# **Microsoft Access 2019**

# **Module 9: Using Action Queries and Advanced Table Relationships**

# **A Guide to this Instructor’s Manual:**

We have designed this Instructor’s Manual to supplement and enhance your teaching experience through classroom activities and a cohesive module summary.

This document is organized chronologically, using the same headings in **blue** that you see in the textbook. Under each heading you will find (in order): Lecture Notes that summarize the section, Teacher Tips, Classroom Activities, and Lab Activities. Pay special attention to teaching tips and activities geared toward quizzing your students, enhancing their critical thinking skills, and encouraging experimentation within the software.

In addition to this Instructor’s Manual, our Instructor’s Resources also contains PowerPoint Presentations, Test Banks, and other supplements to aid in your teaching experience.

**Table of Contents**

|  |  |
| --- | --- |
| [Module Objectives](#Chapter_Objectives)  | 1 |
| [Action Queries](#_AC_442_Action)Understanding Types of Table Relationships[Defining M:N and 1:1 Relationships Between Tables](#_AC_461_Defining)[Understanding Join Types](#_AC_464_Joining) | 1233 |
| [Using Indexes for Table Fields](#_AC_472_Using) | 4 |
| [End of Module Material](#EOT) | 5 |

**Module Objectives**

Students will have mastered the material in this module when they can:

Section 9.1

* Create an action query to create a table
* Create an action query to append data
* Construct an action query to delete data
* Build an action query to update data

Section 9.2

* Define many-to-many relationships between tables
* Define one-to-one relationships between tables
* Understand join types
* Create a query using a self-join
* View and create indexes for tables

**Action Queries**

LECTURE NOTES

* Show how to create an action query to create a table.
* Show how to create a Make-Table Query.
* Show how to create an Append Query.
* Show how to create a Delete Query.
* Show how to create an Update Query.

TEACHER TIP

Students have been creating queries that simply create a temporary view of the data based on the criteria of the query. However, students can create other types of queries that actually make a change to the data in the underlying table(s). Action queries can create a new table (make-table query), add records to a table (append query), delete records from a table (delete query), and update the data in a table (update query). When creating a query that actually makes changes to their database, take extreme care that students do not create a query that performs unintended actions. For this reason, it is a good idea for them to create their action query first as a select query. Students can then view the results and determine if these are the records students indeed want to alter. When students are sure the query will be applied to the records they really want, they can then run the query as an action query.

CLASSROOM ACTIVITIES

1. Class Discussion: To stress the importance of running action queries first as select queries, have the students discuss the possible problems that could occur if they do not follow this guideline.

2. Quick Quiz:

* A(n) \_\_\_\_\_ is a query that adds, changes, or deletes multiple table records at a time. (Answer: a)
1. action query
2. history
3. delete
4. update
* A \_\_\_\_\_\_\_\_\_\_ table contains data that is no longer needed for current processing but that you might need to reference in the future. (Answer: history)
* True or False: After you’ve run an action query, you should save it in your database. (Answer: False)
* True/False: For text expressions that contain quotation marks, you need to type the quotation marks twice. (Answer: True)

TEACHER TIP

You usually create an action query and run it for a special purpose, and in most cases you need to run the query only once. If you create and run an action query and then save it, you might accidentally run it again. Doing so would result in the update of tables in unintended ways. Therefore, after you’ve run an action query, you shouldn’t save it in your database; this will prevent users from running it by mistake.

LAB ACTIVITY

Divide the class into groups of four or five. Have students consider the database used in the module. In the first half of this module, students created update queries for the database. Have each group think of at least three other examples of update queries that might be used in this database. Have them determine what tables and fields would be involved in the query. Each group should also be able to justify (give a reason for) the queries they select. Allow several minutes for this exercise. The groups should share their results with the rest of the class.

**Understanding Types of Table Relationships**

LECTURE NOTES

* Define many-to-many and one-to-one relationships between tables.

TEACHER TIP

Students learned previously that one of the basic concepts of relational databases is specifying relationships between tables. Students learned about one-to-many (1:M) relationships, but there are also other types of relationships. A many-to-many relationship (M:N) exists between tables when the tables involved have multiple matches in each of the tables.

It is important that students have a general knowledge of the rules of referential integrity. The rules state that each non-null foreign key value must have a match to the primary key value in the primary table.

A one-to-one relationship (1:1) can also exist between tables, which means that there is exactly one record in the primary table that matches exactly one record in the related table. Most often you will see this kind of relationship when portions of the data need to be separated for some reason.

CLASSROOM ACTIVITIES

1. Classroom Discussion:

* What is a many-to-many relationship (abbreviated as M:N)? (Answer: a relationship that exists between two tables when each record in the first table matches many records in the second table, and each record in the second table matches many records in the first table)
* What is a one-to-one relationship (abbreviated as 1:1)? (Answer: a relationship that exists between two tables when each record in the first table matches at most one record in the second table, and each record in the second table matches at most one record in the first table)

2. Quick Quiz:

* True or False: When you join tables that have a many-to-many relationship, you can extract data from them as if they were one larger table. (Answer: True)
* The primary use for one-to-one relationships is as \_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_. (Answer: entity subtypes)

**Defining M:N and 1:1 Relationships Between Tables**

LECTURE NOTES

* Show how to define M:N and 1:1 relationships between tables.

TEACHER TIP

Explain to students the concepts of the different types of relationships before they begin this section. This knowledge will help them make decisions when they actually create their relationships. To create the relationships, students open the Relationships window, add the desired table and draw (by dragging) lines between the tables. The direction in which they drag the line is very important. In the case of a 1:M relationship, students should always drag the line from the primary table to the related table.

CLASSROOM ACTIVITIES

1. Quick Quiz:

* A(n) \_\_\_\_\_ relationship exists between two tables when each record in the primary table matches zero, one, or many records in the related table, and when each record in the related table matches at most one record in the primary table. (Answer: 1:M, one-to-many)
* True or False: A 1:1 relationship exists when each record in the first table matches at most one record in the second table. (Answer: True)
* One-to-many and many-to-many relationships are defined in the \_\_\_\_\_\_\_\_\_\_\_\_\_ window. (Answer: relationship)

2. Creative Thinking Activity: Let’s say that you have information pertaining to students such as their Student ID, address, phone number, etc. You are also going to maintain information about their courses, grades, and grade point averages. Another collection of data pertaining to students exists, consisting of information related to their accounts at the school. Keeping all this in mind, do the students think there is good reason to separate the data into separate tables? If so, how should the data be separated? Why should it be separated? What type of relationship would need to be established between the tables?

**Understanding Join Types**

LECTURE NOTES

* Show how to join a table using an inner join, outer join, and a self-join.

TEACHER TIP

Students have already learned how to join two tables together by specifying a relationship between them. There are three types of joins that can be used: inner join, outer join, and self-join.

CLASSROOM ACTIVITIES

1. Class Discussion: What are the differences among an inner join, an outer join, and a self-join? Ask students to provide examples of when they would use each one. (Answer: an inner join is a join in which the DBMS selects records from two tables only when the records have the same value in the common field that links the tables. An outer join is a join in which the DBMS selects all records from one table and only those records from a second table that have matching common field values. A self-join can be either an inner or outer join.)

2. Quick Quiz:

* True or False: Any relationship you create in Access is by default an inner join. (Answer: true)

3. Creative Thinking Activity: Based on the previous Class Discussion activity, ask students to come up with at least two examples of additional joins that might be warranted in the lesson database.

**Using Indexes for Table Fields**

LECTURE NOTES

* Demonstrate creating an Index.

TEACHER TIP

When students establish a primary key for a table, they are actually creating an index for the table. An index is a list that relates field values to the records that contain those field values. The index makes it faster to search for records and to sort the records in a table. There is a tradeoff, however, in that each index actually is a new file and takes up space. If saving space is an issue, the number of indexes created should be kept as low as possible.

CLASSROOM ACTIVITIES

1. Quick Quiz:

* True or False: Access automatically creates an index for a table’s primary key. (Answer: true)
* True or False: With small tables, the increased speed associated with indexes is readily apparent. (Answer: False)
* True or False: You can view the existing indexes for a table. (Answer: true)
* True or False: You cannot create an index for a single field in the Indexes dialog box. (Answer: false)

2. Class Discussion: How do you view a table’s existing indexes? (Answer: open the table in Design view. To view an index for a single field, click the field, and then view the Indexed property in the Field Properties pane. To view all the indexes for a table or to view an index consisting of multiple fields, click the Indexes button in the Show/Hide group on the Design tab.)

LAB ACTIVITY

Divide the class into groups of four or five. Have each group consider the following:

You have a database that contains information about a children’s soccer league. The database contains the following tables: Players (information about each child), Coaches (information about the coaches), Teams (information such as team name, age category, and coach name), and Fields (information about the various soccer fields in the community).

Have each group draw the fields for each table, establish a primary key for each table, and determine what foreign keys exist. Next, have the groups draw relationship arrows and determine what kind of relationship would exist between them. When the groups are finished with their design, have them share with the rest of the class. Each group should be able to defend the decisions that it made.

**End of Module Material**

* **Review Assignments:** Review Assignments provide students with additional practice of the skills they learned in the module using the same module case, with which they are already familiar. These assignments are designed as straight practice and do not include anything of an exploratory nature.
* **Case Problems:** A typical NP module has four Case Problems following the Review Assignments. Short modules can have fewer Case Problems (or none at all); other modules may have five Case Problems. The Case Problems provide further hands-on assessment of the skills and topics presented in the module, but with new case scenarios. There are five types of Case Problems:
* **Apply**. In this type of Case Problem, students apply the skills that they have learned in the module to solve a new problem.
* **Create**. In a Create Case Problem, students are either shown the end result (such as a finished Word document) and asked to create the document based on the figure provided, or, students are asked to create something from scratch in a more free-form manner.
* **Challenge**. A Challenge Case Problem involves one or more Explore steps. These steps challenge students by having them go beyond what was covered in the module, either with guidance in the step or by using online Help as directed.
* **Research**. A Research Case Problem requires students to find information on the Internet to help solve a problem or to include in the file they are creating.
* **Troubleshoot**. In this type of Case Problem, certain steps of the exercise require students to identify and correct errors that are intentionally placed in the files. Completing these steps helps to promote problem solving and critical thinking.

[Top of Document](#_Excel_2007)