Week 2 Quiz 1

1. Which of the following is NOT a reason it is important to have a well-designed database schema?

   - Poorly designed database schemas can result in poor software performance

   - Poorly designed schemas may not align well with all stakeholders' requirements of a piece of software

   - Well designed schemas can eliminate the need to define the relationships between tables in your database

   - Well designed schemas can be used by LLMs or programmers to quick prototype an application

   A schema, even if well defined, will still need to define the relationships between tables in your database,

2. Which of the following best describes the relationship between SQLite and SQLAlchemy?

   - SQLite provides a lightweight database while SQLAlchemy provides cloud storage to host the database.

   - SQLite and SQLAlchemy are both databases. SQLite is used for small to medium-sized projects while SQLAlchemy is used for larger projects

   - SQLAlchemy provides a lightweight database while SQLite provides tools to query and interact with that database

   - SQLite provides a lightweight database. SQLAlchemy provides tools to query and interact with that database

   Both products are used in managing a database, with SQLite actually providing for that database and SQLAlchemy providing tools to interact with it.

3. Which steps can an LLM help with while designing the schema for your database? Select all that apply.

   - Translating high level business requirements into a preliminary schema

     LLMs are able to produce good draft database schemas when provided with a description of the requirements it needs to meet.

   - Propose attributes and keys within each table of the database

     LLMs are capable of translating the attributes and relationships of entities in your database into features of your database schema like database attributes and keys

   - Identify which keys can be used to match records across tables

     LLMs are able to generate code that manages these relationships within your data

   - Write code to implement a prototype database schema

LLMs are capable of writing code to implement a schema they are either provided or that they generate themselves.

4. Which CRUD operation is implemented by the following code?

```
1    def add_user(name, email):
2        new_user = User(name=name, email=email)
3        session.add(new_user)
4        session.commit()
```

- Create

- Update

- Delete

- Read

This code is adding a new user record to the database and therefore is a "Create" operation.

5. In the videos the initial set of CRUD operations generated by the LLM were susceptible to SQL injection attacks. Which of the following approaches could be used to address these security risks?

- Assign the role of a security expert to the LLM before initially prompting it to increase the likelihood of using more secure methods

   Assigning a role to an LLM like this is a useful tool to initially generate code that aligns with your project's needs.

- Have the LLM regenerate its response using the same prompt to improve the odds of generating more secure code

- Ask the LLM to review existing code from the standpoint of a security expert to identify potential issues not identified when the code was first generated

   Asking for expert review of code you already have, whether you wrote it or an LLM, is a good way to identify some issues and generate code closer to the final desired behavior.

- Give feedback to the LLM about potential security flaws you identify in code it initially generated
   Giving feedback to an LLM is a useful way to have it update code it initially generated with a new focus.