**Class Design**

In designing the pizza ordering application, I opted for a singular class structure to maintain simplicity and coherence within the project. Given the relatively straightforward nature of the application's functionalities, consolidating all components into a singular class offered a pragmatic approach, ensuring manageability and clarity throughout the development process.

**Data structure**

**Why choosing your data structure (used in the solution).**

For keeping track of the toppings I used ArrayLists. They're like lists that can change size when needed, which was handy because I didn't always know how many toppings there would be. Using ArrayLists made it easier to add or remove toppings and kept the code neat.

**Compare your chosen data structure with at least one more data structure.**

Compared to other ways of storing data like Lists and Maps, ArrayLists still came out on top for my app.

ArrayLists are part of Lists, and they can grow or shrink as needed. This was perfect for this app because I had checkboxes for toppings that could change in number. ArrayLists also made it simple to manage the toppings, like adding or taking away.

Lists have other versions like LinkedLists, but ArrayLists were faster for what I needed, especially when if user had to find or change toppings often.

Maps are different because they pair things together, like toppings and prices. But for our app, ArrayLists worked better because they let us find toppings quickly, which was important when someone was ordering pizza.

So, even though Lists and Maps have their good points, ArrayLists were the best fit for the pizza app because they were flexible, fast, and easy to use.