4. Generate the same result set described in exercise 2, but use the implicit join syntax.
5. Write a SELECT statement that returns five columns from three tables, all using column aliases:

Vendor	VendorName column
Date	InvoiceDate column
Number	InvoiceNumber column
#	InvoiceSequence column
LineItem	InvoiceLineItemAmount column

Assign the following correlation names to the tables:

v	Vendors table
i	Invoices table
li	InvoiceLineItems table

Sort the final result set by Vendor, Date, Number, and #.

6. Write a SELECT statement that returns three columns:

VendorID	From the Vendors table
VendorName	From the Vendors table
Name	A concatenation of VendorContactFName and VendorContactLName, with a space in between

The result set should have one row for each vendor whose contact has the same first name as another vendor's contact. Sort the final result set by Name.

Hint: Use a self-join.

7. Write a SELECT statement that returns two columns from the GLAccounts table: AccountNo and AccountDescription. The result set should have one row for each account number that has never been used. Sort the final result set by AccountNo.

Hint: Use an outer join to the InvoiceLineItems table.

8. Use the UNION operator to generate a result set consisting of two columns from the Vendors table: VendorName and VendorState. If the vendor is in California, the VendorState value should be "CA"; otherwise, the VendorState value should be "Outside CA." Sort the final result set by VendorName.