Qlik Q

Learning module: **Loading Data from a Database**

Exercise - INSTRUCTIONS

Using the example download files on the right, along with the creation of a new application in Qlik Sense Desktop, experience will be gained in creating and configuring data connections for databases.

Step 1: Create a new app

- Launch Qlik Sense Desktop.
- Create a new app.
    - o   Name the new app: "Database Connections".

Step 2: Prepare the data load script

- Use the **Script editor** (🗏) option to launch the data load editor view.
- **Create a new section** titled "SOP" in preparation for your first database extraction.

Step 3: Create the connection

- Create a new **OLE DB** database connection.
- Use the provider: **Microsoft Jet 4.0 OLE DB Provider(32-bit)**.
- Enter the full drive letter and path where you stored the example database, along with the filename, e.g., "C:\Data\SOP.mdb".
- Click the **Test connection** button.
- Name the connection "SOP".

Step 4: Move the database

- In Windows explorer, create a subfolder underneath your Data folder, called "Company".
- Move the SOP.mdb (and the lock file, 📇 **SOP.ldb** if present) into this subfolder.

Step 5: Retest the connection

- In Qlik Sense Desktop, edit the connection (✏️).
- Test the connection again.
- **Does the connection still function correctly?** (Answer = No, ⚠️ Test failed.)

Step 6: Reconfigure the connection

- Edit the connection string by changing the *Data Source=* path at the end to include the subfolder "Company".   " … `Data Source=C:\Data\Company\SOP.mdb;]` "
- Test the connection once again.
- **Does the connection now function correctly?** (Answer = Yes ✔️ Test successful.)
- Change the connection name from "SOP" to "SOPtrans".

Step 7: Create script to extract the orders table

- Use the **Select data** (📋) option for the *SOPtrans* connection to create the script to extract the *orders* table data from the SOP.mdb database, and click the **Insert script** button.
- Note the line to open the database `LIB CONNECT TO` `'SOPtrans'` at the top.
- Create a table label: "Orders:".
- Add inline comment: "// ** Load data from orders ** ".

Step 8: Load the data

- Use the **Load data** button to execute the script and load data into the Qlik Sense application data model.
- Check the **Data Model Viewer** (🔗) to ensure the table exists.

Step 9: Add a second section, connection and table load

- Create an additional new section, titled: "Employees".
- Create a second OLE DB data connection, using Jet 4.0 as the provider.
- Use the drive letter, folder path and filename name for *Employees.mdb.*
- Name the connection, "Employees".
- Use the Select data (📋) option for this new connection to create script to extract the only user table, *Employees,* and click the **Insert script** button.

Step 10: Label the table and comment

- Create a table label for the new table loading script, "Employees:".

- Create a suitable inline comment, such as: "// ** Load Employees ** ".

- **Load data**.

Step 11: Recreate a common mistake…

- Comment out the line LIB CONNECT TO 'Employees'; using //.
  (You could also use the **Comment/uncomment** toolbar button (//) or keyboard shortcut *CTRL+K.*)

- Reload the data using the **Load data** button.

- Observe the error text: "**The Microsoft Jet database engine cannot find the input table or query 'Employees'…**".

Step 12: …and fix it

- Return to the **Data load editor**.

- Uncomment the LIB CONNECT TO 'Employees'; line to re-enable it.

- Reload the data using the **Load data** button.

- Check the **Data Model Viewer** to ensure two tables from two databases have been loaded and are associated.

Other opportunities to learn more

If you have Microsoft Access installed, feel free to utilize either Employees.mdb or SOP.mdb to test out inserting, updating or deleting data values and reloading the data in Qlik Sense.

Alteration of the underlying database structure is more dangerous, but this training environment is the perfect sandbox to test out what might happen when the structure of a database changes and the Qlik Sense data load script attempts to load fields or entire tables which are no longer present in the source database.