MySQL for Excel

Abstract

This is the MySQL for Excel Reference Manual. It documents MySQL for Excel 1.3 through 1.3.6. Much of the documentation also applies to the previous 1.2 series.

For notes detailing the changes in each release, see the MySQL for Excel Release Notes.

For legal information, see the Legal Notices.

For help with using MySQL, please visit either the MySQL Forums or MySQL Mailing Lists, where you can discuss your issues with other MySQL users.

For additional documentation on MySQL products, including translations of the documentation into other languages, and downloadable versions in variety of formats, including HTML and PDF formats, see the MySQL Documentation Library.

Document generated on: 2016-09-16 (revision: 49023)

Table of Contents

Chapter 1 Introduction

MySQL for Excel enables you to work with a MySQL database from within Microsoft Excel. MySQL data can be imported into Excel, Excel data can be exported into MySQL as a new table or appended to a current table, and MySQL for Excel enables you to edit the MySQL data directly from within Excel.

External Resources That Supplement the Documentation

- **Release Notes**: for notes detailing the changes in each release of MySQL for Excel, see MySQL for Excel Release Notes.
- **Support Forum**: visit the MySQL for Excel forum for additional MySQL for Excel help and support.
- **Video Introduction**: for a short introductory to MySQL for Excel video, see Video: MySQL for Excel Introduction on the official MySQL YouTube channel.
- **Developer Blog**: developers of MySQL for Excel post related blog posts on the official MySQL on Windows Blog.

Chapter 2 Installation

MySQL for Excel is a product for Windows, and it is installed with MySQL Installer. And typically you will not be required to install or configure additional tools to use MySQL for Excel.

MySQL for Excel Requirements

The MySQL Installer installation process will check if these requirements are met, or notify you if further action is required before proceeding with the installation.

- *.NET Framework 4.0* (Client or Full Profile).
- *Microsoft Office Excel* 2007 or greater, for Microsoft Windows.
- *Visual Studio 2010 Tools for Office Runtime*, and MySQL Installer may install this for you.

Note

This requirement is different than *Office Developer Tools for Visual Studio*, which is not a substitute.

- An available MySQL Server connection.

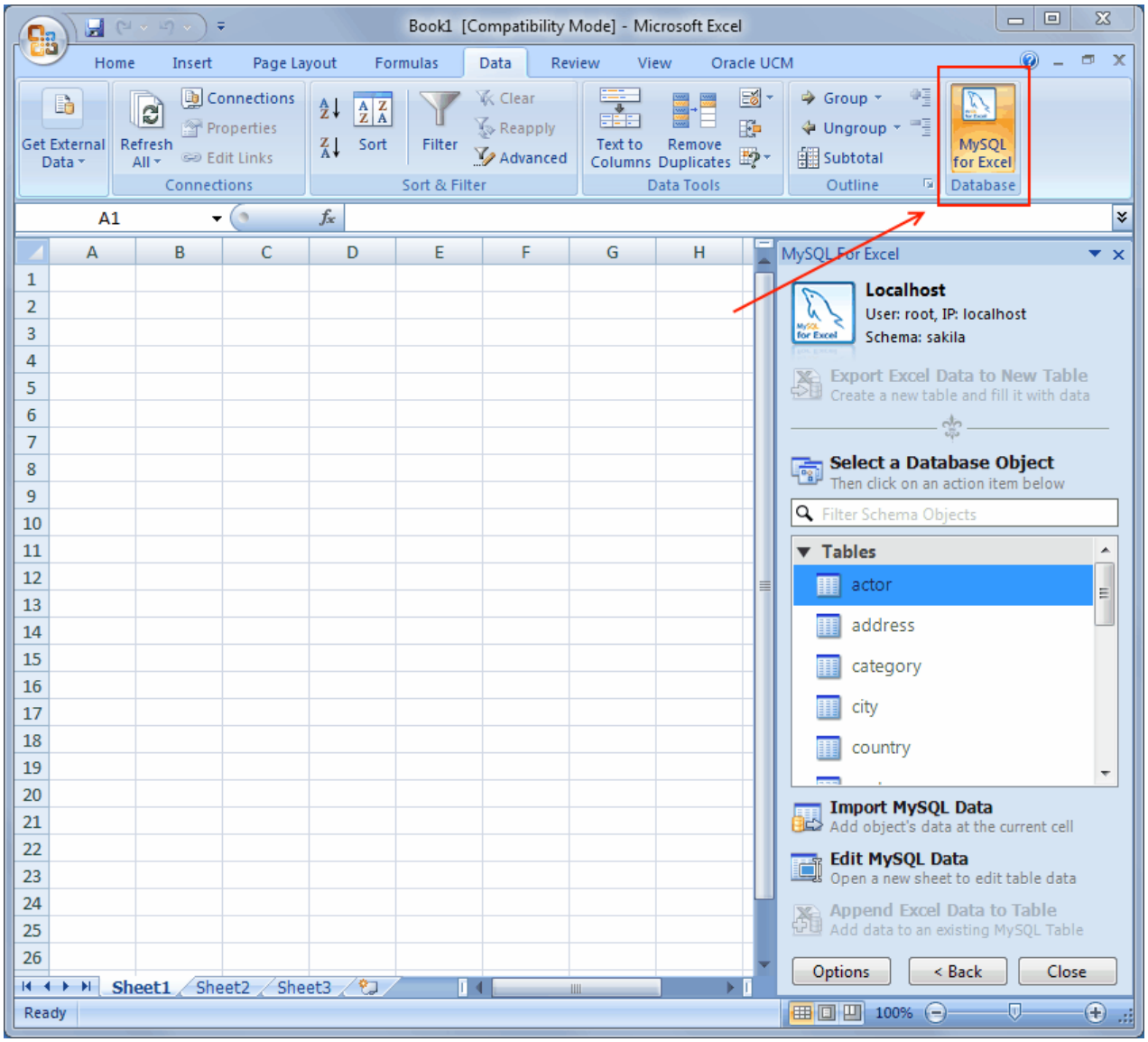
MySQL for Excel Download

Either install MySQL for Excel using the MySQL Installer for Windows (a system that manages installations and updates for all MySQL products on Windows), or download and execute the standalone file. The download links are as follows:

- **MySQL Installer**: Download and execute the MySQL Installer MSI file. Select the MySQL for Excel product and then proceed with the installation. See the MySQL Installer manual for additional details. This is the recommended approach.
- **Standalone**: Download and execute the MySQL for Excel standalone MSI file.

MySQL for Excel is loaded and executed by selecting the **Data** menu tab in Excel, and then choosing the "MySQL for Excel" Database icon. This opens a new Excel sidebar with the available MySQL for Excel options. The navigation bar with the MySQL for Excel icon is shown in the following screenshot:

Figure 2.1 The MySQL for Excel navigation bar



Chapter 3 Configuration

Table of Contents

This section is divided into two overlapping topics; configuring the global options, and managing the MySQL connections.

3.1 Global Options and Preferences

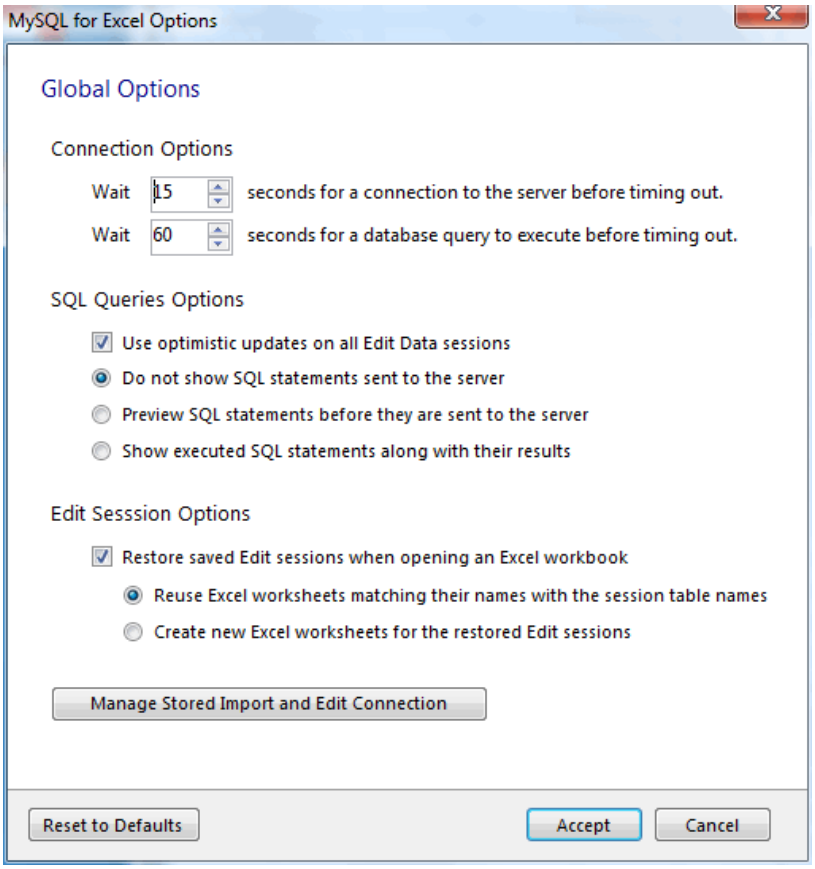
Each action, such as **Import MySQL Data**, has its own set of options. The buttons on these pages include:

Actions include:

- Clicking **Accept**: Saves option changes to your host, and preserves these changes across all sessions and future Excel instances.
- Clicking **Reset to Defaults**: Resets all option values on the current options window to their default settings. Click **Accept** to save the changes.

A set of "global" options affect the entire plugin, as described here:

Figure 3.1 The MySQL for Excel configuration: Global Options



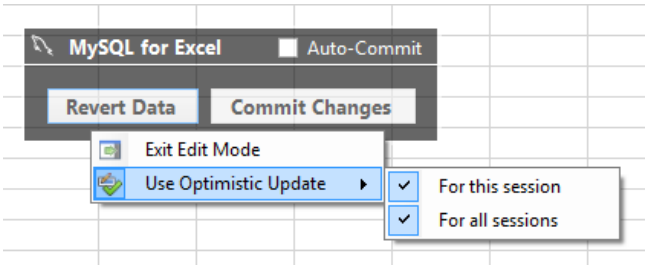
- Connection Options:

- *Wait [ ] seconds for a connection to the server before timing out*. Defaults to 15.
- *Wait [ ] seconds for a database query to execute before timing out*. Defaults to 60.

- SQL Queries Options:
  - *[ ] Use optimistic updates on all Edit Data sessions*. This option helps prevent unintentional data overwrite, in that it checks for external edits before committing your changes. For example, between the time you loaded the data into Excel, made changes in Excel, and committed, a different user could have edited the same cells elsewhere in MySQL using MySQL Workbench or some other means. The optimistic updates feature checks for these changes, and notifies you accordingly.

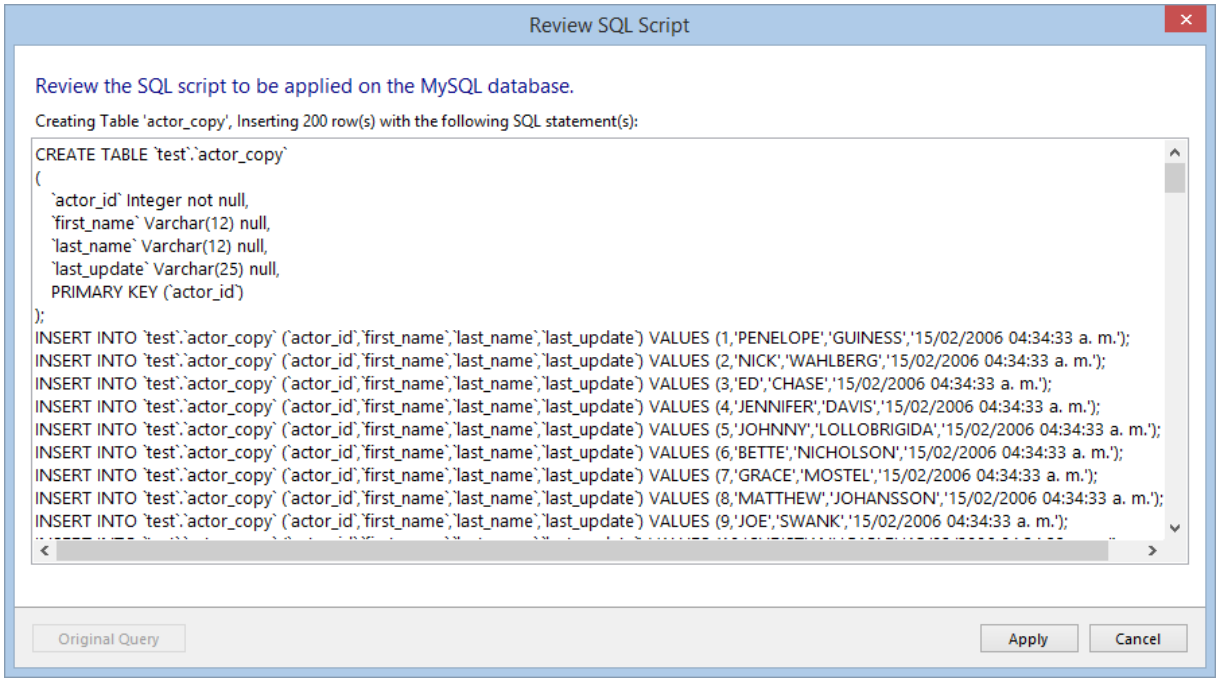
    Optimistic updates can also be configured at runtime for all Edit Sessions, or for a specific edit session by right-clicking the **Edit Session** floating dialog and choosing the desired **Use Optimistic Update** option, as demonstrated below:

    Figure 3.2 Optimistic Updates: Configuring at Runtime

    This option is enabled by default.
  - *( ) Do not show SQL statements sent to the server*: When enabled, SQL statements are now displayed, and only their results are displayed in the information dialog. Enabled by default.
  - *( ) Preview SQL statements before they are sent to the server*: When enabled, it adds an extra step to the Create Schema, Export Data, Append Data and Edit Data operations before a statement is committed to the server. It enables the "Review SQL Script" dialog, as shown below for an "Export Data" operation:

Figure 3.3 The MySQL for Excel configuration: Preview Option



From here you can modify the SQL statements before they are executed, which also enables the **Original Query** button. If clicked, it will revert all modifications to the script to restore the SQL to its original form (when the dialog was first opened).

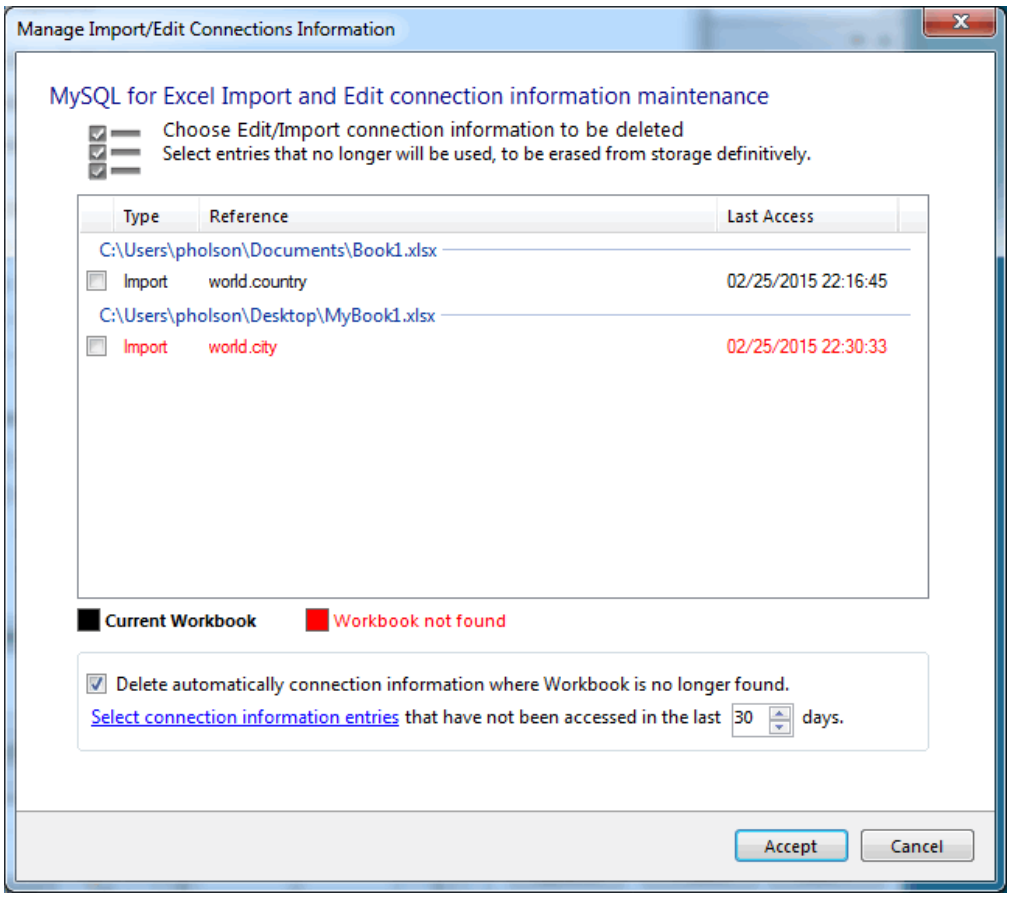
This option is disabled by default.

- *( ) Show executed SQL statements along with their results*: When enabled, SQL statements are first executed and then the information dialog includes both the results and the executed statements. This is helpful when reviewing the recently executed queries when comparing the results.

  This option is disabled by default.

- Edit Session Options:
  - [ ] Restore saved Edit sessions when opening an Excel workbook. Enabled by default.
  - ( ) Reuse Excel worksheets matching their names with the session table names. Enabled by default.
  - ( ) Create new Excel worksheets for the restored Edit sessions. Disabled by default.
- **Manage Stored Import and Edit Connections**: See a list of saved Excel files with linked MySQL connections.

Figure 3.4 MySQL for Excel: Manage Stored MySQL Connections



This lists the connected Excel worksheets that are known to MySQL for Excel. From here you can view these connections, and optionally delete them. By default, clicking **Apply** will delete connections to missing worksheets but this behavior is configurable. Additionally, clicking the *Select connection information entries* link checks (selects for deletion) books that you have not accessed for n days, where n defaults to 30.

Note

This option was added in MySQL for Excel 1.3.0

Note

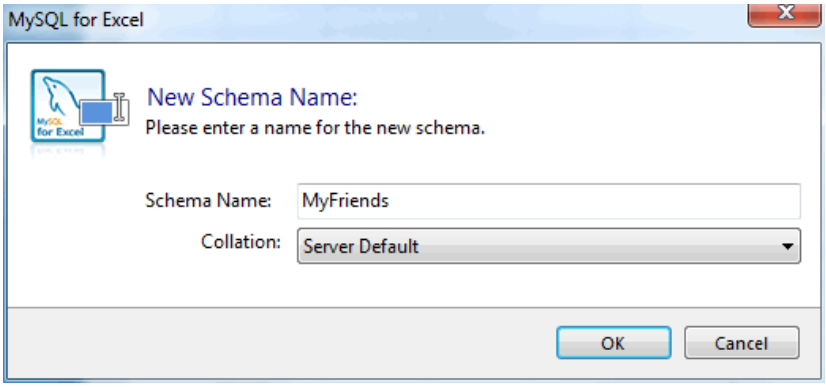
The options to automatically delete missing connections, or delete connections not accessed for n days, were added in MySQL for Excel 1.3.4.

For additional information about managing MySQL connections using MySQL for Excel, see Section 3.2, “Managing MySQL Connections”.

Adding a New Schema

Select a MySQL connection and then click **Create New Schema** from the MySQL for Excel toolbar to add a new and empty MySQL schema.

Figure 3.5 MySQL for Excel: Adding a New Schema



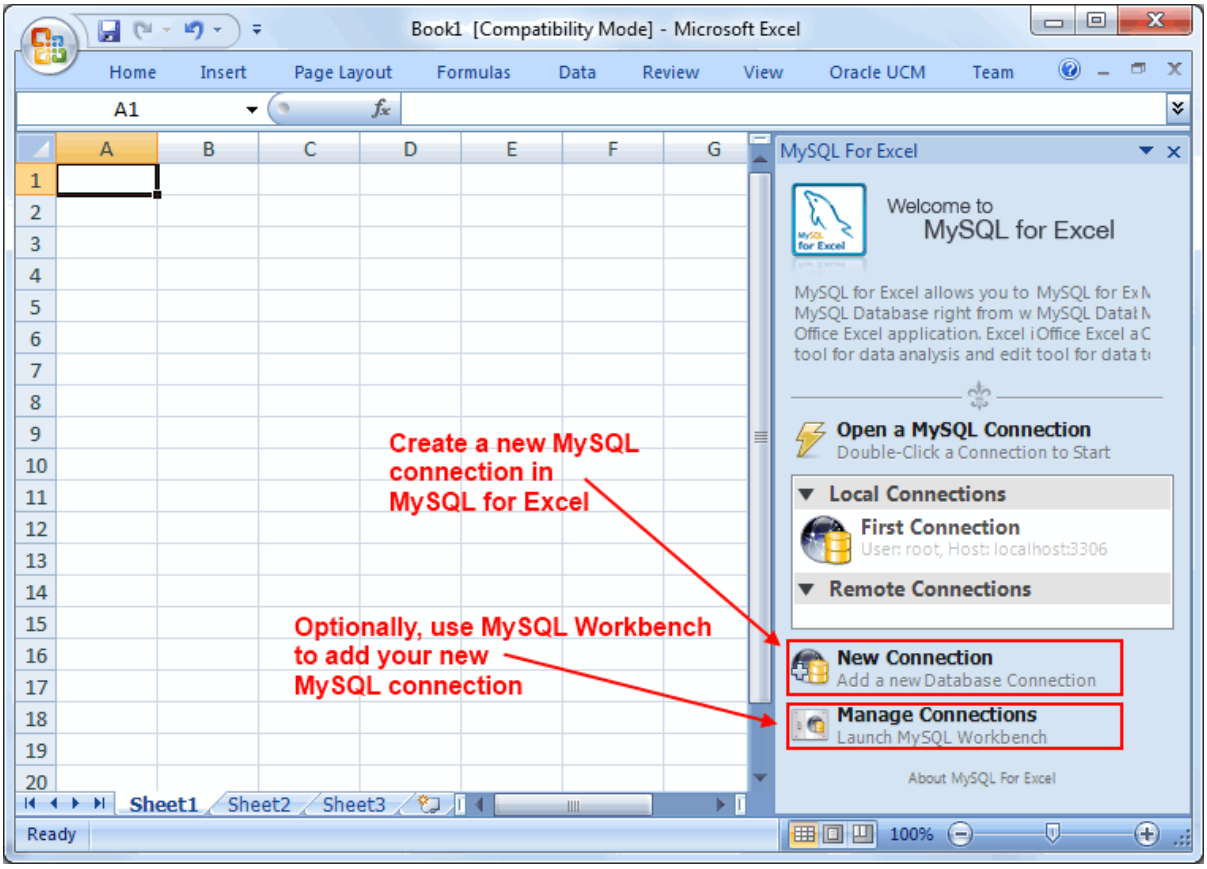
3.2 Managing MySQL Connections

MySQL for Excel shares its MySQL connections with MySQL Workbench, although it is optional to have MySQL Workbench installed. Creating and editing MySQL connections in either application will edit the MySQL connection information for both applications.

Adding MySQL Connections

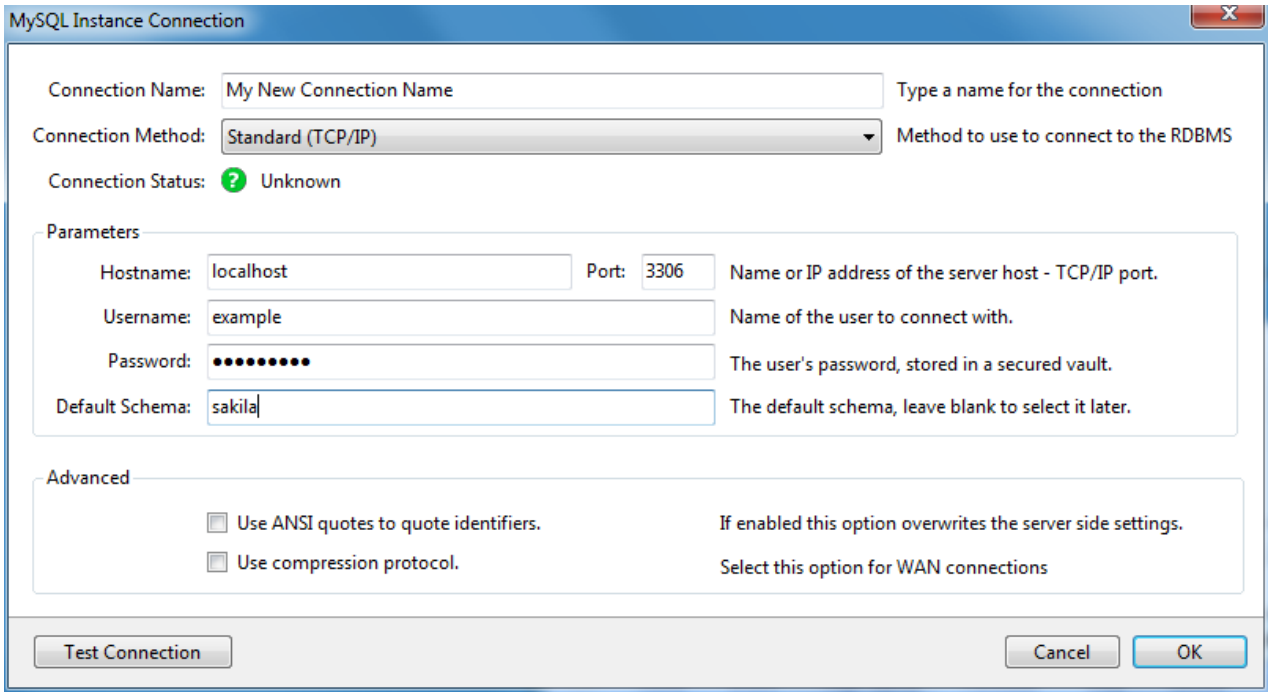
You can use MySQL for Excel or MySQL Workbench to add new MySQL connections.

Figure 3.6 MySQL for Excel: Add a New MySQL Connection



From Excel, click **New Connection** to open the new connection dialog as demonstrated in the following partially filled screenshot:

Figure 3.7 MySQL for Excel: Add a New MySQL Connection Dialog



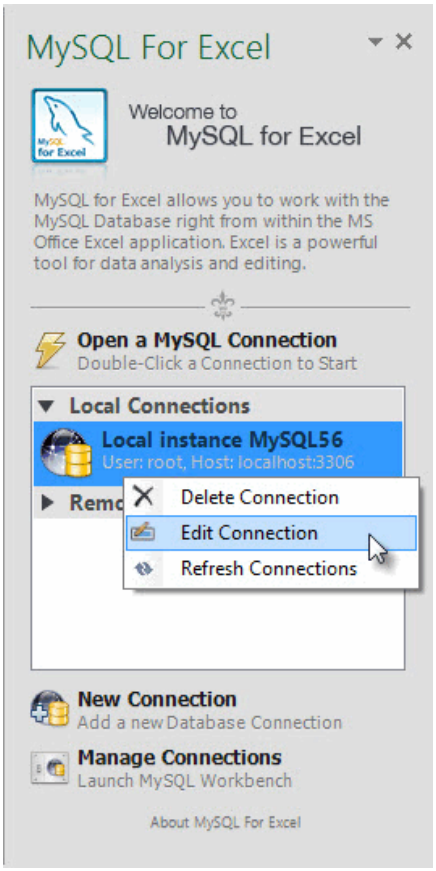
Fill out the connection details, click **Test Connection** to confirm the MySQL connection is valid, and click **OK** to save the new connection.

Editing MySQL Connections

Editing MySQL Connections in MySQL for Excel

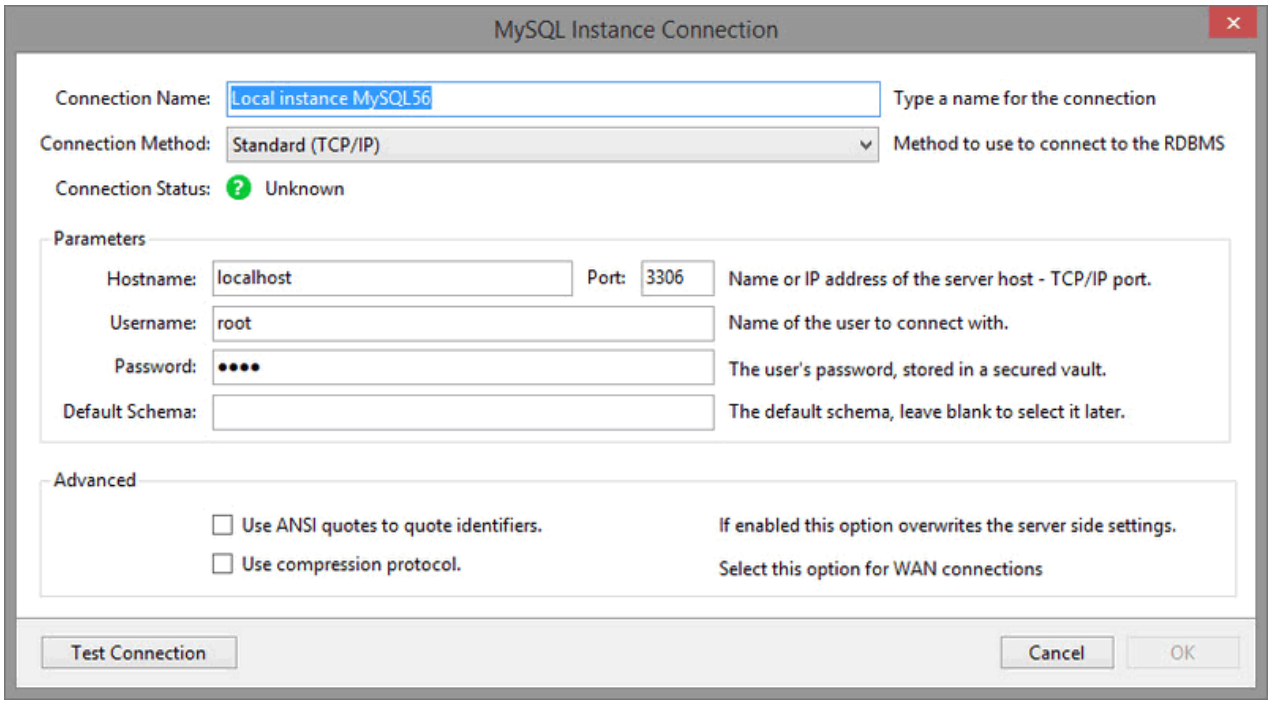
To edit a MySQL connection, right-click the connection you want to modify and select **Edit Connection** from the context menu, like so:

Figure 3.8 MySQL for Excel: Choosing a MySQL Connection to Edit



The MySQL connection edit dialog is similar to the edit dialog in MySQL Workbench. Configure the changes and click **OK** to save your changes:

Figure 3.9 MySQL for Excel: Editing a MySQL Connection



Editing MySQL Connections in MySQL Workbench

Optionally, you can edit your MySQL for Excel MySQL connections using MySQL Workbench. To do this, open MySQL Workbench, edit a MySQL connection, and then refresh the connection list in MySQL for Excel.

For information about editing MySQL connections in MySQL Workbench, see the MySQL Workbench documentation titled MySQL Connections.

Delete MySQL Connections

MySQL connections can be deleted from MySQL for Excel or MySQL Workbench.

Note

MySQL connections cannot be deleted if MySQL Workbench is open. To remove connections, you must first close MySQL Workbench.

To delete an edit or import connection from MySQL for Excel to a particular Excel worksheet, click **Options**, **Manage Stored Import and Edit Connection**, check the desired worksheets, and then click **Apply** to execute the delete action.

Chapter 4 Edit MySQL Data in Excel

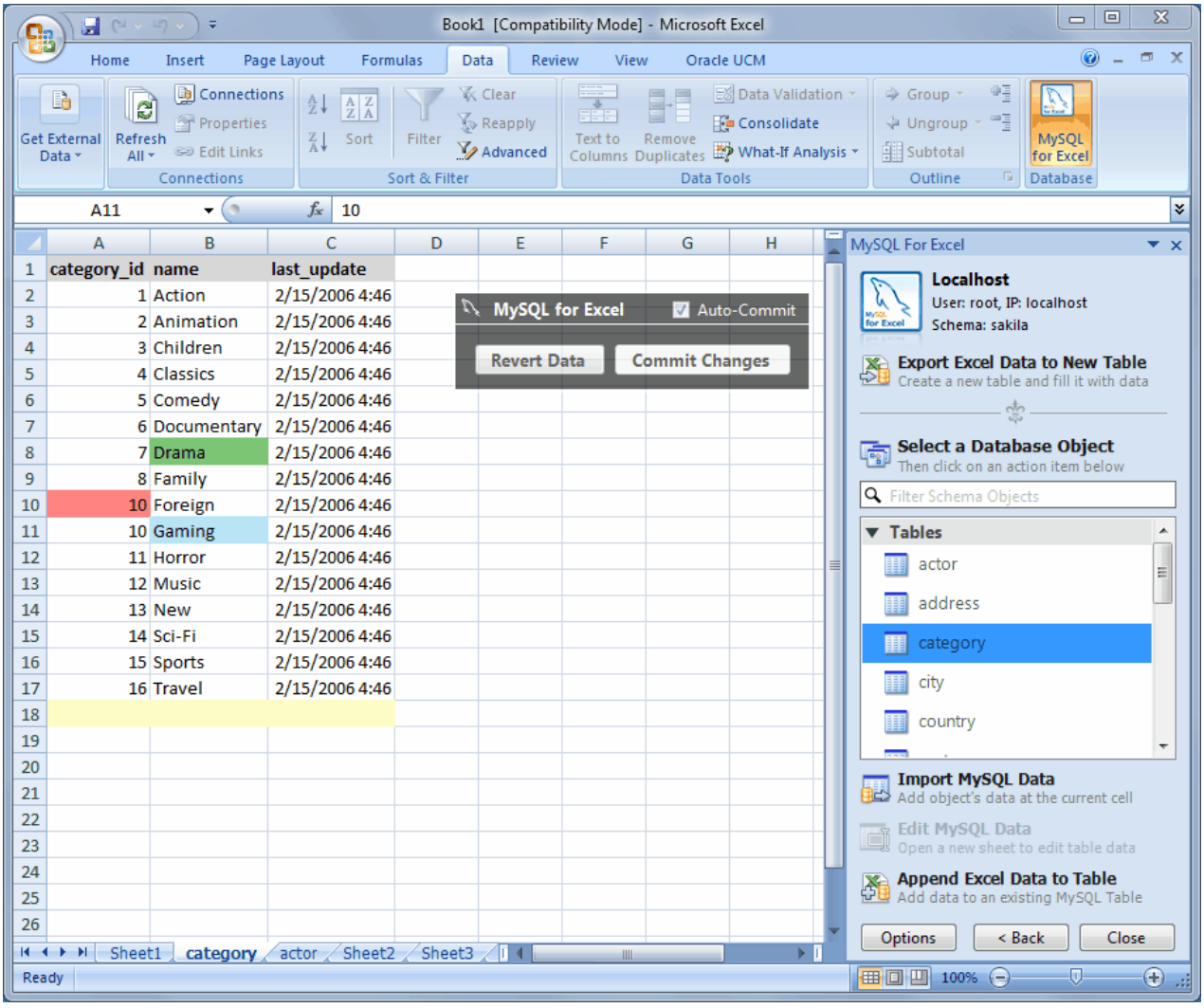
MySQL for Excel enables you to load and edit MySQL data directly from Microsoft Excel. Changes are immediately committed if the **Auto-Commit** option is enabled, or done manually by pressing **Commit Changes**.

The example below uses the `category` table of the example `sakila` database, but the screen will look the same for any table. Within MySQL for Excel, **Open a MySQL Connection**, click the `sakila` schema, **Next**, select the `category` table, click **Edit MySQL Data**, then choose **Import** to import the data into a new Microsoft Excel worksheet for editing.

Note

For additional information about the importing procedure, see Chapter 5, *Import MySQL Data into Excel*.

Figure 4.1 Editing table data with MySQL for Excel



category_id	name	last_update
1	Action	2/15/2006 4:46
2	Animation	2/15/2006 4:46
3	Children	2/15/2006 4:46
4	Classics	2/15/2006 4:46
5	Comedy	2/15/2006 4:46
6	Documentary	2/15/2006 4:46
7	Drama	2/15/2006 4:46
8	Family	2/15/2006 4:46
10	Foreign	2/15/2006 4:46
10	Gaming	2/15/2006 4:46
11	Horror	2/15/2006 4:46
12	Music	2/15/2006 4:46
13	New	2/15/2006 4:46
14	Sci-Fi	2/15/2006 4:46
15	Sports	2/15/2006 4:46
16	Travel	2/15/2006 4:46

The background color represents the status of each cell, and there are four distinct colors that are used while editing table data:

Note

The Green and Blue colors were switched in MySQL for Excel 1.2.0.

Table 4.1 Background cell colors

Color	Description
White	Default color for all cells. This is either the original data, or the data after **Refresh from DB** is clicked.
Green	Cells that were committed with success.
Blue	Cells that were modified but have not yet been committed.
Red	Cells that generated an error when a commit was attempted. An error dialog is also displayed while the commit is attempted.
Orange	Cells that had a commit attempted, but the commit failed due to detected changes from external sources. For example, a different user made a change to a field after it was imported into Excel. This is a feature of Optimistic Updates.
Yellow	Cells that accept new data. Data entered here is inserted into the MySQL table.

In our example, the green "Drama" field was changed and then committed first, then the blue "Gaming" field was changed but not committed, and then **Auto-Commit** was enabled before changing the "9" to a "10" in column 10, which generated an error because this commit would have added a duplicate value as primary key.

Chapter 5 Import MySQL Data into Excel

Table of Contents

Data can be imported from MySQL into a Microsoft Excel spreadsheet by using the **Import MySQL Data** option after selecting either a table, view, or procedure to import.

5.1 Choosing Columns To Export

By default, all columns are selected and will be imported. Specific columns may be selected (or unselected) using the standard Microsoft Windows method of either **Control** + `Mouse click` to toggle the selection of individual columns, or **Shift** + `Mouse click` to select a range of columns.

The background color of a column shows the status of each column. The color white means that the column has been selected, and therefore it will be imported. Conversely, a gray background means that the column will not be imported.

Right-clicking anywhere in the preview grid opens a context-menu with either a `Select None` or `Select All` option, depending on the current status.

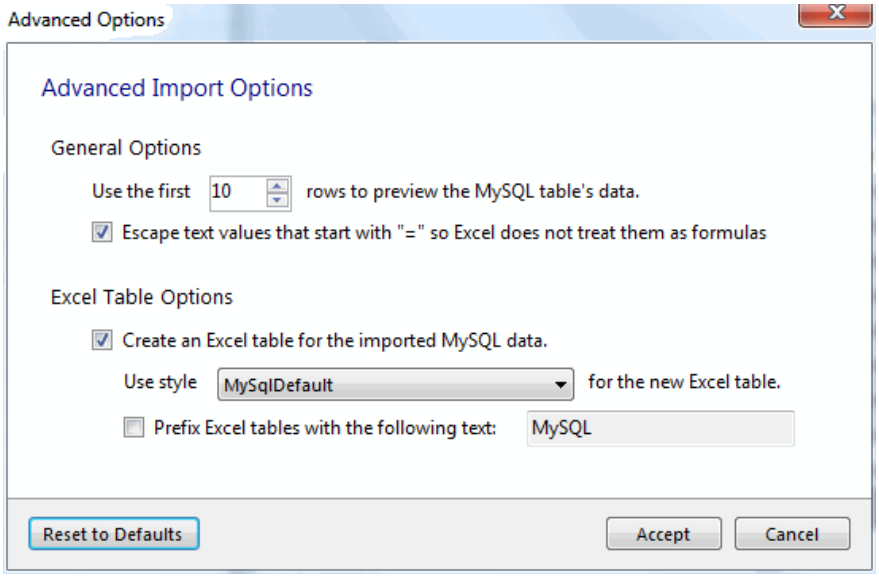
5.2 Importing a Table

The dialog while importing a table includes the following options:

- [ ] **Include Column Names as Headers**: Enabled by default, this inserts the column names at the top of the Microsoft Excel spreadsheet as a "headers" row.
- [ ] **Limit to ___ Rows and Start with Row ___**: Disabled by default, this limits the range of imported data. The `Limit to` option defaults to `1`, and defines the number of rows to import. The `Start with Row` option defaults to `1` (the first row), and defines where the import begins. Each option has a maximum value of COUNT(rows) in the table.
- [ ] **Add Summary Fields**: Disabled by default, this option adds a summary field to each column. For additional information, see Section 5.5, “Adding Summary Fields”.

5.3 Import: Advanced Options

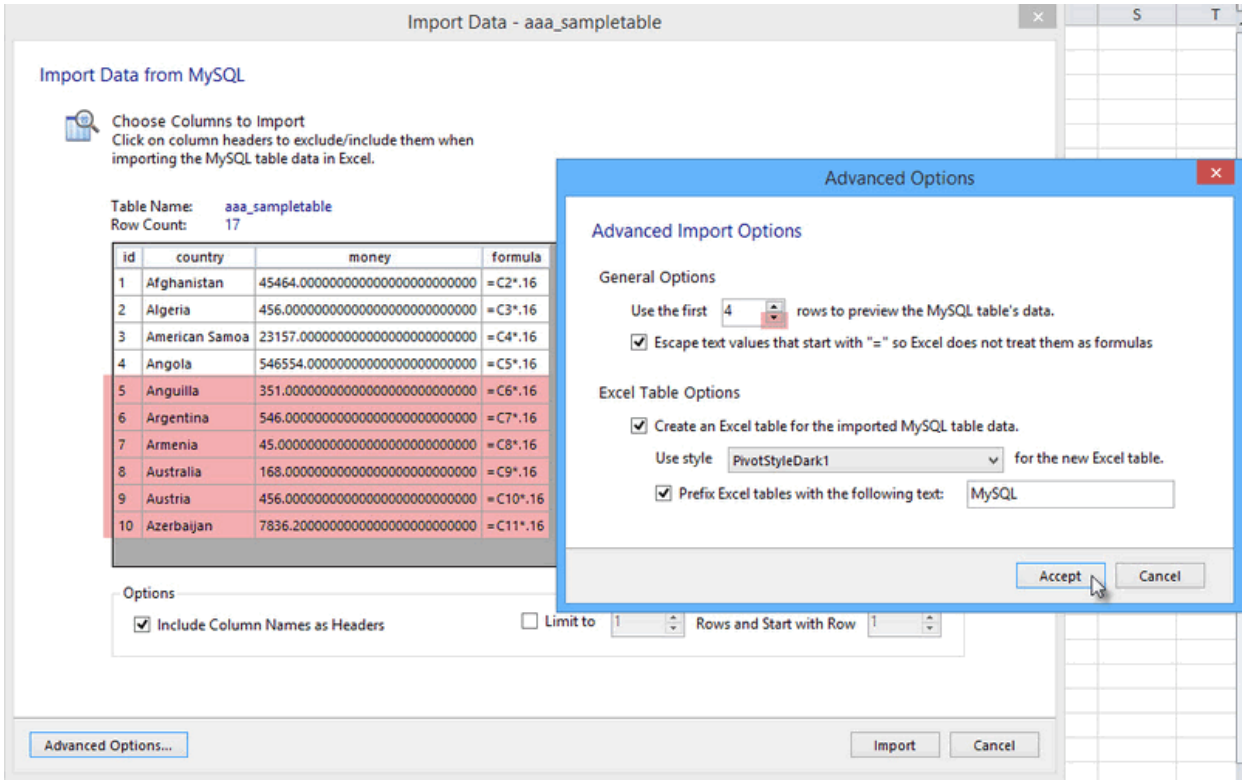
Figure 5.1 Importing table data with MySQL for Excel: Advanced options



General Options:

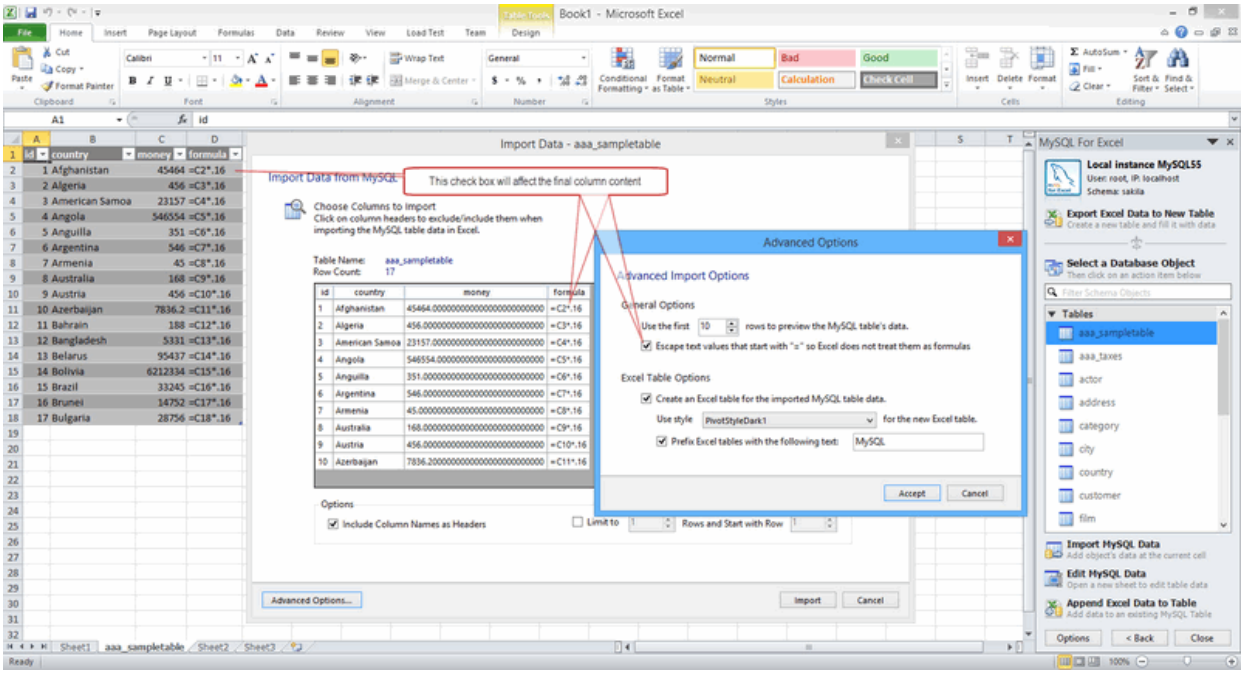
- Use the first [*10*] rows to preview the MySQL tables data. This affects the preview step in the import process, and defaults to 10.

Figure 5.2 MySQL for Excel: Preview



- [ ] Escape text values that start with "=" so Excel does not treat them as formulas, and is enabled by default. This option may not reflect any differences in the preview because it is only applied after the data is imported into the Excel Worksheet.

Figure 5.3 MySQL for Excel: Escape "=" (formulas)

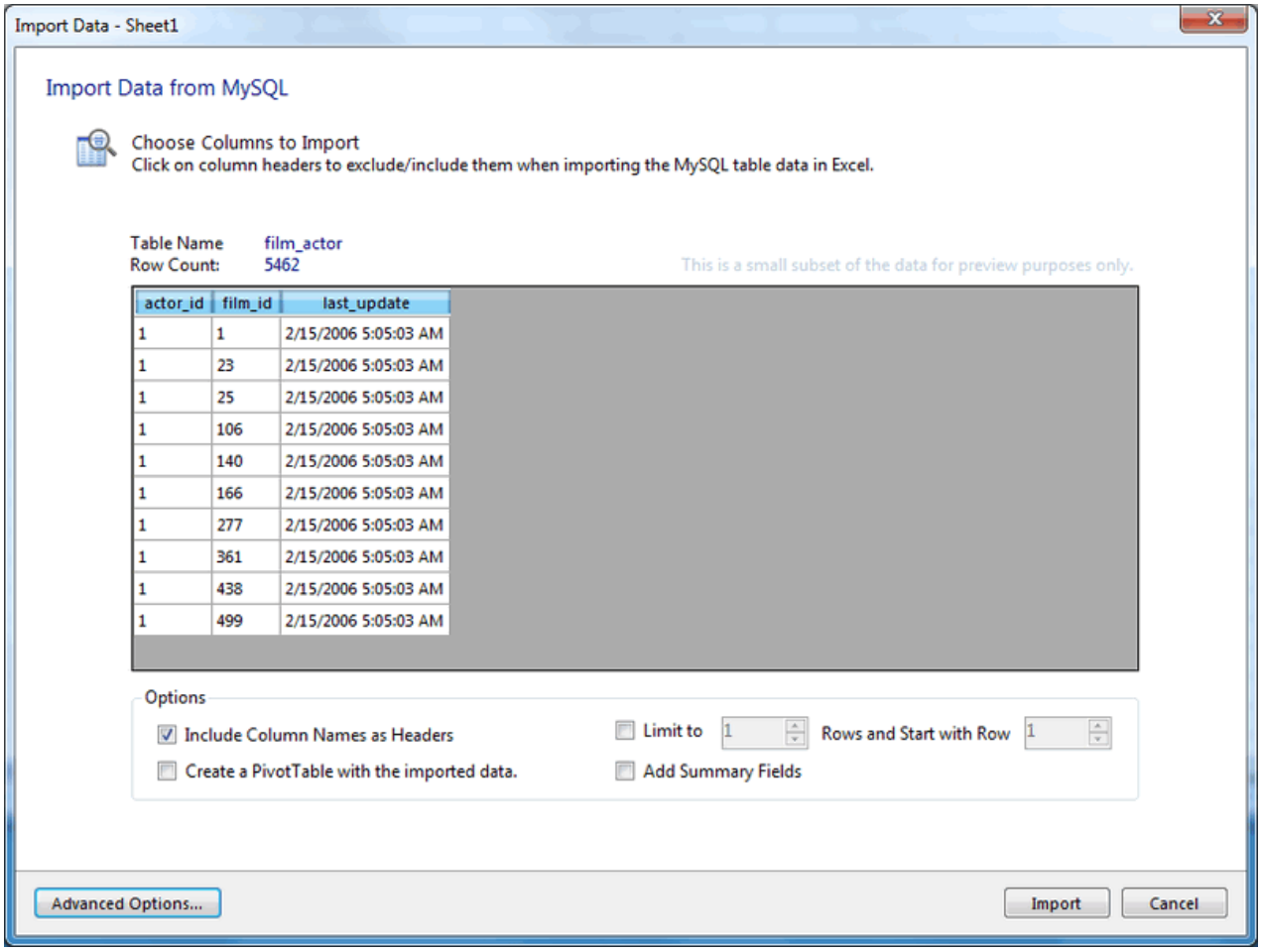


Excel Table Options:

- [ ] Create an Excel table for the imported MySQL table data. Enabled by default.
- Use style [ ] for the new Excel table. Defaults to `MySqlDefault`.
- [ ] Prefix Excel tables with the following text: _____. Disabled by default.

Importing a table displays a dialog similar to the following:

Figure 5.4 Importing table data with MySQL for Excel



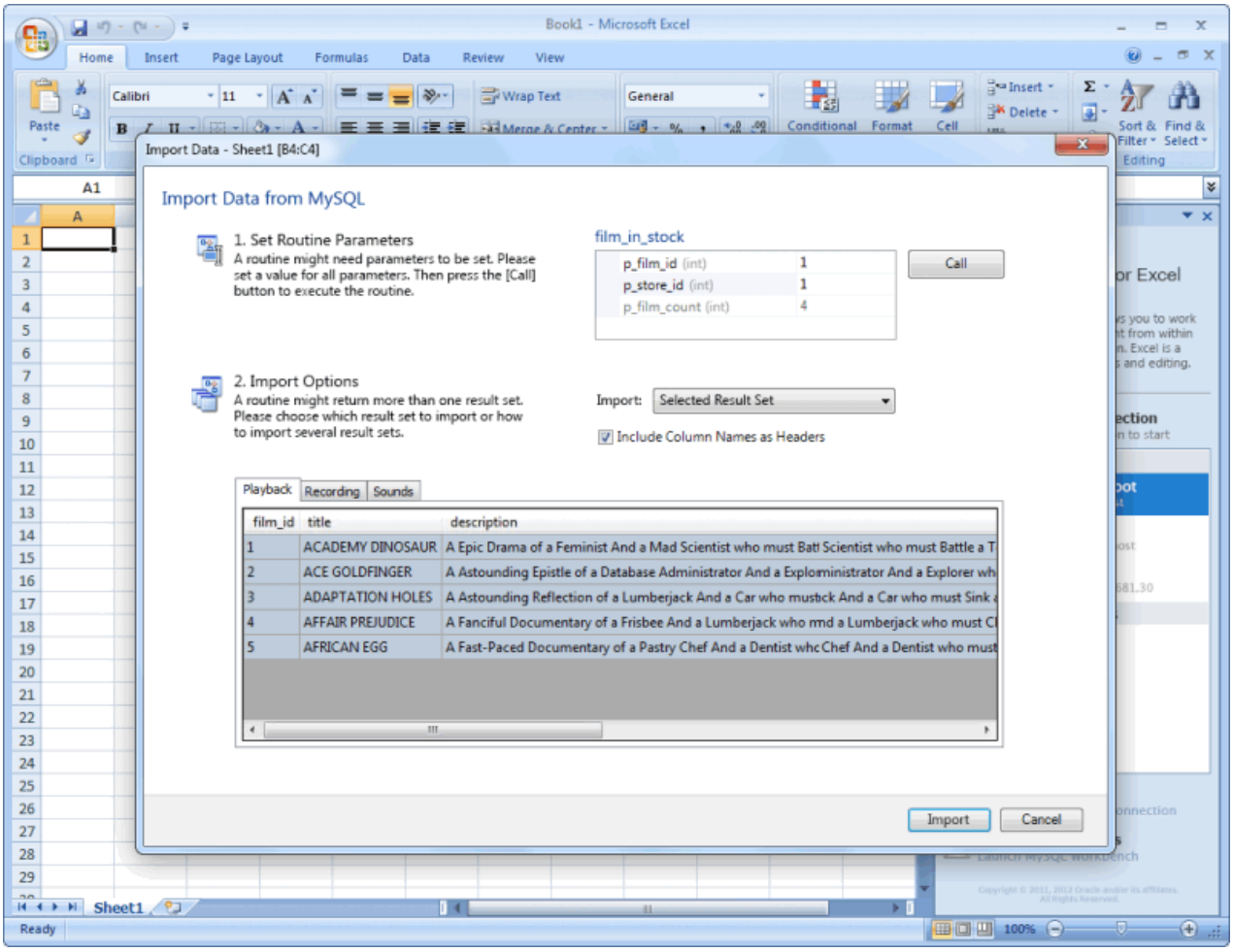
5.4 Importing a View or Procedure

Importing a view or procedure displays a similar dialogue, but with the following options:

- **Include Column Names as Headers**: Enabled by default, this will insert the column names at the top of the Excel spreadsheet as a "headers" row.
- **Import**: Because a procedure might return multiple result sets, the import options include:
  - **Selected Result Set**: Imports the selected tab sheet. This is the default behavior.
  - **All Result Sets - Arranged Horizontally**: Imports all result sets into the Excel Worksheet horizontally, and inserts one empty column between each result set.
  - **All Result Sets - Arranged Vertically**: Imports all result sets into the Excel Worksheet vertically, and inserts one empty row between each result set.

For example, a dialogue like the following is displayed after importing a procedure and pressing the **Call** button to invoke the stored procedure:

Figure 5.5 Importing called stored procedure data with MySQL for Excel



5.5 Adding Summary Fields

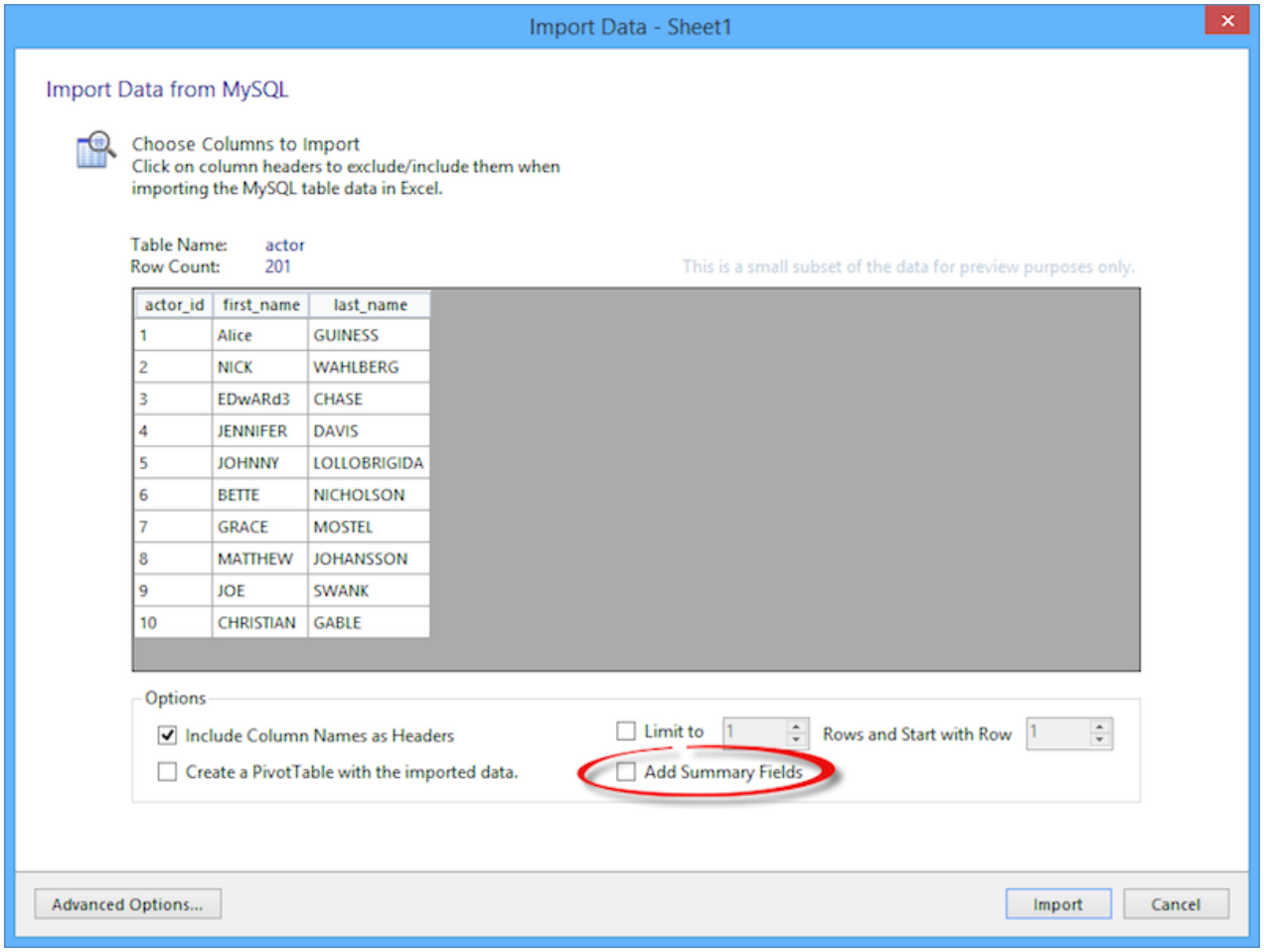
Summary fields are calculated fields, and this option adds summary related functions to each of the imported columns. These fields are added to the last row of the imported table data, and the dropdown of choices includes Average, Sum, Min, and Max.

Note

This feature was added in MySQL for Excel 1.3.0.

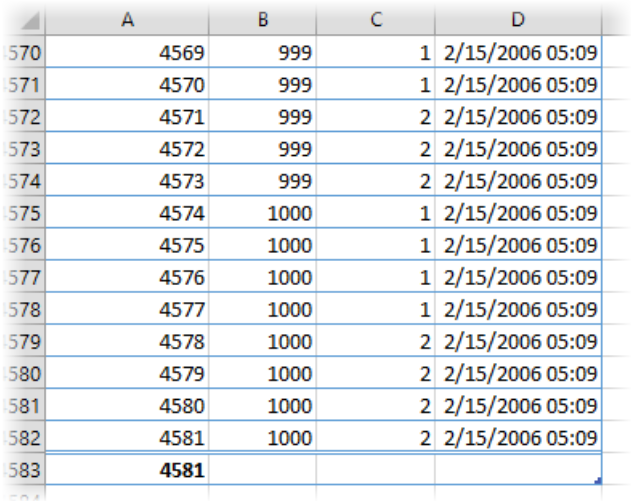
The **Add Summary Fields** option (disabled by default) is listed on the import dialog:

Figure 5.6 The 'Add Summary Fields' option



Enabling this option adds a row of summary fields for the appropriate columns in your imported data. Notice the newly created row on the bottom:

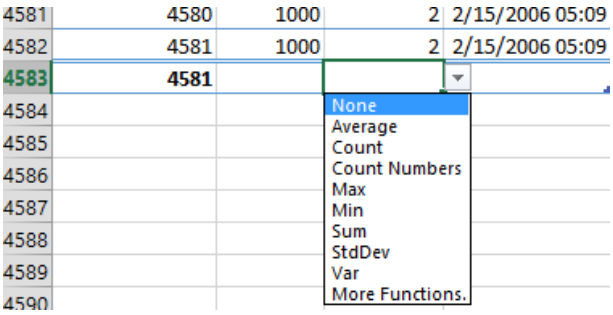
Figure 5.7 The new 'Add Summary Fields': the new row



	A	B	C	D
570	4569	999	1	2/15/2006 05:09
571	4570	999	1	2/15/2006 05:09
572	4571	999	2	2/15/2006 05:09
573	4572	999	2	2/15/2006 05:09
574	4573	999	2	2/15/2006 05:09
575	4574	1000	1	2/15/2006 05:09
576	4575	1000	1	2/15/2006 05:09
577	4576	1000	1	2/15/2006 05:09
578	4577	1000	1	2/15/2006 05:09
579	4578	1000	2	2/15/2006 05:09
580	4579	1000	2	2/15/2006 05:09
581	4580	1000	2	2/15/2006 05:09
582	4581	1000	2	2/15/2006 05:09
583	**4581**			

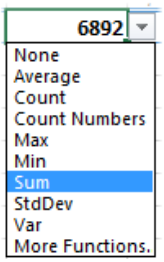
Select the row to reveal a down arrow, and click it to display a set of summary options:

Figure 5.8 The 'Add Summary Fields' row: choices



For example, choosing **Sum**:

Figure 5.9 The 'Add Summary Fields' row: sum example



Adjust each summary field accordingly.

5.6 Creating PivotTables

A PivotTable can be created from imported MySQL tables, views, stored procedures, or the entire Excel Data Model.

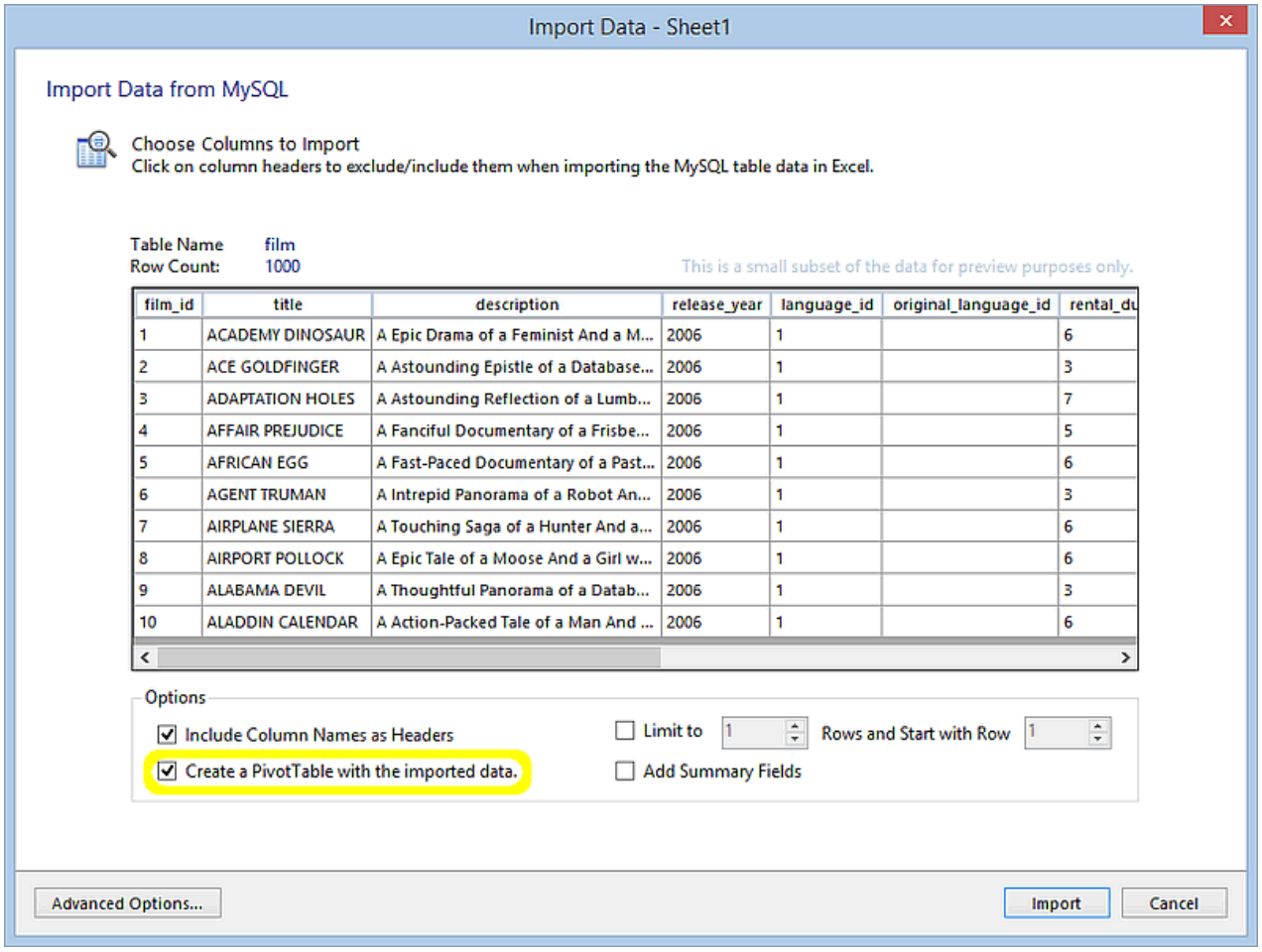
Note

This feature was added in MySQL for Excel 1.3.0.

An Excel PivotTable report summarizes and provides a visual representation of data in many different ways. It is a native Excel feature, see PivotTable reports 101 for additional information about Excel PivotTables.

Our example covers a simple use case where an empty PivotTable is created from an imported MySQL table. This example uses the "film" table of the "Sakila" database. To create the PivotTable, select the "film" table from the database object's selection panel and then click **Import MySQL Data**. On the **Import Data** dialog, check the **Create a PivotTable** before pressing **OK** to execute the operation.

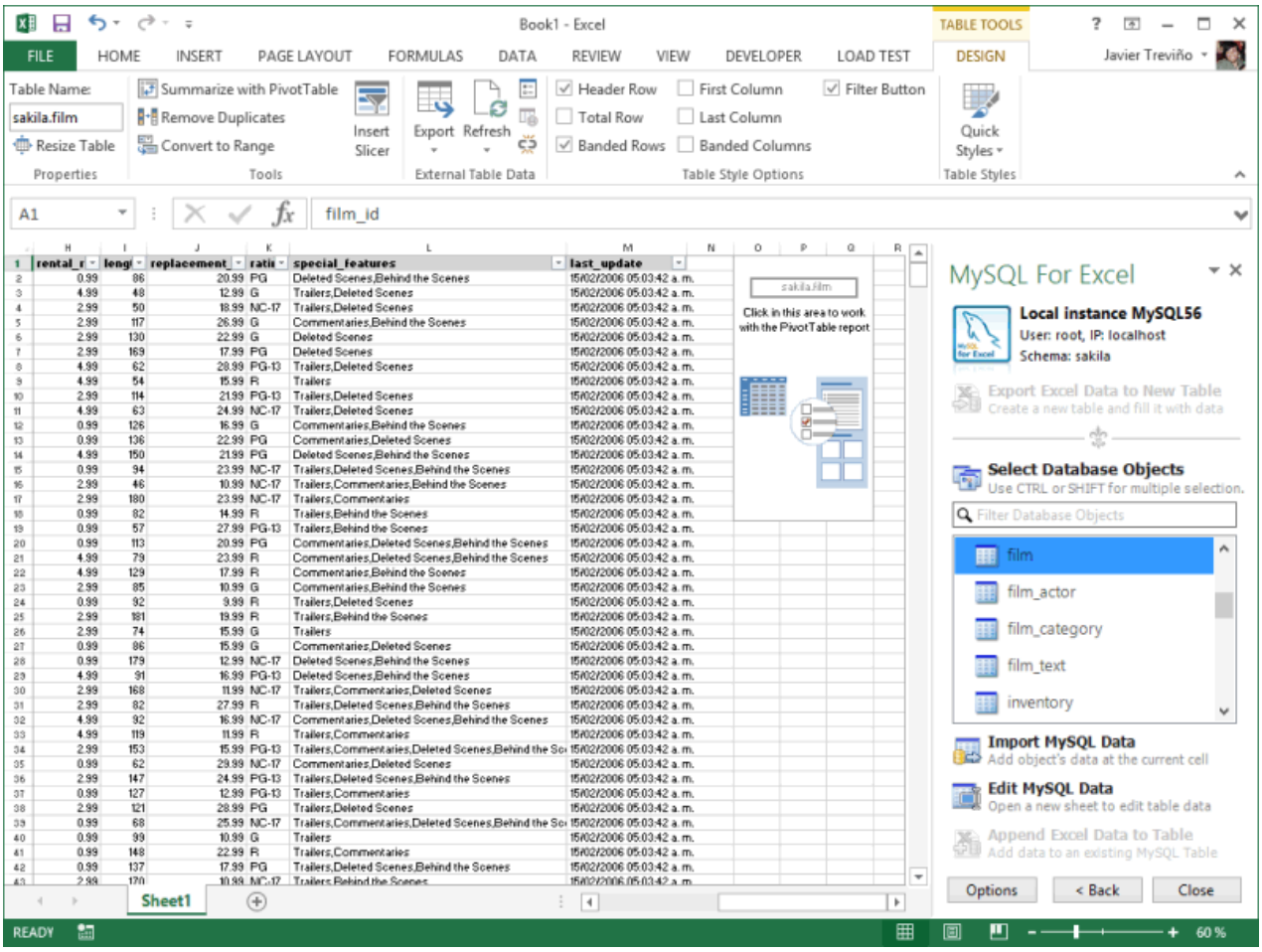
Figure 5.10 Option: Create a PivotTable with the imported data



When the **Create a PivotTable with the imported data** option is checked, an empty PivotTable (or a PivotTable placeholder) is inserted just to the right of the imported data. The PivotTable name follows the same naming rules used for Excel tables created from tje imported data, but PivotTables can be created with or without enabling the **Create an Excel table for the imported MySQL data** advanced option. That means a PivotTable can be created from an imported Excel range (if the aforementioned advanced option is off), or from an imported Excel table (if the option is on).

Click **Import** to dump the film table's data to an active Excel Spreadsheet, and this also creates a PivotTable for that data.

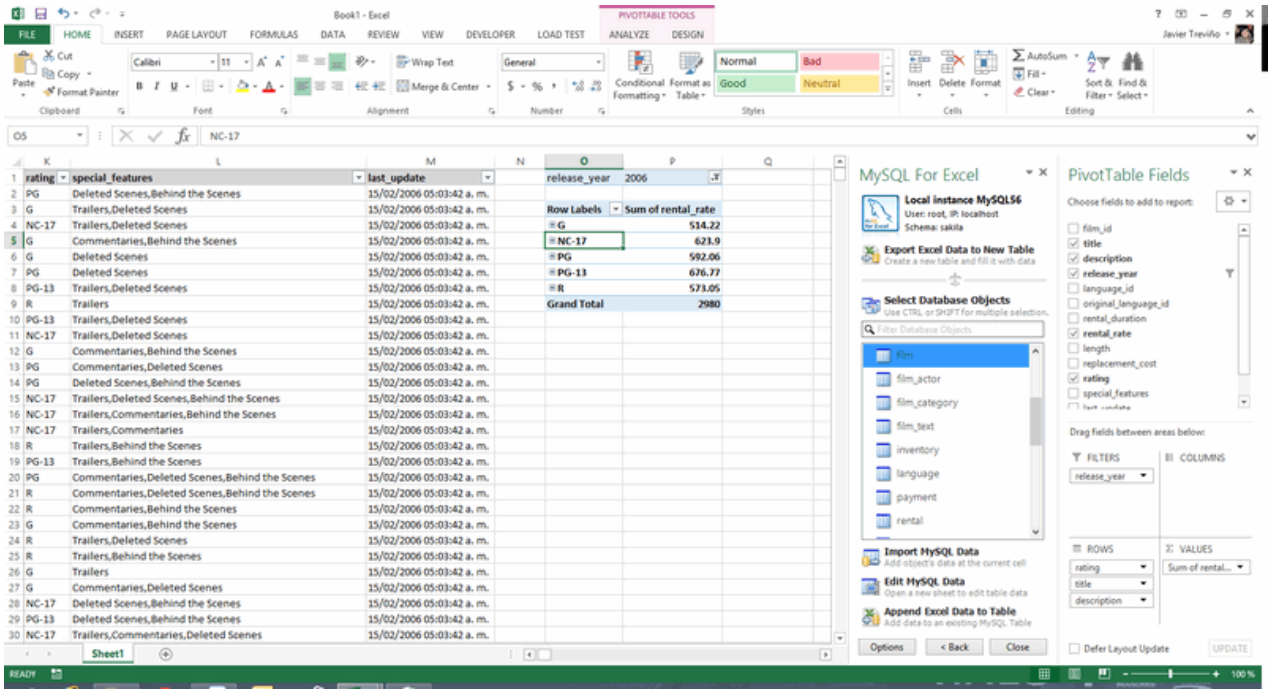
Figure 5.11 PivotTable Example: Empty PivotTable



Clicking the PivotTable opens a PivotTable Fields panel to next to the MySQL for Excel panel, and from here you can select fields you want to summarize in the PivotTable report. Drag and drop fields from the list to any of the FILTERS, COLUMNS, ROWS, or VALUES areas, depending on the visualizations you want in the report. The report is completely dynamic, meaning that you can change the views by moving fields around the areas until you see the visualization you need for your PivotTable report.

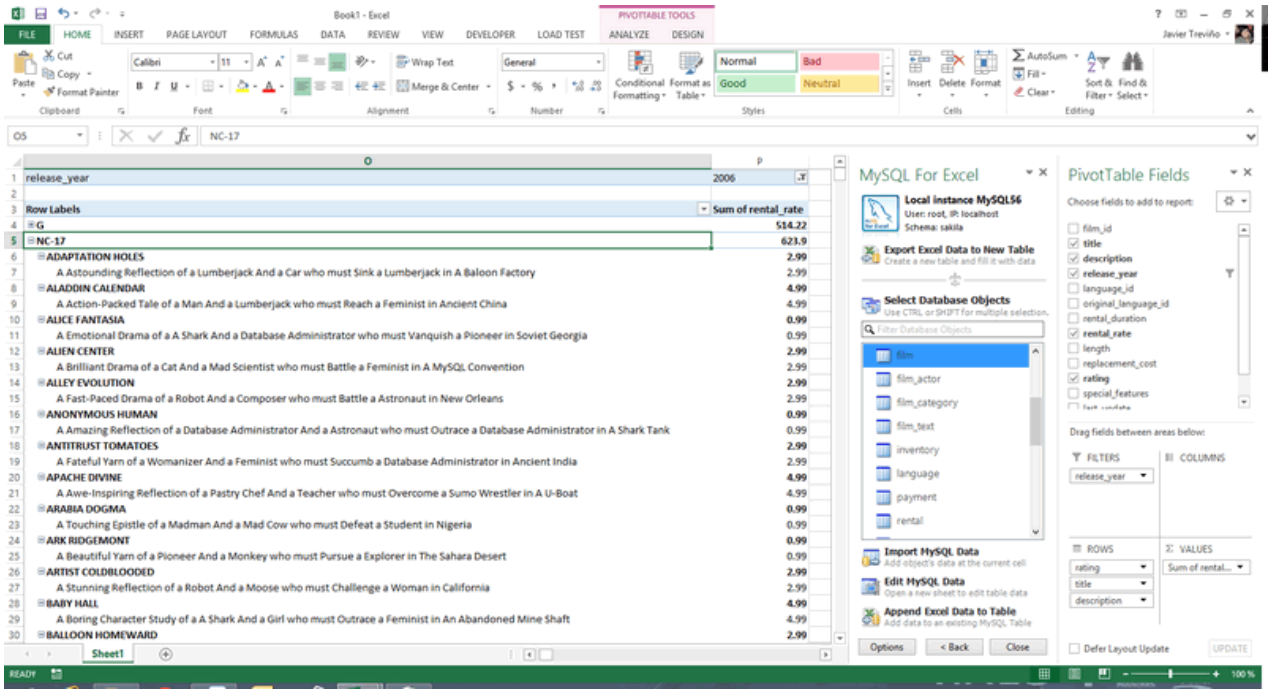
Below is an example PivotTable report using the sakila.film table we imported in our previous example. This report includes a filter by release_year, and it summarizes the rental_rate values while also grouping the data by values in the rating column.

Figure 5.12 PivotTable Example: Film Ratings



Expanding one of the groups reveals its values from the title and description columns.

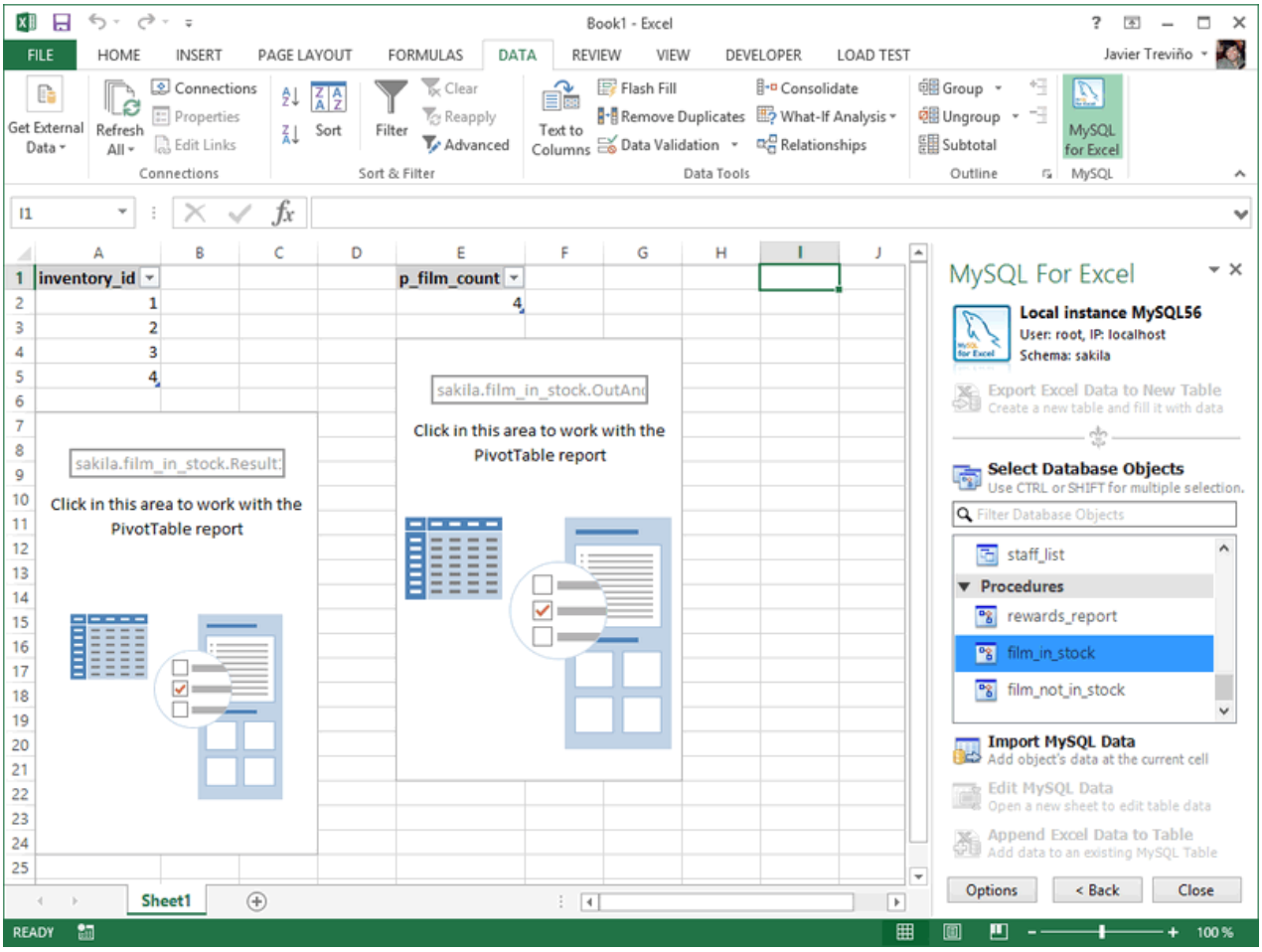
Figure 5.13 PivotTable Example: Expanded Group



We can also do this with data coming from a MySQL view or stored procedures. The only difference is that for stored procedures we can create a PivotTable for each of the imported result sets returned by the procedure's call. Take the following screenshot as an example where we have the film_in_stock stored procedure selected, we configured its input parameter values, and we called the procedure. You can see

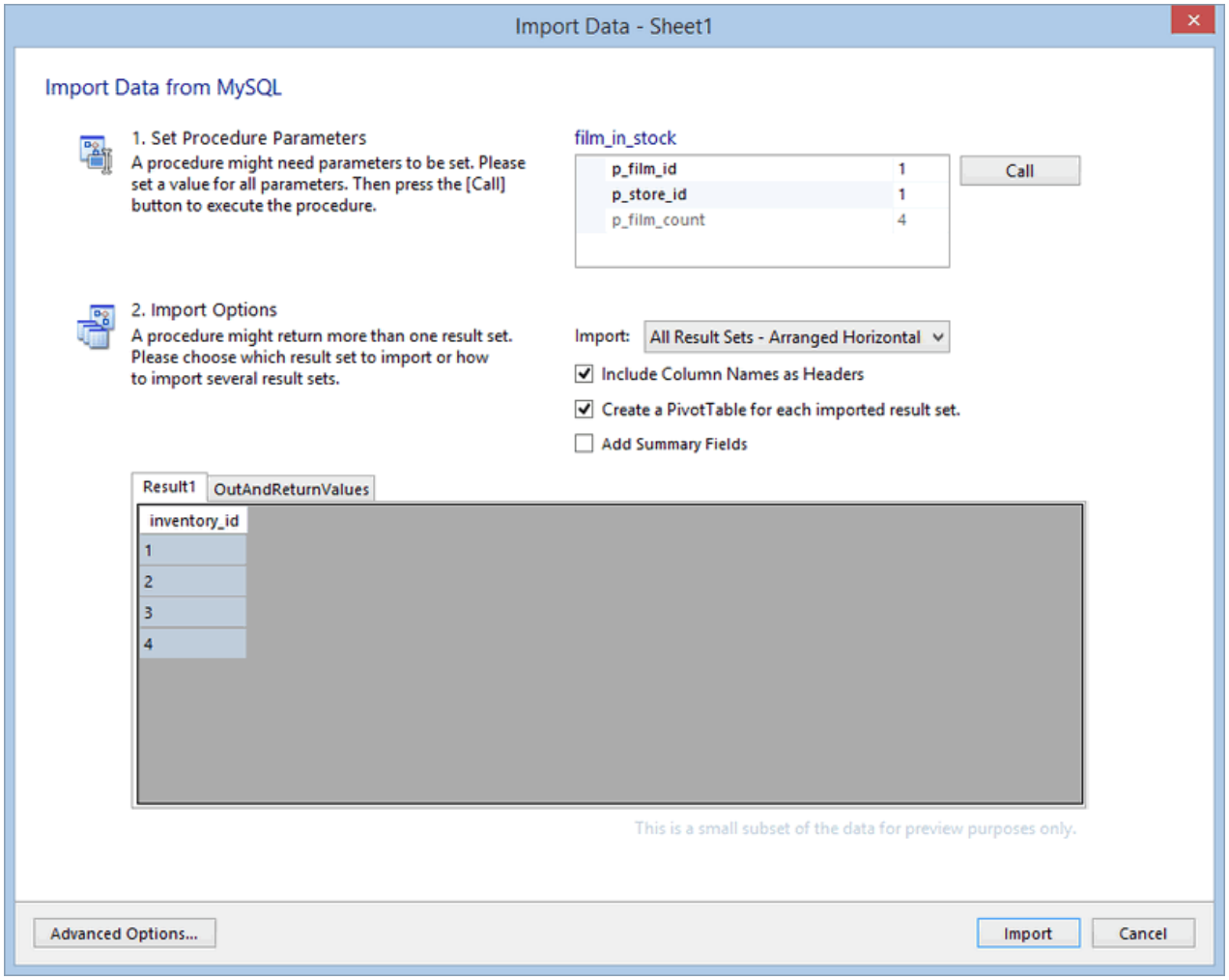
the procedure returned one result set (Result1) and the OutAndReturnValues table (always present if the procedure has output parameters or a return value).

Figure 5.14 PivotTable Example: Stored Procedure



In our example, we selected to import **All Result Sets - Arranged Horizontally**. Because the **Create a PivotTable with the imported data** option was also checked, a PivotTable was created for each returned result set.

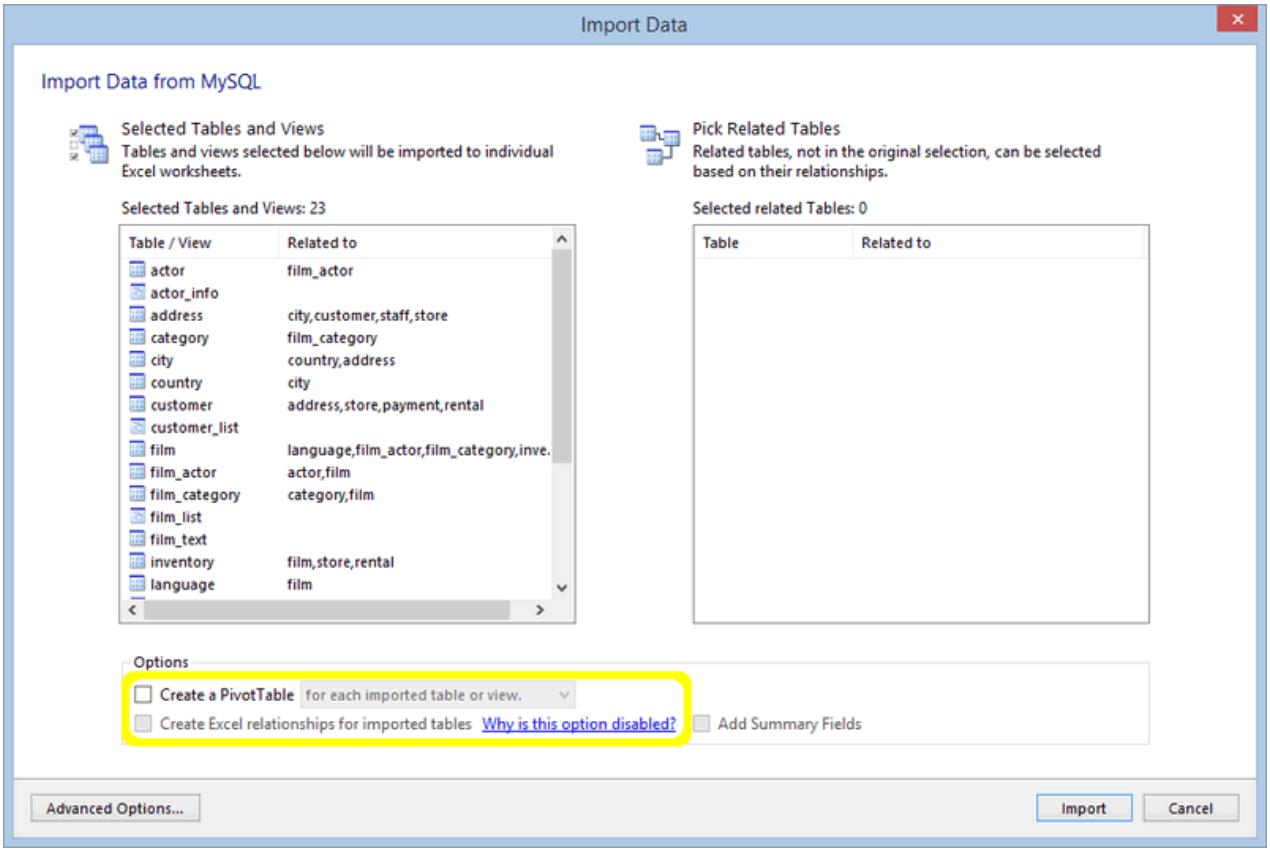
Figure 5.15 PivotTable Example: Arranged Horizontal



An important use case for PivotTables is when we create it for multiple related tables as typically a single table does not contain all of the data needed by a PivotTables report. You can create a single PivotTable tied to the data in the current Excel Data Model that contains fields from several related tables. That way you can use the data in a single report for an entire MySQL schema if needed. However, you can only do this in Excel 2013 (and later) where the Excel Data Model is available.

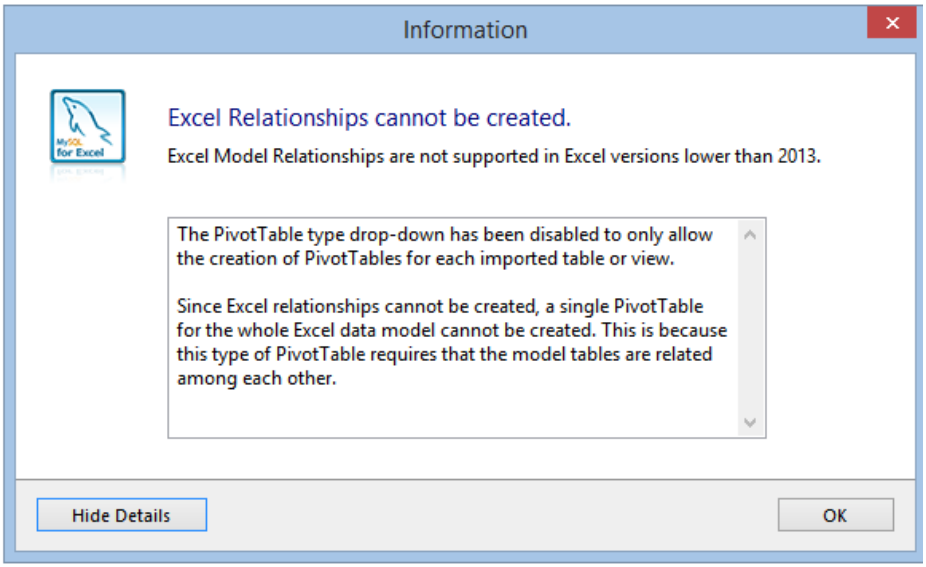
In Excel versions before Excel 2013, only a PivotTable for each imported table or view can be created. This is because a single PivotTable for the entire Excel Data Model requires that the tables are related to each other. If Excel relationships cannot be created, then this type of PivotTable cannot be created. So in these cases, the Import Data dialog looks like the following sample screenshot:

Figure 5.16 Disabled 'Create Excel relationships' option before Excel 2013



Clicking **Why is this option disabled?** displays an information dialog with an explanation of the disabled controls.

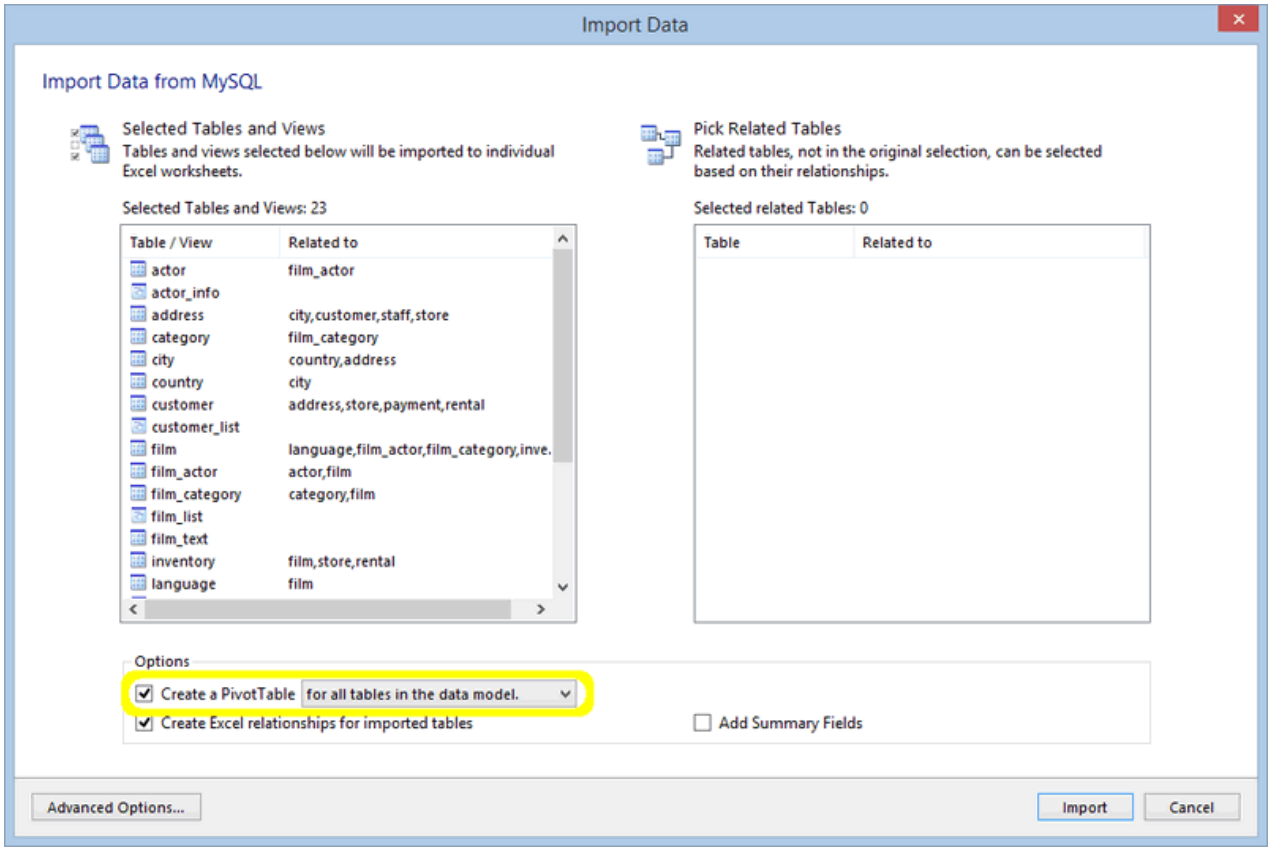
Figure 5.17 Disabled 'Create Excel relationships' option description



Our next example uses all tables in our schema. You can manually choose each table or use **Control** + **A** in the database objects list to select them all. When clicking **Import Multiple Tables and Views**, the

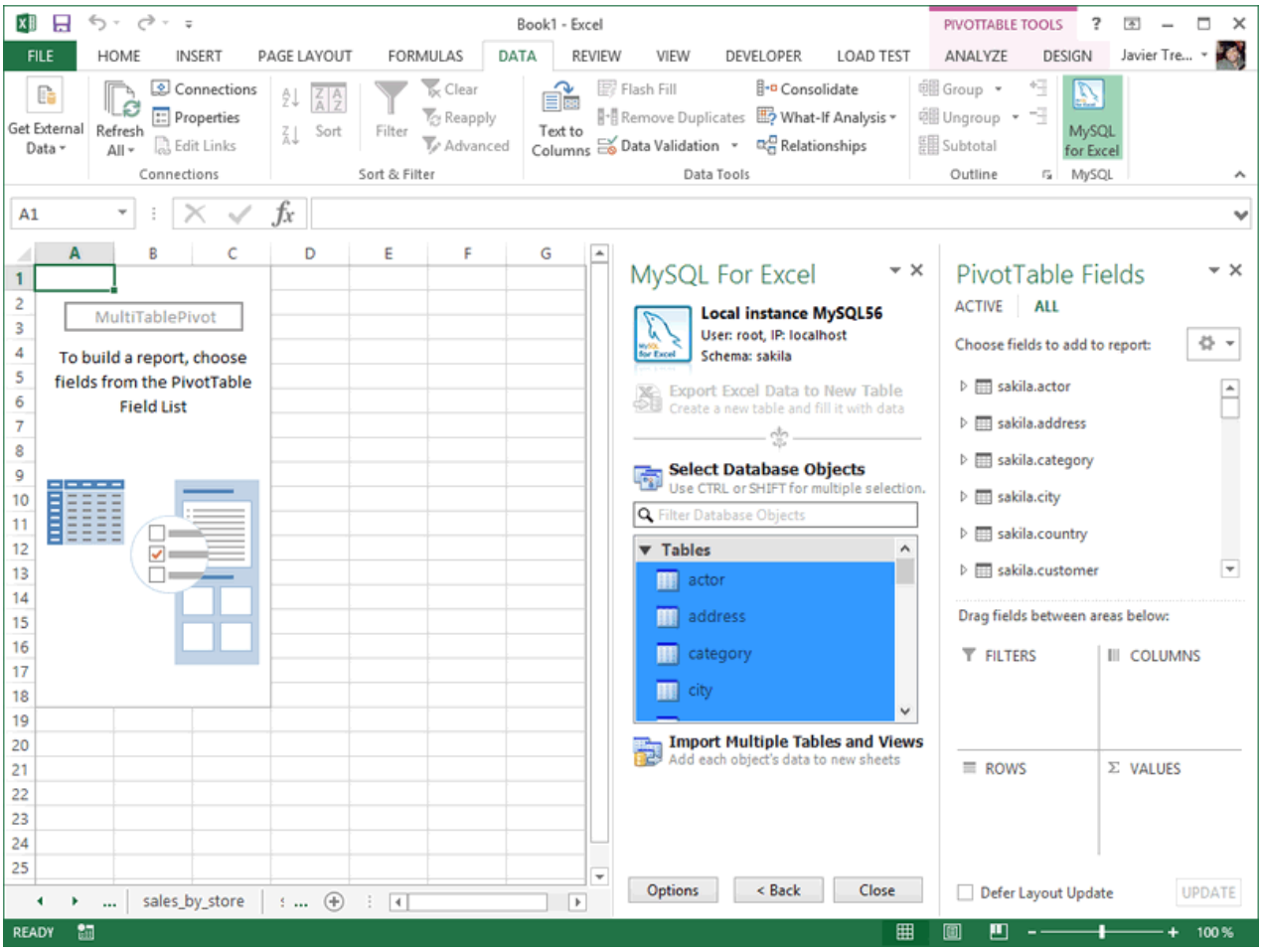
Import Data dialog appears as shown below. We need to check the **Create a PivotTable** option, which by default its drop-down is set to **for all the tables in the data model**. Keep that value.

Figure 5.18 Importing All Tables and Views



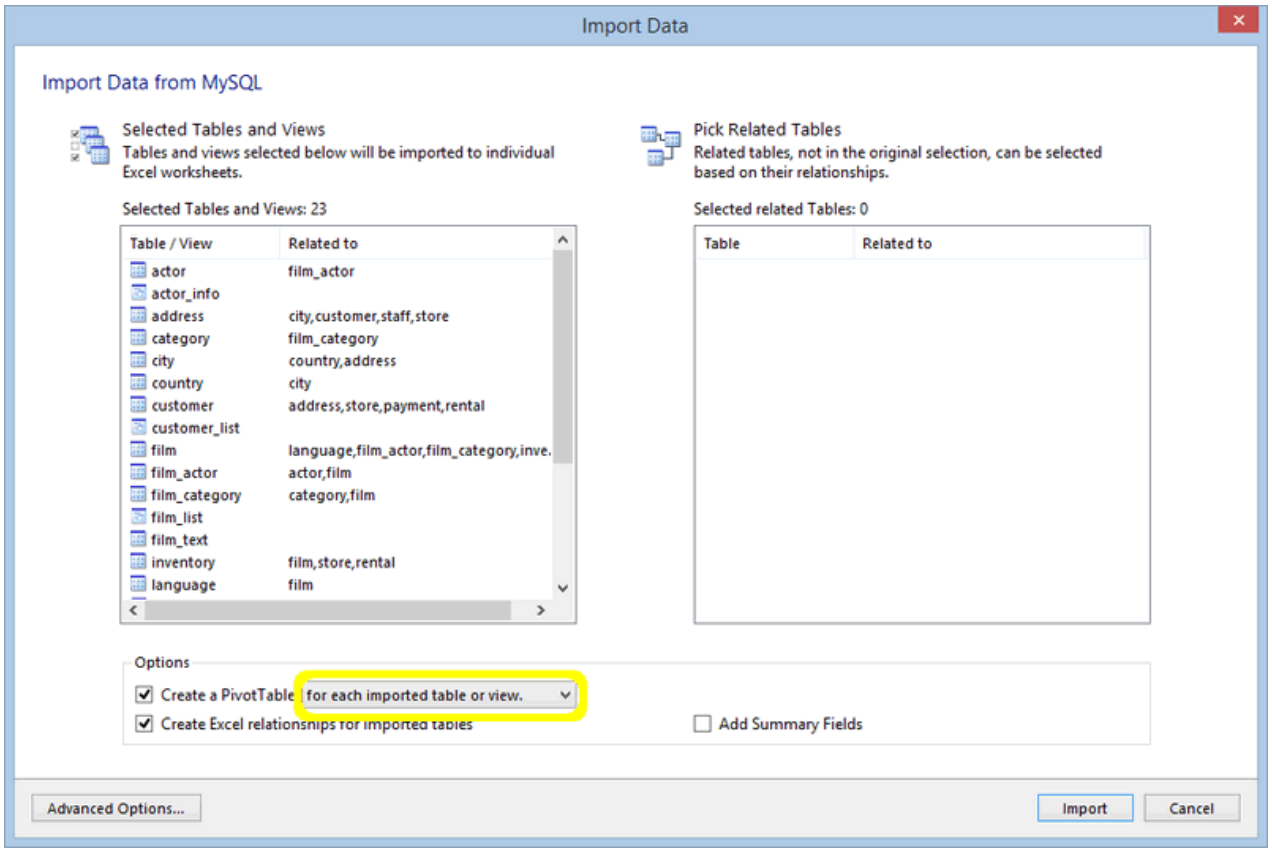
When clicking **Import**, the data in all of the selected tables is imported to Excel, its Data Model and Excel relationships are created, and a new worksheet is created that contains a PivotTable with all of the tables that were imported. This is demonstrated in the screenshot below, and notice that all tables are listed in the **PivotTable Fields** panel.

Figure 5.19 Importing All Tables and Views: Listing



You can also configure the **Create a PivotTable** drop-down list the for each imported table or view, which in turn will create a PivotTable for each of the imported tables or views, as opposed to creating a single PivotTable for all of them.

Figure 5.20 Importing Each Imported Table or View



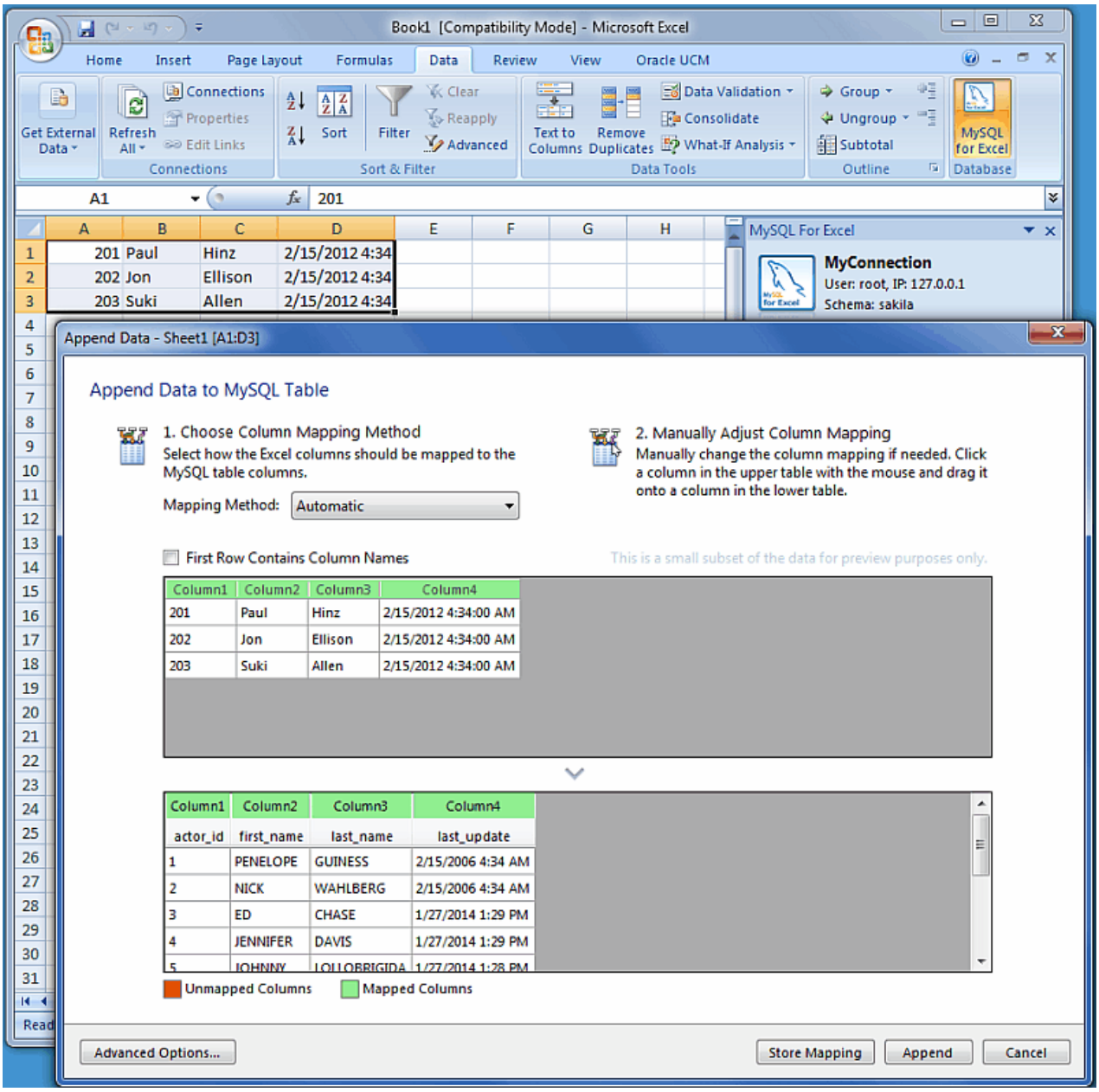
Chapter 6 Append Excel Data into MySQL

Data from a Microsoft Excel spreadsheet can be appended to a MySQL database table by using the **Append Excel MySQL Data to Table** option.

Column Mappings

Mapping the Excel columns to the MySQL columns can be executed automatically (default), manually, or by using a stored mapping routine. An automatic mapping routine is the default, and can be can be tweaked if every column cannot be matched automatically. The following screenshot shows two columns of Excel data, and the preview dialog after choosing **Append Excel Data to Table**:

Figure 6.1 Appending Excel data to MySQL (Automatic mapping)



General Mapping Information

It is common to tweak the column mappings. A few notes about the manual mapping process:

- Manual mapping is performed by dragging a column from the upper source grid (Excel spreadsheet) and dropping it into the lower target column MySQL table grid. Click anywhere within the column to initiate this dragging routine.
- The color of the header field for each column defines the current mapping status of the column. The colors include:
  - **Green**: A source column is mapped to a target column.
  - **Red**: A target column is not mapped.
  - **Gray**: A source column is not mapped.
- A source column may be mapped to multiple target columns, although this action generates a warning dialog.
- Right-clicking a target column shows a context menu with options to either **Remove Column Mapping** for a single column, or to **Clear All Mappings** for all columns. Dragging a target column outside of the grid removes the mapping.

Mapping Methods

The three mapping methods are described below:

- **Automatic**: The automatic mapping method attempts to match the Excel source column names with the MySQL target table column names. It is then possible to manually tweak the mapping afterwards.

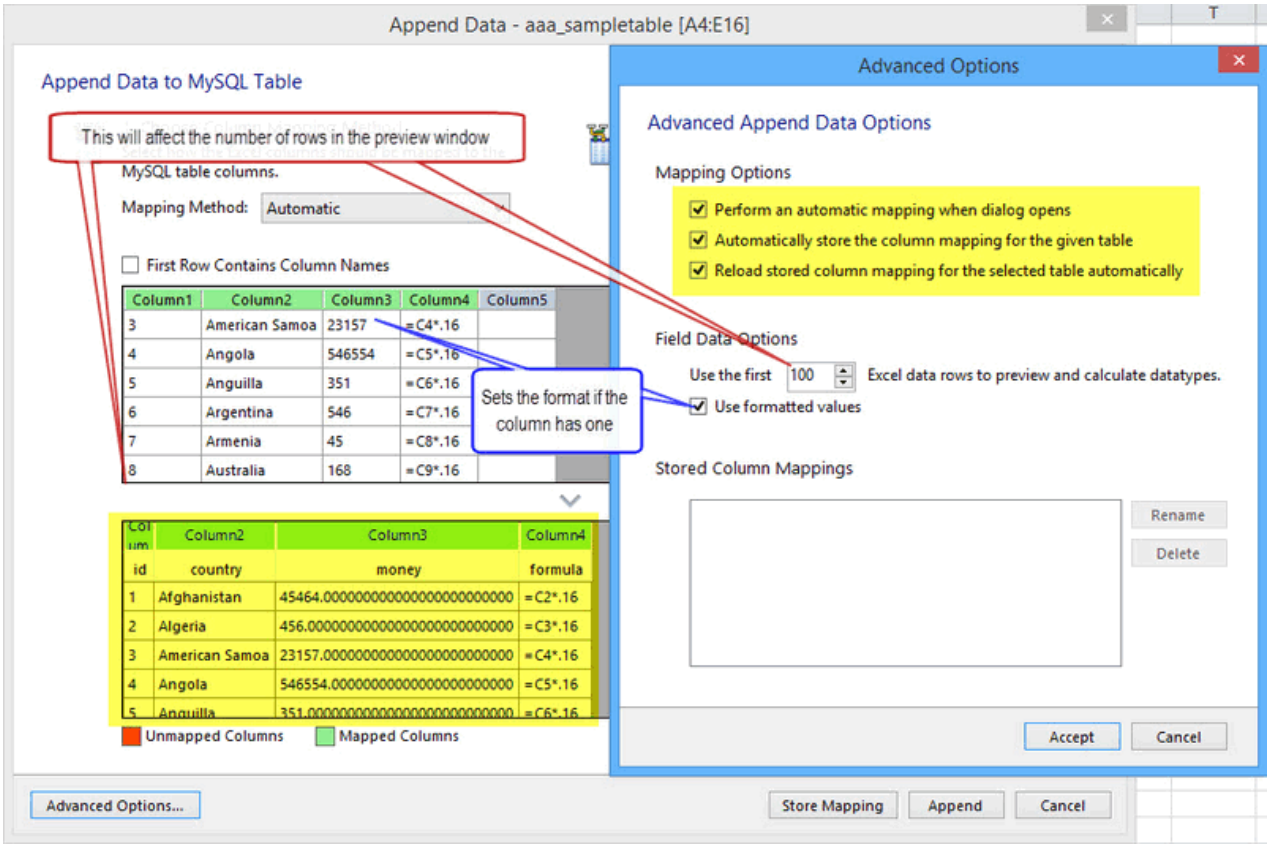
  If the automatic process finds zero columns to match, then a simple 1 to 1 matching routine is attempted. Meaning, SourceColumn #1 to TargetColumn #1, SourceColumn #2 to TargetColumn #2, and so on.
- **Manual**: The source column names are manually dragged (matched) with the target column names. Manual dragging can also be performed after the **Automatic** method is selected.
- **Stored**: Manual mapping styles may be saved using the **Store Mapping** button, which will also prompt for a name and then save it using a "*name* (dbname.tablename)" naming scheme. The saved mapping style will then be available alongside the **Automatic** and **Manual** options.

  Stored mappings may be deleted or renamed within the **Advanced Options** dialog.

Append: Advanced Options

There are several advanced options that are configured and stored between sessions for each Excel user. The advanced options are:

Figure 6.2 Appending Excel data to MySQL (Advanced Options)



The advanced **Mapping Options**:

- `Perform an automatic mapping when dialog opens`: Automatically attempt to map the target and source when the **Append Data** dialog is opened. This feature is enabled by default.
- `Automatically store the column mapping for the given table`: Stores each mapping routine after executing the **Append** operation. The mapping routine is saved using the "tablenameMapping (dbname.tablename)" format. This may also be performed manually using the **Store Mapping** button. It is enabled by default, and this feature was added in MySQL for Excel 1.1.0.
- `Reload stored column mapping for the selected table automatically`: If a stored mapping routine exists that matches all column names in the source grid with the target grid, then it is automatically be loaded. This is enabled by default, and this feature was added in MySQL for Excel 1.1.0.

The advanced **Field Data Options**:

- **Use the first** *`100`* (default) Excel data rows to preview and calculate data types. This determines the number of rows that the preview displays, and the values that affect the automatic mapping feature.
- **Use formatted values**: The data from Excel is treated as `Text`, `Double`, or `Date`. This is enabled by default. When disabled, data is never treated as a `Date` type, so for example, this means that a date would be represented as a number.

The advanced **SQL Queries Options**:

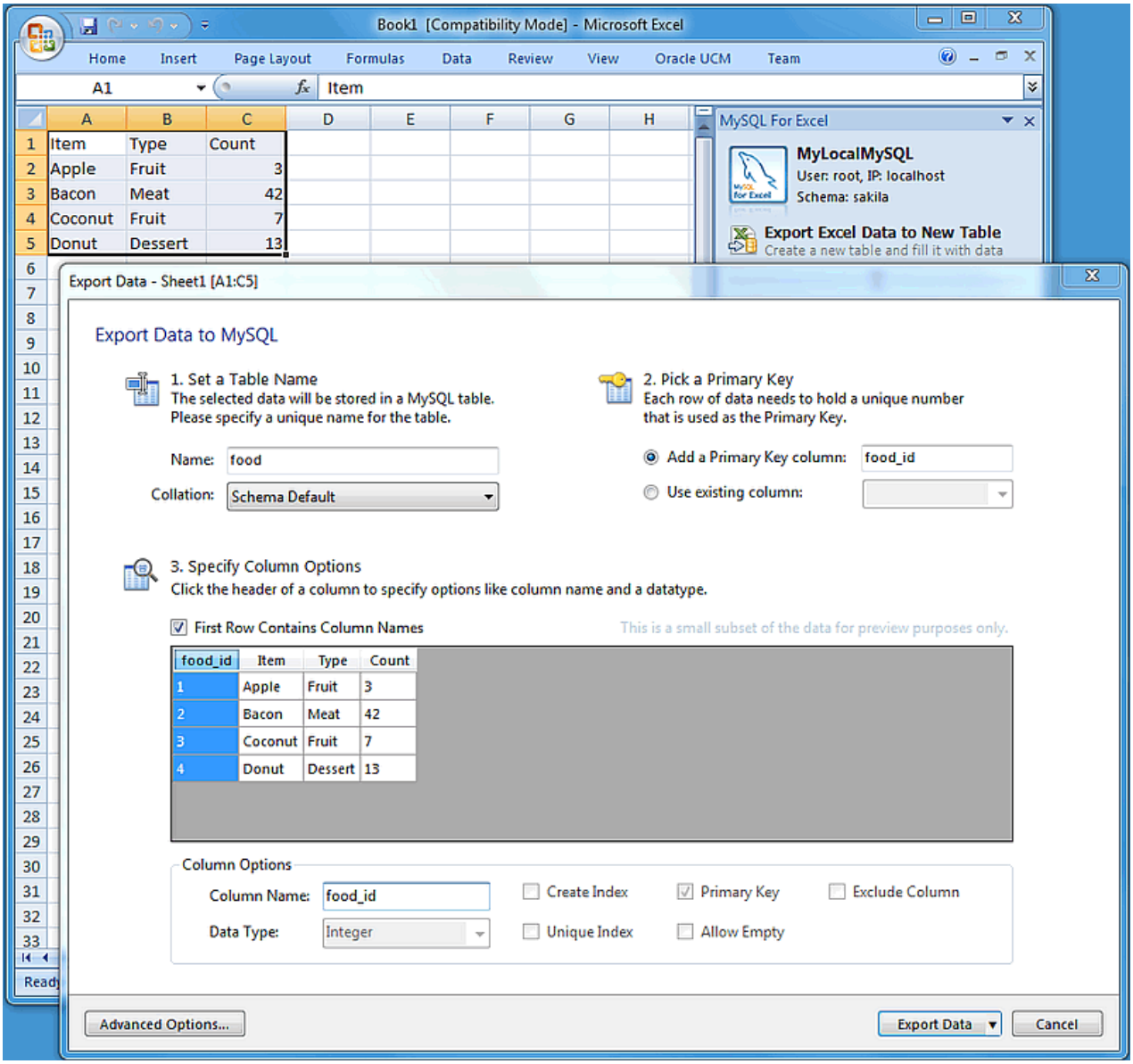
- `Disable table indexes to speed-up rows insertion`: This option is disabled by default, since you must make sure that if unique indexes are present, that the data mapped to that column does not contain duplicate data. This option was added in MySQL for Excel 1.2.1.

The **Stored Column Mappings** is a list of saved column mappings that were saved with the "Automatically store the column mapping for the given table" feature, or manually with the **Store Mapping** option.

Chapter 7 Export Excel Data into MySQL

Data from a Microsoft Excel spreadsheet can be exported to a new MySQL database table by using the **Export Excel Data to New Table** option. Exporting data looks like so:

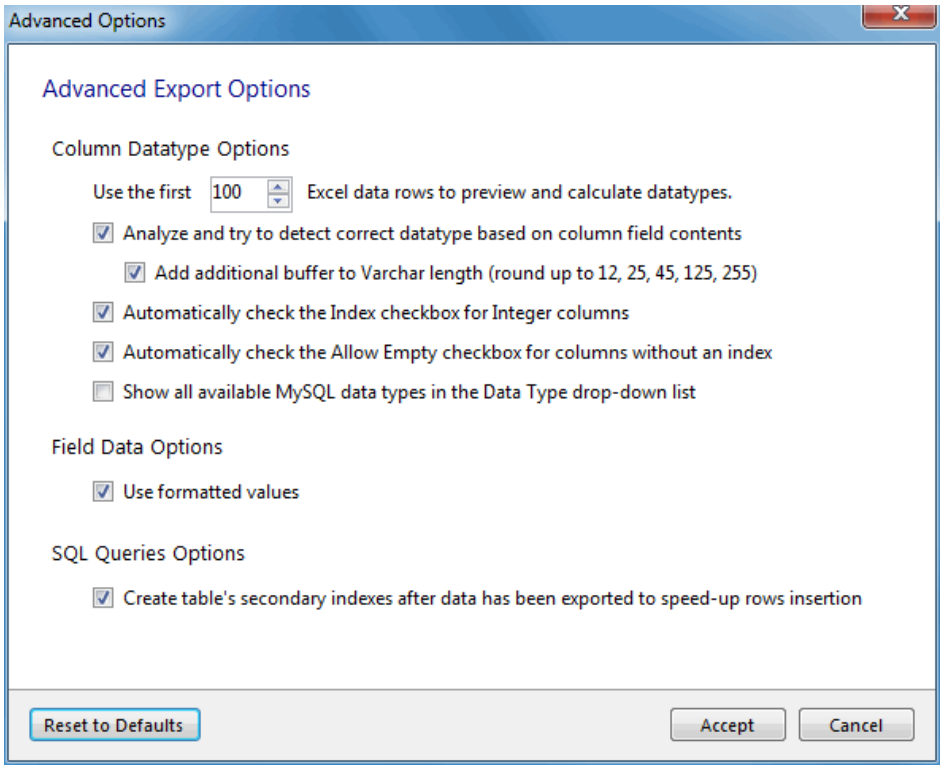
Figure 7.1 Exporting Excel data to MySQL



Advanced Export options

Several advanced options enables you to tweak the exported data. The advanced options dialog looks like so:

Figure 7.2 Exporting Excel data to MySQL (Advanced options)



- **Column Datatype Options**:
  - *Use the first `100` (default) Excel data rows to preview and calculate data types*: This determines the number of rows that the preview displays, and the values that affect the automatic mapping feature.
  - *Analyze and try to detect correct datatype based on column field contents*: Attempts to analyze the data and determine the data type for the column. The column type is defined as VARCHAR if it contains multiple types.
  - *Add additional buffer to VARCHAR length (round up to 12, 25, 45, 125, 255)*: When the data type is automatically detected and is set to VARCHAR, then it calculates the maximum length for all rows within the column, and rounds up the maximum length to one of the defined lengths above.

    If disabled, then the VARCHAR length is set to the length of the longest entry in the Excel spreadsheet.
  - *Automatically check the Index checkbox for Integer columns*: If enabled (default), columns with an Integer data type will have the **Create Index** option enabled by default.
  - *Automatically check the Allow Empty checkbox for columns without an index*: If enabled (default), columns without the **Create Index** checkbox checked will automatically enable the **Allow Empty** configuration option.
  - *Show all available MySQL data types in the Data Type drop-down list*: By default, only the most commonly used data types are displayed. Enable (disabled by default) this setting to see a list of all MySQL data types.

Note

This option was added in MySQL for Excel 1.3.0

- **Field Data options**:
  - *Use formatted values*: When enabled (default), the data from Excel is treated as `Text`, `Double`, or `Date`. When disabled, data is never treated as a `Date` type, so for example this means that a date would be represented as a number.
- **Other options**:
  - *Create table's secondary indexes after data has been exported to speed-up rows insertion*: This saves disk I/O for bulk inserts (thousands of rows) since re-indexing will not happen every time a row is inserted, but only once at the end of the data insertion. This option is enabled by default.

Note

This option was added in MySQL for Excel 1.2.1.

- 

Note

The following option was `Removed` in MySQL for Excel 1.2.1. Now, the default behavior is to always remove empty columns from the calculations.

Remove columns that contain no data, otherwise flag them as "Excluded": If enabled, columns without data in Excel are removed and not shown in the preview panel. If disabled (default), these columns will exist but have the **Exclude Column** option checked.

Additional Notes

- Entering "0" into a date column.

  Entering the value "0" into an Excel date column will convert the value to "12/30/1899" in MySQL. This is because Excel first translates the value to the minimum date in Excel, which is "1/0/1900", because dates are internally stored in Excel as numbers (the days that have passed since "1/0/1900". However, because "1/0/1900" is not a valid date, the committed value to MySQL will change to "12/30/1899" because the .NET MySQL connector automatically converts "1/0/1900" to "12/30/1899", which is the closest valid date.

Chapter 8 What Is New In MySQL for Excel

Table of Contents

This section summarizes how MySQL for Excel progressed with each minor and major release.

8.1 What Is New In MySQL for Excel 1.3

Most of the new features added to MySQL for Excel 1.3.x involve improvements to the **Data Import** functionality.

- You can now refresh imported data from the source MySQL database by clicking **Refresh** from the context-menu, or **Refresh All** from the navigation menu. These check for changes in the source MySQL database and update your imported MySQL data accordingly.

  Use case: A colleague sends you a MySQL Excel spreadsheet with data exported from a MySQL database. You open the file several days later, and worry that the data is outdated so you click **Refresh**.

- A new **Refresh To Defaults** button was added to the options pages. It changes each option to the default value, and you then confirm (or cancel) the application of these changes.

- Enabling the new **Add Summary Fields for Numeric Columns** option adds a summary field to the end of each numeric column in Excel. From here, you choose the desired function for the column, such as total or average.

- You may now import data from multiple objects in a single operation. Use **Control** or **Shift** to select multiple objects (tables and/or views) from the MySQL for Excel panel, and click **Import** to open the new dialog for selecting additional objects that have direct relationships to the objects you selected. Each object opens in its own Worksheet.

  From this new dialog, you may also generate a **Relationships** model in Excel. This functionality requires Excel 2013 or higher, or Excel 2010 with the PowerPivot add-in.

- A new **Create a PivotTable with the Imported Data** option was added. This creates a Pivot Table in Excel.

- All options now have descriptive tooltips. Hover over an option/preference to view helpful information about its use.

- You may now specify a collation for created schemas. The collation type defaults to "Server Default." These statements can be reviewed before execution.

- All MySQL data types are now available when performing **Data Export** operations. By default, only the most commonly used data types are listed, which was only behavior in previous versions of MySQL for Excel. You may still type in a type instead of selecting it.

8.2 What Is New In MySQL for Excel 1.2

- `Edit Connections`: MySQL connections can now be edited from within the MySQL for Excel plugin by right-clicking and choosing **Edit Connection**. Before, these connections could only be edited with MySQL Workbench.

- `Optimistic Updates`: Previously, only "Pessimistic Updates" were used, which means that pressing **Commit Changes** would overwrite changes performed outside of MySQL for Excel for the edited cells.

  Both options remain available today, and optimistic updates are enabled by default. This update type can be set either as a preference, or toggled per session.

- The **Append Data** dialog will now notify you of incompatible types (with visual warnings) when mapping source Excel columns to target MySQL columns.

  If a mismatch is discovered, then the column in the source grid that contains the mapped Excel data turns red, and selecting this column displays a warning with text explaining that the source data is not suitable for the mapped target column's data type.

- New preview preferences allow you to enable one of the following three options:

  - **Preview SQL statements before they are sent to the Server**: View (and optionally) edit the MySQL UPDATE/INSERT statements before they are committed.

  - **Show executed SQL statements along with the results**: View the statements after they are committed, which is the current behavior.

  - **Do not show the MySQL statements**: Only show summary information, such as number of affected rows, and not MySQL statements. This is enabled by default.

- Create Table: The **Data Export** feature now has the option to only create the table without inserting the data.

  To execute, toggle the **Export Data** button to **Create Table**, and then click.

- The selected schema name is now displayed on top of the MySQL for Excel Database Object Selection panel.

- The **Advanced Options** dialogs opened from the Import, Export and Append Data windows now immediately apply the option changes, when before the **Advanced Options** dialog had to be reopened before the changes could be previewed.

- **Edit Data** sessions can now be saved: Using the new **Edit Session** preferences, these sessions were automatically closed after closing an Excel workbook. This data, such as the Workbench connection ID, MySQL schema, and MySQL table name, can now be preserved if the Excel workbook is saved to disk, and available when the Excel workbook is reopened.

- Excel tables are automatically created for any data imported from MySQL to an Excel worksheet, with a name like "Schema.DB-Object-name". The DB object name can be a MySQL table, view, or stored procedure. Options for this feature are listed under **Import Data**, **Advanced Options**. The newly created Excel tables can be referenced for data analysis in Pivot Tables or reports.

Chapter 9 MySQL for Excel Frequently Asked Questions

Questions

Questions and Answers

9.1: When I'm using Excel to edit data from a live MySQL database, will my changes overwrite changes made by other users? For example, maybe they used MySQL Workbench to edit the same data at the same time.

The optimistic updates feature checks for external edits and notifies you of their existence before committing any changes. If an external edit is discovered, you can then choose whether or not to overwrite their changes. This option is enabled by default and can be disabled (to use pessimistic updates). Disabling this option means external changes will always be overwritten. In other words, the choice is yours.

9.2: I installed the MySQL for Excel plugin, but can't find it in Microsoft Excel. How do I start it?

The MySQL for Excel plugin is automatically added to Microsoft Excel's data menu when it is installed. Look for the MySQL for Excel icon, by default it will be listed on the right side of the main menu.

If it's not there, then you might have to reinstall the plugin. But before doing so, first check if it's listed under "Add/Remove Programs" in Microsoft Windows. If not, then it has not been installed. Next, check the Excel Add-Ins list. For Office 2007 this is found by clicking the Office logo in Excel (top left corner), click **Excel Options**, then select **Add-Ins**. Is MySQL for Excel listed as a COM Add-in? If so, then consider filing a bug report (bugs.mysql.com), or attempt to reinstall the plugin.

9.3: I click on Edit Data and after importing the table data into Excel, I can't sort or move columns around. How can I do that?

In order to maintain the mapping of rows and columns in the Excel Worksheet against the rows and columns in the MySQL table, no alteration is permitted on the worksheet (i.e. sorting, deleting rows, deleting columns). If you need to alter the data there you can do that by right-clicking the **Edit Data** window and selecting **Exit Edit Mode**.

9.4: When editing a MySQL table's data, the Excel worksheet where the data is dumped is protected. How can unprotect it?

The Excel worksheet is protected to not allow alterations to the order of rows and columns. The password used for the protection is a GUID auto-generated at runtime so that the protection is not violated in any way. If you wish to unprotect the worksheet to manipulate your data, you can do that by right-clicking the **Edit Data** window and selecting **Exit Edit Mode**.

9.5: The MySQL Workbench connections that use SSH tunneling appear grayed out (disabled) in MySQL for Excel. How can I use a SSH connection?

This is a known limitation of MySQL for Excel. MySQL for Excel uses MySQL Connector/NET to connect and communicate with MySQL databases. Connector/NET does not have SSH support, so the behavior will change if Connector/NET supports it in the future.