC	Total days away from work cases	numeric(12)	NO	
DEATH	Total deaths	numeric(12)	NO	
INJKEY	Concatenation of organization code, year, and quarter	varchar(12)	NO	
IOrg_ID	Unique record number	numeric(12)	NO	
LWC	Total lost workday cases	numeric(12)	NO	
NFC	Total non-fatal cases without lost workdays	numeric(12)	NO	
ORG	Organization code	varchar(7)	NO	
OSHA	OSHA classification code	varchar(2)	NO	
QTR	Quarter	varchar(1)	NO	
TERMS	Total terminations and transfers	numeric(12)	NO	
timestamp	Record timestamp	varchar(14)	NO	Format is yyyyymmddhhmmss.
TRC	Total recordable cases	numeric(12)	NO	
WDL	Number of workdays lost	numeric(12)	NO	
WDLR	Number of workdays restricted	numeric(12)	NO	
YEAR	Year	varchar(4)	NO	Format is yyyy.

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: injuryroll (quarterly organization summary table for injury data)				
currentflag	Current record flag (Y/N)	char(1)	YES	
DAFWC	Total days away from work cases	numeric(12)	NO	
DEATH	Total deaths	numeric(12)	NO	
FQTR	Fiscal quarter	varchar(1)	YES	
FYEAR	Fiscal year	varchar(4)	YES	
hours	Workhours	numeric(12)	NO	
INJKEY	Concatenation of organization code, year, and quarter	varchar(12)	NO	
LWC	Total lost workday cases	numeric(12)	NO	
nfc	Total non-fatal cases without lost workdays	numeric(12)	NO	
ORG	Organization code	varchar(7)	NO	
QTR	Quarter	varchar(1)	NO	
terms	Total terminations and transfers	numeric(12)	NO	
timestamp	Record timestamp	varchar(14)	NO	Format is yyyyymmddhhmmss.
TRC	Total recordable cases	numeric(12)	NO	
WDL	Number of workdays lost	numeric(12)	NO	
wdlr	Number of workdays restricted	numeric(12)	NO	
YEAR	Year	varchar(4)	NO	Format is yyyy.
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: LossEvents (Loss producing event code lookup table)				
Code	Loss producing event code	char(4)	NO	
Description	Loss producing event description	varchar(255)	NO	
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: materials (equipment, materials, and chemicals text table -- (this table is currently not populated with text)				
CURRENTFLAG	Current record flag (Y/N)	char(1)	YES	
GIC	Report number	varchar(8)	NO	
LINENUM	Line identifier for each 255-character record	numeric(2)	NO	
TEXT	Equipment, materials, chemicals text	varchar(255)	NO	
TIMESTAMP	Record timestamp	varchar(14)	NO	

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: Objects (primary and secondary source code lookup table)				
Code	Source and target code	char(4)	NO	
Description	Source and target description	varchar(255)	NO	
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: Occupations (occupation code lookup table)				
Code	Occupation code	char(4)	NO	
Description	Occupation description	varchar(255)	NO	
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: office (field office lookup table)				
Active	Active flag (Y/N)	varchar(1)	YES	
ORG	Seven character Field Office code	varchar(8)	NO	
ORG_NAME	Field Office name	varchar(40)	YES	
ORG_NAME_SHORT	Field Office name short	varchar(20)	YES	
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: Opertype (operation type code lookup table)				
Code	Operation type code	char(4)	NO	
Description	Operation type description	varchar(255)		

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: orgd (reporting organization table -- the only up-to-date information in this table are the Active, ORG, ORG_NAME, and ORG_NAME_SHORT fields)				
Active	Active flag (Y/N)	varchar(1)	YES	
ORG	Organization code	varchar(8)	NO	
ORG_CITY	City	varchar(30)	YES	
ORG_COUNTRY	Country	varchar(20)	YES	
ORG_EDRS_FAC	EDRS Facility	varchar(6)	YES	
ORG_EDRS_OP	Operations Office	varchar(6)	YES	
ORG_FTS	FTS prefix- obsolete	varchar(4)	YES	
ORG_NAME	Organization name	varchar(40)	YES	
ORG_NAME_SHORT	Short organization name	varchar(20)	YES	
ORG_PHONE	Phone	varchar(14)	YES	
ORG_PO_BOX	P.O. Box	varchar(16)	YES	
ORG_STATE	State	varchar(2)	YES	
ORG_STREET	Street	varchar(30)	YES	
ORG_UOR_FAC	UOR facilities	varchar(36)	YES	
ORG_UOR_HQ_CODE	Program Office	varchar(2)	YES	
ORG_ZIP	Zip Code	varchar(10)	YES	
TIMESTAMP	Record timestamp	varchar(12)	YES	Format is yyyyymmddhhmmss.
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: propertyorg (quarterly organization summary table for individual property loss codes)				
CODE	Property loss code	varchar(2)	NO	
currentflag	Current record flag (Y/N)	char(1)	YES	
GLOSS	Total DOE loss	numeric(12)	NO	
OCCUR	Number of occurrences	numeric(12)	NO	
OLOSS	Total non-DOE loss	numeric(12)	NO	
ORG	Organization code	varchar(7)	NO	
PROPKEY	Concatenation of organization code, year, and quarter	varchar(12)	NO	
QTR	Quarter	varchar(1)	NO	
timestamp	Record timestamp	varchar(14)	NO	Format is yyyyymmddhhmmss.
TLOSS	Total DOE and non-DOE loss	numeric(12)	NO	
YEAR	Year	varchar(4)	NO	Format is yyyy.

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: propertyroll (quarterly organization summary table for property loss)				
currentflag	Current record flag (Y/N)	char(1)	YES	
FGLOSS	Total fire DOE loss	numeric(12)	YES	
FOCCUR	Number of fire occurrences	numeric(12)	YES	
FOLOSS	Total fire non-DOE loss	numeric(12)	YES	
FQTR	Fiscal quarter	varchar(1)	YES	
FYEAR	Fiscal year	varchar(4)	YES	
NGLOSS	Total non-fire DOE loss	numeric(12)	NO	
NOCCUR	Number of non-fire occurrences	numeric(12)	NO	
NOLOSS	Total non-fire non-DOE loss	numeric(12)	NO	
ORG	Organization code	varchar(7)	NO	
PROPKEY	Concatenation of organization code, year, and quarter	varchar(12)	NO	
QTR	Quarter	varchar(1)	NO	
timestamp	Record timestamp	varchar(14)	NO	Format is yyyyymmddhhmmss.
VALUATION	Property valuation	numeric(12)	NO	
YEAR	Year	varchar(4)	NO	Format is yyyy.
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: propertythreshold (quarterly organization summary table for individual property loss codes with losses above the reporting threshold)				
CODE	Property loss code	varchar(2)	NO	
currentflag	Current record flag (Y/N)	char(1)	YES	
GLOSS	Total DOE loss	numeric(12)	NO	
OCCUR	Number of occurrences	numeric(12)	NO	
OLOSS	Total non-DOE loss	numeric(12)	NO	
ORG	Organization code	varchar(7)	NO	
PROPKEY	Concatenation of organization code, year, and quarter	varchar(12)	NO	
QTR	Quarter	varchar(1)	NO	
timestamp	Record timestamp	varchar(14)	NO	Format is yyyyymmddhhmmss.
TLOSS	Total DOE and non-DOE loss	numeric(12)	NO	
YEAR	Year	varchar(4)	NO	Format is yyyy.

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: propertythresroll (quarterly organization summary table for property loss above the reporting threshold)				
currentflag	Current record flag (Y/N)	char(1)	YES	
FGLOSS	Total fire DOE loss	numeric(12)	YES	
FOCCUR	Total fire occurrences	numeric(12)	YES	
FOLOSS	Total fire non-DOE loss	numeric(12)	YES	
FQTR	Fiscal quarter	varchar(1)	YES	
FYEAR	Fiscal year	varchar(4)	YES	
NGLOSS	Total non-fire DOE loss	numeric(12)	NO	
NOCCUR	Total non-fire occurrences	numeric(12)	NO	
NOLOSS	Total non-fire non-DOE loss	numeric(12)	NO	
ORG	Organization code	varchar(7)	NO	
PROPKEY	Concatenation of organization code, year, and quarter	varchar(12)	NO	
QTR	Quarter	varchar(1)	NO	
timestamp	Record timestamp	varchar(14)	NO	Format is yyyyymmddhhmmss.
VALUATION	Property valuation	numeric(12)	NO	
YEAR	Year	varchar(4)	NO	Format is yyyy.
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: PropLossType (property loss type code lookup table)				
Code	Property loss type code	char(4)	NO	
Description	Property loss type description	varchar(255)	NO	
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: ProtectionEq (protective equipment code lookup table)				
Code	Protective equipment code	char(4)	NO	
Description	Protective equipment description	varchar(255)	NO	
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: valuation (property valuation table)				
CurrentFlag	Current record flag (Y/N)	char(1)	NO	
EXP_KEY	Concatenation of organization code and year	varchar(11)	NO	
ORG	Organization code	varchar(7)	NO	
TIMESTAMP	Record timestamp	varchar(14)	NO	Format is yyyyymmddhhmmss.
VALUATION	Property valuation	numeric(12)	NO	
YEAR	Year	varchar(4)	NO	Format is yyyy.

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: vehiclemilesroll (quarterly organization summary table for vehicle accidents)				
CONV	Number of conveyances	numeric(12)	NO	
CurrentFlag	Current record flag (Y/N)	char(1)	YES	
FQTR	Fiscal quarter	varchar(1)	YES	
FYEAR	Fiscal year	varchar(4)	YES	
GLOSS	Total DOE loss	numeric(12)	NO	
MILES	Miles or hours operated	numeric(12)	NO	
OCCUR	Number of occurrences	numeric(12)	NO	
OLOSS	Total non-DOE loss	numeric(12)	NO	
ORG	Organization code	varchar(7)	NO	
QTR	Quarter	varchar(1)	NO	
Timestamp	Record timestamp	varchar(14)	NO	Format is yyyymmddhhmmss.
TLOSS	Total DOE and non-DOE loss	numeric(12)	NO	
VEHKEY	Concatenation of organization code, year, and quarter	varchar(12)	NO	
YEAR	Year	varchar(4)	NO	Format is yyyy.
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: vehicleorg (quarterly organization summary table for individual vehicle type codes)				
CODE	Vehicle type code	varchar(3)	NO	
currentflag	Current record flag (Y/N)	char(1)	YES	
GLOSS	Total DOE loss	numeric(12)	NO	
OCCUR	Number of occurrences	numeric(12)	NO	
OLOSS	Total non-DOE loss	numeric(12)	NO	
ORG	Organization code	varchar(7)	NO	
QTR	Quarter	varchar(1)	NO	
timestamp	Record timestamp	varchar(14)	NO	Format is yyyymmddhhmmss.
TLOSS	Total DOE and non-DOE loss	numeric(12)	NO	
VEHKEY	Concatenation of organization code, year, and quarter	varchar(12)	NO	
YEAR	Year	varchar(4)	NO	Format is yyyy.

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: vehic lethreshold (quarterly organization summary table for individual vehicle type codes with losses above the reporting threshold)				
CODE	Vehicle type code	varchar(3)	NO	
currentflag	Current record flag (Y/N)	char(1)	YES	
GLOSS	Total DOE loss	numeric(12)	NO	
OCCUR	Number of occurrences	numeric(12)	NO	
OLOSS	Total non-DOE loss	numeric(12)	NO	
ORG	Organization code	varchar(7)	NO	
QTR	Quarter	varchar(1)	NO	
timestamp	Record timestamp	varchar(14)	NO	Format is yyyymmddhhmmss.
TLOSS	Total DOE and non-DOE loss	numeric(12)	NO	
VEHKEY	Concatenation of organization code, year, and quarter	varchar(12)	NO	
YEAR	Year	varchar(4)	NO	Format is yyyy.
FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: vehic lethreshroll (quarterly organization summary table for vehicle accidents with losses above the reporting threshold)				
CONV	Number of conveyences	numeric(12)	NO	
currentflag	Current record flag (Y/N)	char(1)	YES	
FQTR	Fiscal quarter	varchar(1)	YES	
FYEAR	Fiscal year	varchar(4)	YES	
GLOSS	Total DOE loss	numeric(12)	NO	
MILES	Miles or hours operated	numeric(12)	NO	
OCCUR	Number of occurrences	numeric(12)	NO	
OLOSS	Total non-DOE loss	numeric(12)	NO	
ORG	Organization code	varchar(7)	NO	
QTR	Quarter	varchar(1)	NO	
timestamp	Record timestamp	varchar(14)	NO	Format is yyyymmddhhmmss.
TLOSS	Total DOE and non-DOE loss	numeric(12)	NO	
VEHKEY	Concatenation of organization code, year, and quarter	varchar(12)	NO	
YEAR	Year	varchar(4)	NO	Format is yyyy.

FIELD	DESCRIPTION	TYPE NAME	CAN BE NULL	COMMENTS
Table: VehType (vehicle type code lookup table)				
Code	Vehicle type code	char(4)	NO	
Description	Vehicle type description	varchar(255)	NO	

Appendix C - Sample Template Reports

Exercise Solutions

Exercise 1

1. Write the SQL SELECT statement that selects all records with a date after April 15, 1998 and before May 16, 1998. Show all columns on your report.

```
SELECT *  
FROM TEST.ACCIDENT  
WHERE ACCIDENT.DATE BETWEEN '19980416' AND '19980515'
```

2. Write the SQL SELECT statement that selects all records for injury type "0012." Show only the org, WDLR, and WDL columns on your report.

```
SELECT ACCIDENT.ORG, ACCIDENT.WDLR, ACCIDENT.WDL  
FROM TEST. ACCIDENT  
WHERE ACCIDENT.INJURY_TYPE = '0012'
```

3. Write the SQL SELECT statement that selects all records for body part "3101" and org "0502009." Show all columns on your report.

```
SELECT *  
FROM TEST. ACCIDENT  
WHERE ACCIDENT.BODY_PART = '3101' AND  
ACCIDENT.ORG = '0502009'
```

Exercise 2

1. Write the SQL SELECT statement that groups records by injury type. Show columns for injury type, org, and date on your report.

```
SELECT ACCIDENT.INJURY_TYPE, ACCIDENT.ORG, ACCIDENT.DATE  
FROM TEST.ACCIDENT  
GROUP BY ACCIDENT.INJURY_TYPE, ACCIDENT.ORG,  
ACCIDENT.DATE
```

2. Modify the SELECT statement to show the count of records by injury type. Label the column "COUNT."

```
SELECT ACCIDENT.INJURY_TYPE,  
COUNT(ACCIDENT.NUMBER) 'COUNT'  
FROM TEST.ACCIDENT  
GROUP BY ACCIDENT.INJURY_TYPE
```

3. Modify the SELECT statement to show only those counts greater than four records.

```
SELECT ACCIDENT.INJURY_TYPE  
COUNT(ACCIDENT.NUMBER) 'COUNT'  
FROM TEST. ACCIDENT  
GROUP BY ACCIDENT.INJURY_TYPE  
HAVING COUNT(ACCIDENT.INJURY_TYPE) > 4
```

Exercise 3

Write the SQL SELECT statement that sorts records by organization and then sorts the records in descending order by date. Display only the organization and date columns on your report.

```
SELECT ACCIDENT.ORG, ACCIDENT.DATE  
FROM TEST.ACCIDENT  
ORDER BY ACCIDENT.ORG, ACCIDENT.DATE DESC
```

Exercise 4

Write the SQL SELECT statement that joins the **ACCIDENT** table and the **INJURY** table and selects only rows from the joined table when the **INJURY_TYPE** column in the **ACCIDENT** tables equals the **CODE** column in the **INJURY** table. Display only injury type, description, and date columns on your report.

```
SELECT ACCIDENT.INJURY_TYPE, INJURY.DESCRPTION,  
ACCIDENT.DATE  
FROM TEST.ACCIDENT, TEST.INJURY  
WHERE ACCIDENT.INJURY_TYPE = INJURY.CODE
```

Exercise 5

1. Specify criteria to select the latest version of all 1997-1998 reports for Lawrence Berkeley Laboratory (8003003) and Lawrence Livermore National Laboratory (8004003).
 - a. Select **Add Criteria** from the **CRITERIA** menu to open the **ADD CRITERIA** dialog box.
 - b. Select **currentflag** from the **FIELD** selection box.
 - c. Type **Y** in the **VALUE** edit box.
 - d. Click on the **ADD** command button.
 - e. Select **org** from the **FIELD** selection box.
 - f. Select **is one of** from the **OPERATOR** selection box.
 - g. Click on the **VALUES** command button to open the **SELECTED VALUE(S)** dialog box.
 - h. Select **8003003** and **8004003** from the **VALUES** selection box.
 - i. Click on the **OK** command button.
 - j. Click on the **ADD** command button.
 - k. Select **ayear** from the **FIELD** selection box.
 - l. Select **is between** from the **OPERATOR** selection box.
 - m. Click on the **VALUES** command button to open the **SELECTED VALUE(S)** dialog box.
 - n. Select **1997** and **1998** from the **VALUES** selection box.
 - o. Click on the **OK** command button.
 - p. Click on the **ADD** command button.
 - q. Click on the **CLOSE** command button.
2. Display the accident year and occupation code as report columns.
 - a. Select **Add Column** from the **RECORDS** menu to open the **ADD COLUMN** dialog box.
 - b. Select **ayear** from the **FIELD** selection box.
 - c. Type a column heading in the **COLUMN HEADING** edit box.
 - d. Click on the **ADD** command button.
 - e. Select **occup** from the **FIELD** selection box.
 - f. Type a column heading in the **COLUMN HEADING** edit box.
 - g. Click on the **ADD** command button.
 - h. Click on the **CLOSE** command button.
3. Save the query.
 - a. Select **Save** from the **FILE** menu to open the **SAVE AS** dialog box.
 - b. Type the file name into the **FILE NAME** edit box.
 - c. Click on the **SAVE** command button.

Exercise 6

1. Specify criteria to select the latest version of all injury/illness cases for the Richland Field Office.
 - a. Select **Add Criteria** from the **CRITERIA** menu to open the **ADD CRITERIA** dialog box.
 - b. Select **currentflag** from the **FIELD** selection box.
 - c. Type **Y** in the **VALUE** edit box.
 - d. Click on the **ADD** command button.
 - e. Select **fo** from the **FIELD** selection box.
 - f. Click on the **VALUES** command button to open the **SELECTED VALUE(S)** dialog box.
 - g. Type **75** in the **VALUE** edit box.
 - h. Click on the **OK** command button.
 - i. Click on the **ADD** command button.
 - j. Select **atype** from the **FIELD** selection box.
 - k. Click on the **VALUES** command button to open the **SELECTED VALUE(S)** dialog box.
 - l. Select **I** from the **VALUES** selection box.
 - m. Click on the **OK** command button.
 - n. Click on the **ADD** command button.
 - o. Click on the **CLOSE** command button.
2. Display the organization code and accident year as report columns.
 - a. Select **Add Column** from the **RECORDS** menu to open the **ADD COLUMN** dialog box.
 - b. Select **org** from the **FIELD** selection box.
 - c. Type a column heading in the **COLUMN HEADING** edit box.
 - d. Click on the **ADD** command button.
 - e. Select **ayear** from the **FIELD** selection box.
 - f. Type a column heading in the **COLUMN HEADING** edit box.
 - g. Click on the **ADD** command button.
 - h. Click on the **CLOSE** command button.
3. Do a primary ascending sort for organization code and a secondary ascending sort for accident year.
 - a. Select **Sort** from the **RECORDS** menu to open the **SORT** dialog box.
 - b. Select **org** from the **COLUMN** selection box.
 - c. Click and the **ADD** command button.
 - d. Select **ayear** from the **COLUMN** selection box.

Exercise 6 (cont.)

4. Organize your result set into groups by organization and accident year.
 - a. Select **Query Properties** from the **VIEW** menu to open the **QUERY PROPERTIES** dialog box.
 - b. Check the **GROUP RECORDS** check box.
 - c. Click on the **OK** command button.

5. Save the query.
 - a. Select **Save** from the **FILE** menu to open the **SAVE AS** dialog box.
 - b. Type the file name into the **FILE NAME** edit box.
 - c. Click on the **SAVE** command button.

Exercise 7

1. Specify criteria to select the latest version of all injury/illness cases that were reported for 1997.
 - a. Select **Add Criteria** from the **CRITERIA** menu to open the **ADD CRITERIA** dialog box.
 - b. Select **currentflag** from the **FIELD** selection box.
 - c. Type **Y** in the **VALUE** edit box.
 - d. Click on the **ADD** command button.
 - e. Select **atype** from the **FIELD** selection box.
 - f. Click on the **VALUES** command button to open the **SELECTED VALUE(S)** dialog box.
 - g. Type **I** in the **VALUE** edit box.
 - h. Click on the **OK** command button.
 - i. Click on the **ADD** command button.
 - j. Select **ayear** from the **FIELD** selection box.
 - k. Click on the **VALUES** command button to open the **SELECTED VALUE(S)** dialog box.
 - l. Select **1997** from the **VALUES** selection box.
 - m. Click on the **OK** command button.
 - n. Click on the **ADD** command button.
 - o. Click on the **CLOSE** command button.

2. Display by field office the total number of cases, the total number of lost workdays (LWD), and the average number of lost workdays.
 - a. Select **Add Column** from the **RECORDS** menu to open the **ADD COLUMN** dialog box.
 - b. Select **fo** from the **FIELD** selection box.
 - c. Type a column heading in the **COLUMN HEADING** edit box.
 - d. Click on the **ADD** command button.
 - e. Select **gic** from the **FIELD** selection box.
 - f. Type a column heading in the **COLUMN HEADING** edit box.
 - g. Select **Count** from the **TOTAL** selection box.
 - h. Click on the **ADD** command button.
 - i. Click on the **CLOSE** command button.
 - j. Select **lwd** from the **FIELD** selection box.
 - k. Type a column heading in the **COLUMN HEADING** edit box.
 - l. Select **Sum** from the **TOTAL** selection box.
 - m. Click on the **ADD** command button.
 - n. Click on the **CLOSE** command button.

Exercise 7 (cont.)

2. (cont.)
 - o. Select **lwd** from the **FIELD** selection box.
 - p. Type a column heading in the **COLUMN HEADING** edit box.
 - q. Select **Avg** from the **TOTAL** selection box.
 - r. Click on the **ADD** command button.
 - s. Click on the **CLOSE** command button.

3. Save the query.
 - a. Select **Save** from the **FILE** menu to open the **SAVE AS** dialog box.
 - b. Type the file name into the **FILE NAME** edit box.
 - c. Click on the **SAVE** command button.

Exercise 8

1. Specify criteria to select the latest version of all 1998 injury/illness.
 - a. Select **Add Criteria** from the **CRITERIA** menu to open the **ADD CRITERIA** dialog box.
 - b. Select **currentflag** from the **FIELD** selection box.
 - c. Type **Y** in the **VALUE** edit box.
 - d. Click on the **ADD** command button.
 - e. Select **ayear** from the **FIELD** selection box.
 - f. Click on the **VALUES** command button to open the **SELECTED VALUE(S)** dialog box.
 - g. Select **1998** from the **VALUES** selection box.
 - h. Click on the **OK** command button.
 - i. Click on the **ADD** command button.
 - j. Click on the **CLOSE** command button.

2. Display the loss producing event description, loss producing event code, and number of cases, workdays away (WDL), and workdays restricted (WDLR) for each lost producing event. Finally, display a column that calculates the average lost workday (LWD).
 - a. Select **Add Tables** from the **TABLE** menu to open the **ADD TABLES** dialog box.
 - b. Select **LossEvents** from the **TABLE** selection box.
 - c. Click on the **ADD** command button.
 - d. Click on the **CLOSE** command button.
 - e. Drag the **LOSSEVENTCODE** field on the **GICSVIEW** table to the **CODE** field on the **LOSSEVENTS** table to create a join line. (When the warning message is displayed, click on the **YES** command button.)
 - f. Select **Add Column** from the **RECORDS** menu to open the **ADD COLUMN** dialog box.
 - g. Select **LossEvents.Description** from the **FIELD** selection box.
 - h. Type a column heading in the **COLUMN HEADING** edit box.
 - i. Click on the **ADD** command button.
 - j. Select **gicsView.losseventcode** from the **FIELD** selection box.
 - k. Type a column heading in the **COLUMN HEADING** edit box.
 - l. Click on the **ADD** command button.
 - m. Select **gicsView.gic** from the **FIELD** selection box.
 - n. Type a column heading in the **COLUMN HEADING** edit box.
 - o. Select **Count** from the **TOTAL** selection box.
 - p. Click on the **ADD** command button.

Exercise 8 (cont.)

2. (Cont.).
 - q. Select **gicsView.wdl** from the **FIELD** selection box.
 - r. Type a column heading in the **COLUMN HEADING** edit box.
 - s. Select **Sum** from the **TOTAL** selection box.
 - t. Click on the **ADD** command button.
 - u. Select **gicsView.wdlr** from the **FIELD** selection box.
 - v. Type a column heading in the **COLUMN HEADING** edit box.
 - w. Select **Sum** from the **TOTAL** selection box.
 - x. Click on the **ADD** command button.
 - y. Select **gicsView.lwd** from the **FIELD** selection box.
 - z. Type a column heading in the **COLUMN HEADING** edit box.
 - aa. Select **Avg** from the **TOTAL** selection box.
 - bb. Click on the **ADD** command button.
 - cc. Click on the **CLOSE** command button.
3. Order the result set, in ascending order, by the loss producing event code.
 - a. Select **Sort** from the **RECORDS** menu to open the **SORT** dialog box.
 - b. Select **gicsView.losseventcode** in the **COLUMN** selection box.
 - c. Click on the **ADD** command button.
 - d. Click on the **CLOSE** command button.
4. Save the query.
 - a. Select **Save** from the **FILE** menu to open the **SAVE AS** dialog box.
 - b. Type the file name into the **FILE NAME** edit box.
 - c. Click on the **SAVE** command button.

Exercise 9

1. Specify criteria to select the latest version of all 1998 injury/illness.
 - a. Select **Add Criteria** from the **CRITERIA** menu to open the **ADD CRITERIA** dialog box.
 - b. Select **currentflag** from the **FIELD** selection box.
 - c. Type **Y** in the **VALUE** edit box.
 - d. Click on the **ADD** command button.
 - e. Select **ayear** from the **FIELD** selection box.
 - f. Click on the **VALUES** command button to open the **SELECTED VALUE(S)** dialog box.
 - g. Select **1998** from the **VALUES** selection box.
 - h. Click on the **OK** command button.
 - i. Click on the **ADD** command button.
 - j. Click on the **CLOSE** command button.

2. Display the injury type description, body part injured description, and number of cases and average lost workdays (LWD) per injury type/body part injured grouping.
 - a. Select **Add Tables** from the **TABLE** menu to open the **ADD TABLES** dialog box.
 - b. Select **Bodyparts** from the **TABLE** selection box.
 - c. Click on the **ADD** command button.
 - d. Select **IIType** from the **TABLE** selection box.
 - e. Click on the **ADD** command button.
 - f. Click on the **CLOSE** command button.
 - g. Click on the join line and press the *[Delete]* key.
 - h. Drag the **BODYPARTII** field on the **GICSVIEW** table to the **CODE** field on the **BODYPARTS** table to create a join line. (When the warning message is displayed, click on the **YES** command button.)
 - i. Drag the **IITYPE** field on the **GICSVIEW** table to the **CODE** field on the **IITYPE** table to create a join line. (When the warning message is displayed, click on the **YES** command button.)
 - j. Select **Add Column** from the **RECORDS** menu to open the **ADD COLUMN** dialog box.
 - k. Select **IIType.Description** from the **FIELD** selection box.
 - l. Type a column heading in the **COLUMN HEADING** edit box.
 - m. Click on the **ADD** command button.
 - n. Select **Bodyparts.Description** from the **FIELD** selection box.
 - o. Type a column heading in the **COLUMN HEADING** edit box.
 - p. Click on the **ADD** command button.

Exercise 9 (cont.)

2. (Cont.)
 - q. Select **gicsView.gic** from the **FIELD** selection box.
 - r. Type a column heading in the **COLUMN HEADING** edit box.
 - s. Select **Count** from the **TOTAL** selection box.
 - t. Click on the **ADD** command button.
 - u. Select **gicsView.lwd** from the **FIELD** selection box.
 - v. Type a column heading in the **COLUMN HEADING** edit box.
 - w. Select **Avg** from the **TOTAL** selection box.
 - x. Click on the **ADD** command button.
 - y. Click on the **CLOSE** command button.

3. Place a frequency limit on the groups of 40 or more cases and 5 or more average lost workdays. (You may want to change the limit values a few times and view the results.)
 - a. Select **Add Criteria** from the **CRITERIA** menu to open the **ADD CRITERIA** dialog box.
 - b. Select **Count** from the **TOTAL** selection box.
 - c. Select **gicsView.gic** from the **FIELD** selection box.
 - d. Select **is greater than or equal to** from the **OPERATOR** selection box.
 - e. Type **40** in the **VALUE** edit box.
 - f. Click on the **ADD** command button.
 - g. Select **Avg** from the **TOTAL** selection box.
 - h. Select **gicsView.lwd** from the **FIELD** selection box.
 - i. Select **is greater than or equal to** from the **OPERATOR** selection box.
 - j. Type **5** in the **VALUE** edit box.
 - k. Click on the **ADD** command button.
 - l. Click on the **CLOSE** command button.

4. Save the query.
 - a. Select **Save** from the **FILE** menu to open the **SAVE AS** dialog box.
 - b. Type the file name into the **FILE NAME** edit box.
 - c. Click on the **SAVE** command button.

Exercise 10

1. Specify criteria to select the latest version of all cases for a user-defined field office, accident type, and accident year.
 - a. Click on the **AUTO QUERY** tool bar button so it is no longer pressed in.
 - b. If the **CRITERIA PANE** is not displayed in the query window, click on the **SHOW/HIDE CRITERIA** tool bar button.
 - c. Click on the first cell in the **CRITERIA FIELD** row.
 - d. Click on the down arrow in the cell and select **fo** from the list.
 - e. Click on the first cell in the **VALUE** row and type [**Enter a Field Office**].
 - f. Click on the second cell in the **CRITERIA FIELD** row.
 - g. Click on the down arrow in the cell and select **atype** from the list.
 - h. Click on the second cell in the **VALUE** row and type [**Enter an Accident Type**].
 - i. Click on the third cell in the **CRITERIA FIELD** row.
 - j. Click on the down arrow in the cell and select **ayear** from the list.
 - k. Click on the third cell in the **VALUE** row and type [**Enter an Accident Year**].
 - l. Click on the fourth cell in the **CRITERIA FIELD** row.
 - m. Click on the down arrow in the cell and select **currentflag** from the list.
 - n. Click on the fifth cell in the **VALUE** row and Type **'Y'**.

2. Display the key, accident date, reporting organization, and one-line description.
 - a. Select **Add Column** from the **RECORDS** menu to open the **ADD COLUMN** dialog box.
 - b. Select **akey** from the **FIELD** selection box.
 - c. Type a heading in the **COLUMN HEADING** edit box.
 - d. Click on the **ADD** command button.
 - e. Select **adate** from the **FIELD** selection box.
 - f. Type a heading in the **COLUMN HEADING** edit box.
 - g. Click on the **ADD** command button.
 - h. Select **org** from the **FIELD** selection box.
 - i. Type a heading in the **COLUMN HEADING** edit box.
 - j. Click on the **ADD** command button.
 - k. Select **descript** from the **FIELD** selection box.
 - l. Type a heading in the **COLUMN HEADING** edit box.
 - m. Click on the **ADD** command button.
 - n. Click on the **CLOSE** command button.

Exercise 10 (cont.)

3. Run the query.
 - a. Click on the **QUERY NOW** tool bar button.
 - b. Type a field office code in the **ENTER THE FIELD OFFICE** edit box.
 - c. Click on the **OK** command button.
 - d. Type an occurrence category in the **ENTER AN ACCIDENT TYPE** edit box.
 - e. Click on the **OK** command button.
 - f. Type a status in the **ENTER AN ACCIDENT YEAR** edit box.
 - g. Click on the **OK** command button.

4. Save the query.
 - a. Select **Save** from the **FILE** menu.
 - b. Type a file name in the **FILE NAME** edit box.
 - c. Click on the **SAVE** command button.

Exercise 11

1. Open the file saved in **Exercise 5**.
 - a. Select **Open** from the **FILE** menu to open the **OPEN QUERY** dialog box.
 - b. Locate and click on the saved query. Click on the **OPEN** command button.
 - c. Enter your user ID and password in the **SQL SERVER LOGIN** dialog box.
 - d. Click on the **OK** command button.

2. Modify the criteria to accept a user-defined value for organization.
 - a. In the **CRITERIA PANE**, click on the cell in the **VALUE** row under the **ORG** criteria field.
 - b. Type [**Enter an Organization Code**]

3. Change the **OCCUPATION** column to display the description of occupation.
 - a. Select **Add Tables** from the **TABLE** menu to open the **ADD TABLES** dialog box.
 - b. Select **Occupations** from the **TABLE** selection box and click on the **ADD** command button.
 - c. Drag the **OCCUP** field on the **GICSVIEW** table to the **CODE** field on the **OCCUPATIONS** table to create a join line. (When the warning message is displayed, click on the **YES** command button.)
 - d. Click on the **OCCUPATION** column heading on the **DATA PANE**.
 - e. Select **Edit Column** from the **RECORDS** menu to open the **EDIT COLUMN** dialog box.
 - f. Select **Occupations.Description** from the **FIELD** selection box and click on the **OK** command button.

4. Add a report column that displays the number of reports per occupation.
 - a. Click in the empty column next to the occupation column.
 - b. Select **Add Column** from the **RECORDS** menu to display the **ADD COLUMN** dialog box.
 - c. Select **gicsView.gic** from the **FIELD** selection box.
 - d. Type **Number** in the **COLUMN HEADING** edit box.
 - e. Select **Count** from the **TOTAL** selection box.
 - f. Click on the **ADD** command button. Click on the **CLOSE** command button.

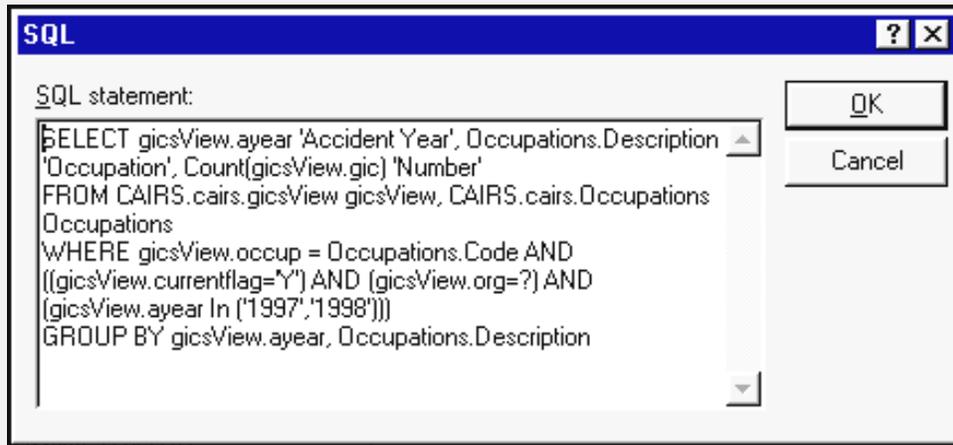
NOTE



If you get a syntax error when you run the query, it is because Microsoft Query put your criteria in a **HAVING** clause. User-defined parameters are not permitted in a **HAVING** clause.
(Continued on next page.)

Exercise 11 (cont.)

You will need to modify the GROUP BY clause to list only the column names in the SELECT clause and adding the HAVING clause predicates to the WHERE clause (shown below):

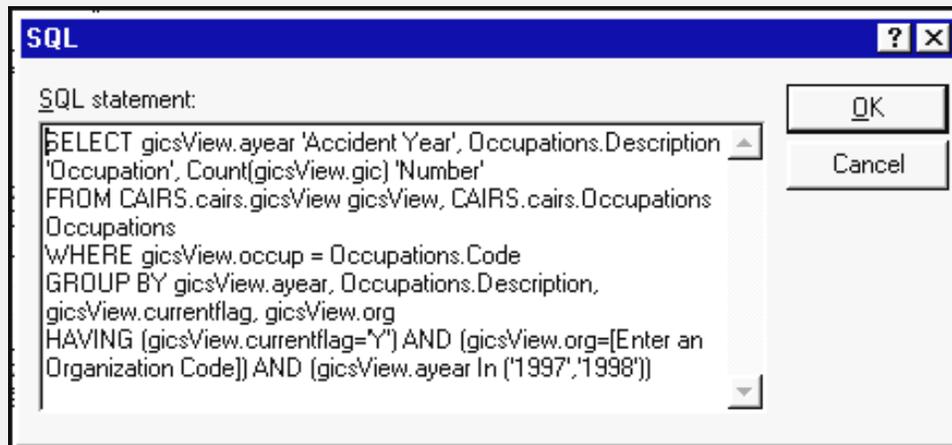


The screenshot shows an SQL dialog box with a blue title bar containing a question mark and a close button. The text area contains the following SQL statement:

```
SQL statement:
SELECT gicsView.ayear 'Accident Year', Occupations.Description
'Occupation', Count(gicsView.gic) 'Number'
FROM CAIRS.cairs.gicsView gicsView, CAIRS.cairs.Occupations
Occupations
WHERE gicsView.occup = Occupations.Code AND
((gicsView.currentflag='Y') AND (gicsView.org=?)) AND
(gicsView.ayear In ('1997','1998'))
GROUP BY gicsView.ayear, Occupations.Description
```

Buttons for 'OK' and 'Cancel' are visible on the right side of the dialog.

An example of a SQL statement with the user-defined parameter correctly placed in the WHERE clause.



The screenshot shows an SQL dialog box with a blue title bar containing a question mark and a close button. The text area contains the following SQL statement:

```
SQL statement:
SELECT gicsView.ayear 'Accident Year', Occupations.Description
'Occupation', Count(gicsView.gic) 'Number'
FROM CAIRS.cairs.gicsView gicsView, CAIRS.cairs.Occupations
Occupations
WHERE gicsView.occup = Occupations.Code
GROUP BY gicsView.ayear, Occupations.Description,
gicsView.currentflag, gicsView.org
HAVING (gicsView.currentflag='Y') AND (gicsView.org=[Enter an
Organization Code]) AND (gicsView.ayear In ('1997','1998'))
```

Buttons for 'OK' and 'Cancel' are visible on the right side of the dialog.

An example of a SQL statement with user-defined parameter incorrectly placed in the HAVING clause.