Lists

CMPT 140

Lists

A concise way of referring to lots of data

List Basics

We need to know how to:

- Create lists
- Add data to lists
- Access data in lists
- Remove data from lists

Solar System Builder

Over the next few exercises, we will construct a program that allows a user to create a simple solar system using these variables:

```
planet_x = []  # list of planet centre x-coordinates
planet_y = []  # list of planet centre y-coordinates
planet_diameter = 50  # planet diameter in pixels
```

Exercise 1

Initialize `planet_x` and `planet_y` with three planets whose (x, y) coordinates are as follows:

- $(250, 250)$
- $(100, 400)$
- $(400, 100)$

Exercise 2

Write Processing code which draws all the planets in the lists.

- the canvas should be black
- all the planets have the same diameter, given by `planet_diameter`
- the first planet should be yellow (it's the sun)
- all other planets should be blue

Exercise 3

Write Processing code that:

- On a mouse click, appends a new planet's (x, y) coordinates to the lists. The new planet's coordinates are the same as the mouse coordinates

- On pressing the 'r' key, deletes the most recently added planet's coordinates from the lists, except that we cannot delete the very last planet (it's the sun!)

Exercise 4

Write Processing code which shifts the location of all planets by 20 pixels in the appropriate direction for these keypresses:

- 'w' shifts all the planets upwards
- 'a' shifts all the planets left
- 's' shifts all the planets downwards
- 'd' shifts all the planets right

Exercise 5

Write Processing code which displays the **number of planets** in the top-left corner of the canvas. Underneath this counter, display the **index**, **x-coordinate**, and **y-coordinate** for every planet in the list.

Exercise 6

What is printed to the console by this program?

```
1  classlist = ["Grover", "Ernie", "Bert", "Oscar", "Cookie"]
2  p = False
3  if "Ernie" in classlist:
4      i = classlist.index("Ernie")
5      if i > 0:
6          if classlist[i-1] == "Bert":
7              p = True
8
9      if i < len(classlist) - 1:
10         if classlist[i+1] == "Bert":
11             p = True
12
13 print(p)
```

Exercise 7

The police are working on a case. The suspect's first name starts with a "J" and works with a "Michael Thorton".

Write a **function** `find_suspects()` which takes a list **parameter** `employees` and **returns** another list containing names of possible suspects.

- `employees` is a list of strings; each string is in `"FirstName LastName"` format
- If there is no `"Michael Thorton"` in the list, then no one is a suspect.
- Otherwise, anyone whose name starts with "J" is a suspect
- Hint: We can index strings in the same way as lists!

Exercise 7 (ctn'd)

Example Input:

```
employees = ["Michael Thorton", "Jason Bourne", "Jack
                                Bauer"]
```

Example Output:

```
["Jack Bauer","Jason Bourne"]
```