

Exercises - Session 31



In case you get stuck anywhere, don't be afraid to ask the coaches! They are here to help and will gladly explain everything to you! Take notes during the exercises. Even if you never look at them again, they will help you memorise things!

The goal of this exercise is to build a simple address management tool with sinatra. We won't explain in detail how to do this in the exercise. We won't give you any source code to start with either. Try to find out how to do things by googling and looking at earlier exercises. We realise this could be quite challenging. Don't get discouraged too easily! ;-)

Data model

1. ON PAPER: In pairs, discuss how the database table of an address management tool should look like:
 - a. Which columns do you need?
 - b. What are the data types for each column?

Database setup script

We're going to start implementation by writing a database setup script. Running this script will create the SQLite database and fill it with some sample data using the sequel gem. Create a file called `db-setup.rb` and implement the following functionality:

1. Create the database using the sequel gem. You find information on how to do that in the following link: <http://sequel.jeremyevans.net/rdoc/classes/Sequel/Database.html>
2. Fill the database with at least 5 entries. If you are not sure which statement to use, google it.

Address application

Now we're going to start implementing the sinatra application. Do this in another Ruby file.

1. Implement the index action for your sinatra application. It will display a table of all the addresses in the database. When you go to <http://localhost:4567/> this is the thing you should see.
2. A desirable feature is to be able to sort the table by individual columns. To achieve this, change the table headers to links. Each link should include a query parameter for the column they belong to. In the sinatra application, use this parameter to sort the returned list of database entries by the desired column.

Hint: First, find out how the SQL query would look to sort by an individual column, then try to use the sequel gem to achieve the sorting.

3. We want our users to be able to find entries in the table quickly. To do this, we will add a search field to the application. Above the entries table, add a form for this. Change the sinatra application to take the search parameter sent by the form. Instead of returning all the addresses, if the search parameter is given, return a list of entries filtered by the search parameter.

Hint: First, find out how the SQL query would look, then try to do the same with the sequel gem. Find out how to query do the query for one column first and add more later.

Optional

1. What do you have to do to sort the table columns in descending instead of ascending order? How can you offer both options to the user?
2. Can you add another column to the table? What would the SQL for this look like and how would you call the sequel gem to do this?
3. What happens to the existing rows of the table if you do so?
4. What problems can occur with that and how can you solve them?