1. **What is the value of SQLAlchemy?**

The value of SQLAlchemy is that it allows developers to use the full power of SQL easier, regardless of the type of database you are interacting with. Since it is database agnostic, SQLAlchemy issues the same commands, regardless of whether you are using PostgreSQL, SQLite, MySQL, etc. Instead of writing SQL directly, you interact with SQLAlchemy models that issue the SQL queries automatically for you.

2. **What is a model?**

A model is a way of reading and writing to a database and is represented by SQLAlchemy. Model-View-Architecture in Flask applications assume that applications deal with objects and in turn, objects are modeled after things applications deal with. One example of a model could be a model of a person which could store a name field and an age field. The provided Post model for photo-app in Lab04 is modeled after our photo-app cards and allows us to reference all the data typically stored in a post.

3. **What is a view?**

A view in SQL is a virtual table and can be created by selecting specific rows and columns from our database. Views are similar to an access point to our database. Views contain a List resource for returning lists of resources and creating new resources as well as a Detail resource for reading, updating, and deleting individual resources. Views are useful for many reasons in databases such as restricting data access, hiding data complexity, and simplifying queries.