

Exercise 1.7

What is an Object Relational Mapper and what are the advantages of using one?

An Object Relational Mapper (ORM) is a tool that facilitates communication between an object-oriented programming language and a relational database. It maps objects to database tables and handles queries and data conversions. Advantages of using an ORM:

- Productivity: Reduces boilerplate code, allowing developers to work with objects instead of writing SQL queries.
- Portability: Abstracts away database-specific details, making it easier to switch between different database systems.
- Maintainability: Encourages cleaner code by separating database operations from business logic.
- Security: Helps mitigate common security vulnerabilities, such as SQL injection, by sanitizing input and using parameterized queries.
- Performance: While there may be slight overhead, modern ORM libraries are optimized and often include caching mechanisms to improve performance.

By this point, you've finished creating your Recipe app. How did it go? What's something in the app that you did well with? If you were to start over, what's something about your app that you would change or improve?

I thought the recipe app went well. I thought I did well with understanding the SQL queries and how they can be used within a python script. If I were to start over, I would look into adding some kind of graphics to the app, as well as expanding the access to the app beyond just the terminal. When you're forced to interact with the app through the terminal, you are very limited in how aesthetically appealing you can make the app look.

Imagine you're at a job interview. You're asked what experience you have creating an app using Python. Taking your work for this Achievement as an example, draft how you would respond to this question.

In my recent project, I developed a Recipe app using Python, which leveraged SQLAlchemy for database management. This experience allowed me to integrate Python's powerful programming capabilities with a relational database, using ORM for efficient data handling. I focused on creating a user-friendly command-line interface, ensuring the app was intuitive and accessible. The project honed my skills in Python, particularly in managing data, error handling, and creating interactive applications. It's a testament to my ability to develop functional and user-friendly software solutions.

What went well during this Achievement?

My understanding and proficiency with Python increased significantly during this achievement.

What's something you're proud of?

I am proud of how my app looks. I think it is clean and efficient (as much as it can be within the terminal), and I also think it is user-friendly.

What was the most challenging aspect of this Achievement?

I had never worked with virtual environments and the iPython shell before, so understanding those concepts and utilizing them when writing code was challenging.

Did this Achievement meet your expectations? Did it give you the confidence to start working with your new Python skills?

This achievement met my expectations. I feel confident to embark on new projects using the Python skills I learned in this achievement.

What's something you want to keep in mind to help you do your best in Achievement 2?

Stay patient. I sometimes impatient about not finding a new software engineering job yet, but that isn't the kind of thing you can rush through.