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

ORMs convert database contents and structure to classes and objects that can be modified more easily. They are advantageous because the contents can be handled more similarly to any other Python object and not require extra special syntax.

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 think I did well with learning MySQL and ORM, they were easier to pick up than I thought they would be. If I were to start over I would start by structuring my application better by applying my steps in a different order. Instead of making a each full function one by one with error handling implemented, I would rather make a simple "draft" version with simple functions and then expand on it from there. I noticed that taking it in that order made me less likely to run into indent errors that took a while to get to work properly.

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.

I created an app that lets users make a list of recipes, as well as modify or delete them. It has a feature to calculate the difficulty of each recipe based off of its cooking time and length of ingredients list, and a feature to search recipes by ingredient(s). The search feature is beneficial to users because they can search by multiple ingredients they have on hand to find a recipe they can make.