**Sample Question-1**

1. Design the database schema for the Inventory Management System. Create a file named "inventory\_schema.sql" and write the SQL code. The database should have the following tables:

Table Name: Items

Columns:

● ID (Primary Key with Auto Increment)

● Item Name (Text)

● Quantity (Integer)

● Price (Decimal)

1. Create a configuration file named "config.php" to store the database connection details.
2. Develop a web page named "index.php" that displays a list of inventory items from the "Items" table. Show item names, quantities, and prices.
3. Allow users to add new items to the inventory. Provide input fields for item name, quantity, and price. Enhance the "index.php" page to allow users to update item details.
4. Integrate a simple API to improve inventory management. Create a new webpage named "api.php" that provides an API endpoint to retrieve inventory items from the "Items" table.

**Sample Question-2**

1. Create the following database table.

Database name: StudentINFO

Table name: Student

|  |  |  |  |
| --- | --- | --- | --- |
| ID | Email | Name | Status |
| 1122 | [1122@gmail.com](mailto:1122@gmail.com) | Tom | Passed |
| 1123 | [1123@gmail.com](mailto:1123@gmail.com) | Peter | Failed |
| 1123 | [1124@gmail.com](mailto:1124@gmail.com) | Anna | Passed |
| 1125 | [1125@gmail.com](mailto:1125@gmail.com) | Lucy | Failed |

1. Develop a frontend.php/frontend.html file, where you can click on a view button and it will show you information of all the students. (backend.php, using XHR)
2. Develop a frontend2.php/frontend2.html file, where you provide any id as input and click on submit button. If the student status is “passed” you can see his/her information otherwise the student information will be deleted from the database by using appropriate queries. (backend2.php, fetch api)