1. Create your own unique database table

A screenshot of a computer

Description automatically generated

2. Add a foreign key from your “other” database table to table web\_user (meaning that “other”.web\_user\_id shall reference web\_user.web\_user\_id). If you are unsure how to do this, re-read the MySQL Workbench tutorial, but recall that this is how you added a foreign key from web\_user.user\_role\_id to user\_role.user\_role\_id.

A screenshot of a computer

Description automatically generated

3. Try to add an “other” record with an invalid FK to web\_user. Screen capture the Invalid FK from “other” to web\_user and paste this into your document.

A screenshot of a computer

Description automatically generated

4. Try to delete a web\_user record that’s being referenced by an “other” record. Screen capture the Error - cannot delete a referenced web\_user record and paste this into your document.

A screenshot of a computer

Description automatically generated

5. A select statement showing all the data from your “other” table (all rows, all columns, SELECT \* is OK), sorted by the id of your “other” table.

A screenshot of a computer

Description automatically generated

6. A select statement of your “other” table joined with web\_user (all rows), showing just the descriptive field from your “other” table and then the user email address from web\_user. To have a primary then secondary sort, you just specify two columns after the ORDER BY (e.g., ORDER BY col1, col2). Note: there should be as many rows in this result set as there were in the above result set.

A screenshot of a computer

Description automatically generated

7. A select statement of your “other” table joined with web\_user (but just some rows). It would be the same select statement as above, but filtering out some of the rows by adding additional criteria to your WHERE clause.

A screenshot of a computer

Description automatically generated