EXPLORE 6. Create a query to display all matching records from the tblProgram and tblMember tables, selecting the ProgramType and MonthlyFee fields from the tblProgram table, and the FirstName and LastName fields from the tblMember table. Add a calculated field named **MonthlyFeeStatus** as the last column that equals Active if the MembershipStatus field is equal to Active and equals Not Active otherwise. Set the Caption property for the calculated field to **Monthly Fee Status**. Save the query as **qryMonthlyFeeStatus**, run the query, resize all columns to their best fit, and then save and close the query.

7. Make a copy of the qryRichmondOnHold query using the new name **qryRichmondAndChesterActive**. Modify the new query to select all records in which the City field value is Richmond or Chester and the MembershipStatus field value is Active. Save and run the query, and then close the query.

8. Create a parameter query to select the tblMember table records for a City field value that the user specifies. If the user doesn't enter a City field value, select all records from the table. Display all fields from the tblMember table in the query recordset. Save the query as **qryMemberCityParameter**. Run the query and enter no value as the City field value, and then run the query again and enter **Ashland** as the City field value. Close the query.

9. Create a crosstab query based on the qryMonthlyFeeStatus query. Use the ProgramType field values for the row headings, the MonthlyFeeStatus field values for the column headings, the sum of the MonthlyFee field values as the summarized value, and include row sums. Save the query as **qryMonthlyFeeCrosstab**, resize the columns in the query recordset to their best fit, and then save and close the query.

10. Create a find duplicates query based on the tblMember table. Select ExpirationDate as the field that might contain duplicates, and select all other fields in the table as additional fields in the query recordset. Save the query as **qryDuplicateMemberExpirationDates**, run the query, and then close it.

11. Create a find unmatched query that finds all records in the tblProgram table for which there is no matching record in the tblMember table. Select all fields from the tblProgram table. Save the query as **qryProgramsWithoutMembers**, run the query, and then close it.

12. Create a new query based on the tblMember table. Display the FirstName, LastName, Phone, ExpirationDate, MembershipStatus, and ProgramID fields, in this order, in the query recordset. Sort in ascending order by the ExpirationDate field, and then use the Top Values property to select the top 25% of records. Save the query as **qryUpcomingExpirations**, run the query, and then close it.

EXPLORE 13. Use the Input Mask Wizard to add an input mask to the JoinDate field in the **tblMember** table. Select the Short Date input mask, and then modify the default Short Date input mask by changing the two slashes to dashes. Next for the JoinDate field, set the Format property to **mm-dd-yyyy** to specify the date format with dashes instead of slashes. Test the input mask by typing over an existing Join Date column value, being certain not to change the value by pressing the Esc key after you type the last digit in the Join Date column. Finally, repeat the same procedure to add the same input mask and Format property setting to the ExpirationDate field, and then save and close the table.

14. Define a field validation rule for the MonthlyFee field in the **tblProgram** table. Acceptable field values for the MonthlyFee field are values between 15 and 55. Enter the message **Value must be between 15 and 55, inclusive** so it appears if a user enters an invalid MonthlyFee field value. Save your table changes and then test the field validation rule for the MonthlyFee field; be certain the field values are the same as they were before your testing, and then close the table.