01UDFOV/01TXYOV Applicazioni Web I / Web Applications I [2024/2025]

Lab 2: Database and methods

This lab builds directly on the object model you created in Lab 1, applying persistent storage concepts to your specific exam track scenario. You will integrate your JavaScript application with a local database. In the first part, you'll design your database structure, then you will implement functions to retrieve data from the database, and in the last part, you will make modifications to the data stored in the database.

1. Design and Create Your Database

Starting from the objects you identified during Lab 1:

1. Together with your group, map the previously identified objects to database tables.
2. Describe the main tables and attributes on your README file.
3. Create your SQLite database[1] and upload it to your repository.

2. Retrieve Data from the Database

Modify the application you developed in the previous lab to implement the following data retrieval capabilities:

a. Retrieve all items from one of your tables and return a Promise that resolves to an array of objects. If useful, you can think of repeating this step for other tables.
b. Retrieve all items that respect a specific condition from one of your tables and return a Promise that resolves to an array of objects. If useful, you can think of repeating this step for other tables or conditions. Possible examples of conditions (that may or may not be relevant to your selected topic) are:
    - Condition on dates.
    - Objects with specific attributes.
    - Search for specific substrings in some attributes.

After implementing these methods, test and verify their functionality by calling each method and displaying the results.

3. Modify the Stored Data

Before proceeding with this exercise, make a copy of the local database file, as the following methods will permanently modify its content.

Implement the following data modification methods:

a. Store a new item in your database. After completion, display a confirmation/failure message.

[1] You can create it with DB Browser for SQLite, https://sqlitebrowser.org/dl/.

b. Delete an item from your database using its ID. After completion, display a confirmation/failure message.
c. Update a specific item or a specific property of multiple items across your database. After completion, display a confirmation/failure message.

Notes:

1. These database operations must be implemented as asynchronous methods using Promises or async/await.
2. As covered in the lectures, you can connect to an SQLite database using the **sqlite3** (https://www.npmjs.com/package/sqlite3) module
3. To browse the content of the database, you can use one of the two following tools:
    a.
    b. Visual Studio Code *SQLite Viewer extension*, https://marketplace.visualstudio.com/items?itemName=qwtel.sqlite-viewer
    c. *DB Browser for SQLite*, https://sqlitebrowser.org/dl/