CS112 - Spring 2024

Lab18

Instructor: Paul Haskell

INTRODUCTION

Let's write a program that takes advantage of polymorphism to make dessert!

Polymorphism

For this in-class assignment, you will make dessert. In a file called **MyDesserts.java**, you will start by making an `abstract class Dessert`. This class must have several abstract methods:

- `String ingredients()`, which returns a `String` with all of the ingredients of a dessert
- `String name()`, which returns a dessert's name
- `String where()`, which gives the best place on earth to get this dessert

`class Dessert` should have a non-abstract `toString()` method that uses the abstract methods to return a `String` containing something like:

> <<name>> contains <<ingredients>> and the best comes from <<where>>

Now you get to be creative. Please choose 3 of your favorite desserts and build a child class for each. Populate the abstract methods in each class.

In `class MyDesserts`, put your `main()` method. The `main()` method should randomly select one of your three `Dessert` classes (a different random choice every time the program is run) and print out the result of the `toString()` method.

Reminder

Put **MyDesserts.java** in a **Lab18** directory and push to GitHub before the deadline. This assignment must be turned in by the **end of class** on Monday April 1st.

Conclusion

This program gives you some experience making derived classes and using polymorphism. Your software that prints out the dessert information can be independent of the specific dessert classes. It should only use and know about the base `class Dessert`. You should be able to come back later and add five more dessert types, and the printing part of the program should remain unchanged. That obviously seems quite useful!

Grading Rubric

MyDesserts.java is worth 15 points: 10 point for correct output (output should be randomly chosen). Software design and quality is worth 5 points.