Week: 11 Date: 11/14/2024

15-110 Recitation Week 11

Reminders

- Check 6-1 due Monday (11/17) at noon!
- Check 5 & Hw 5 revisions due Tuesday (11/18) at noon
- Exam 2 grades have been released
- Recitation feedback form

Overview

- MVC Review
- Simulations: Code Writing
- Debugging Practice
- Working with Data

Problems

MVC Review + Simulation

Match the simulation part to its definition

Model	Repeatedly displays current state of the model
View	Updates components to make changes in the simulation
Controller	Stores the core components and rules of our simulation

Consider this setup that draws a circle and a box:

```
def makeModel(data):
    # Initialize the box dimensions and position
    data["bx"] = 150
    data["by"] = 150
    data["bwidth"] = 200
    data["bheight"] = 200

    # Initialize the circle
    data["cx"] = 250
    data["cy"] = 250
    data["radius"] = 30
    data["color"] = "purple"

def makeView(data, canvas):
    # Draw the box
    canvas.create_rectangle(data["bx"], data["by"],
                            data["bx"] + data["bwidth"],
                                data["by"] + data["bheight"],
                                outline="black")

    # Draw the circle
    canvas.create_oval(data["cx"] - data["radius"],
                       data["cy"] - data["radius"],
                       data["cx"] + data["radius"],
                       data["cy"] + data["radius"],
                       fill=data["color"])
```

Write the runRules function such that a new random circle (random x and y) is drawn on the 500 x 500 canvas every 300 ms. If the circle's center is within the box, then its color should be purple. If it is not in the box then it should be blue (don't worry about the timer, that is done for you).

Debugging Large Projects

Reference the starter code. There are 2 errors spread out in the debugging section—can you find them?

Note: Remember some of the strategies that we have practiced for debugging! You can try adding print statements, talking through the logic with a classmate, and testing a variety of different inputs.

Bug 1:
Bug 2:

Working with Data

Your Global Business professor gives you some sample corporate data to analyze for homework. To complete this task, make sure to download the starter code and csv files from the 15110 website and store them in the same folder.
You will be implementing the following items:

1) First, we want to read in the corporate data from the csv file. Use the skills you learned in class to write the function **readCSVFile** that takes in a file path and saves the contents to a 2D list called data.
2) Write a function **departmentNameDict** that stores the information in an input list as a dictionary. The input list is a 2D list where each inner list contains 4 elements: department_id, department_name, location_id, and department_expenses. For each inner list, the output dictionary should contain a key-value pair that maps department_name to a two-element list [location_id, department_expenses].

3) Write a function **departmentInfo** that takes in a dictionary like the one outputted above, as well as a list of departments, and finds the mean expenses across the departments, as well as the most common location among the departments. The function should also print this information. You may want to import a package to help you with this function!