13. Define a field validation rule for the Gender field in the tblStudent table. Acceptable field values for the Gender field are F or M. Use the message "Gender values must be F or M" to notify a user who enters an invalid Gender field value. Save your table changes, test the field validation rule for the Gender field, making sure any tested field values are the same as they were before your testing, and then close the table.
14. Define a table validation rule for the **tblContract** table to verify that ContractStartDate field values precede ContractEndDate field values in time. Use an appropriate validation message. Save your table changes, test the table validation rule, making sure any tested field values are the same as they were before your testing, and then close the table.
15. Designate the Access2\Case1 folder as a trusted folder. (*Note:* Check with your instructor before adding a new trusted location.)
16. Close the Contract database without exiting Access, make a backup copy of the database, and compact and repair and then close the Contract database.

Apply your skills to work in a database for a health and fitness center.

APPLY

Case Problem 2

Data File needed for this Case Problem: Training.accdb

Parkhurst Health & Fitness Center Martha Parkhurst owns and operates the Parkhurst Health & Fitness Center in Richmond, Virginia. The center offers the usual weight training equipment and fitness classes and also offers specialized programs designed to meet the needs of athletes who participate in certain sports or physical activities. Martha created the Training database to maintain information about the members who have joined the center and the types of programs offered. To make the database easier to use, Martha wants you to create several queries and to make changes to its table design. Complete the following steps:

1. Open the **Training** database, which is located in the Access2\Case2 folder provided with your Data Files.
2. Modify the first record in the **tblMember** table datasheet by changing the First Name and Last Name column values to your first and last names. Close the table.
3. Create a query to find all records in the tblProgram table in which the MonthlyFee field value is 20, 30, or 40. Use a list-of-values match for the selection criterion, and include all fields from the table in the query recordset. Sort the query in descending order by the ProgramID field. Save the query as **qrySelectedPrograms**, run the query, and then close it.
4. Make a copy of the qrySelectedPrograms query using the new name **qrySelectedProgramsModified**. Modify the new query to find all records in the tblProgram table in which the MonthlyFee field value is not 20, 30, or 40. Save and run the query, and then close it.
5. Create a query to display all records from the tblMember table, selecting the LastName, FirstName, Street, and Phone fields, and sorting in ascending order by LastName and then in ascending order by FirstName. Add a calculated field named **MemberName** as the first column that concatenates FirstName, a space, and LastName. Set the Caption property for the MemberName field to **Member Name**. Do not display the FirstName and LastName fields in the query recordset. Create a second calculated field named **CityLine**, inserting it between the Street and Phone fields. The CityLine field concatenates City, a space, State, two spaces, and Zip. Set the Caption property for the CityLine field to **City Line**. Save the query as **qryMemberNames**, run the query, resize all columns to their best fit, and then save and close the query.