Assignment 2: If Statements and Mathematical Expressions

For this assignment, our objective is to learn how to write and use If statements and mathematical expressions in python. We will be building off our Assignment 1 code for this assignment.

Deliverables: a PDF document with your figures and your answers to the questions at the end of the assignment

IF, ELIF, and ELSE statements:

IF: This week we are focusing on If statements. If statements work by checking a given condition, and if that condition is true, the code below will be executed. If the condition is false the code will continue running and pass the code within the if statement. For example:

```
if x == 2:
    print("hi")

print("bye")
```

For this example, let's first say we set x to equal 1. Since x does not equal 2, the code will not print "hi" and will only print "bye." But if we set x to equal 2, the code will print "hi" followed by "bye."

IF/ELSE: What if you only want your code to print "bye" if x is not equal to 2? Use an if/else statement.

```
if x == 2:
    print("hi")
else:
     print("bye")
```

For this example, let's first say we set x to equal 1. Since x does not equal 2, the code will not print "hi" and move on to the else statement and print "bye." But if we set x to equal 2, the code will print "hi" and *not* move into the else statement, and will *not* print "bye."

What if you have multiple conditions? There are a few ways to do this.

(1) **MULTIPLE IFs:** If you want your code to check for all of your conditions, just write multiple if statements. For example:

```
if x == 2:
    print("hi")
```

```
if x%2 == 0:
     print("cat")

print("bye")
```

The operation A%B means to divide A by B and return the remainder. So x%2 will equal 0 if x is an even number.

- First, let's say again we set x equal to 1. The code will then pass by both if statements and only print "bye."
- Next, let's say we set x equal to 4. This code will not print "hi" since x is not equal to 2, but will print "cat" since 4 is an even number. The code will then print "bye" after the if statements.
- Finally, let's say we set x equal to 2. This code will print both "hi" since x is equal to 2, then "cat" since x%2 is 0, then "bye" after both if statements. You can see here *both* conditions are checked regardless of whether the prior if statement was true or not.

In this case, if you want else statements, you will need an else statement for every if statement. If statements can stand on their own, but else statements *must* follow an if statement.

(2) **IF/ELIF/ELSE:** If you want your code to check conditions in order, until one is found to be true you will use a if/elif/else statement.

```
if x == 2:
    print("hi")

elif x%2 == 0:
    print("cat")

else:
    print("bye")
```

- First, let's say again we set x equal to 1. The code will then pass by both if statements and move into the else statement and only print "bye."
- Next, let's say we set x equal to 4. This code will not print "hi" since x is not equal to 2, but will move on to the elif statement and print "cat" since 4 is an even number. The code will then *not* move into the else statement, since the prior elif statement was true. The code will *not* print "bye."
- Finally, let's say we set x equal to 2. This code will print "hi" since x is equal to 2, but will *not* move on to the following elif statements and will *not* print "cat" even though x%2 is 0. The code will additionally *not* print "bye."

Note: You cannot use if/elif without an else statement!

Now, let's move onto the assignment.

STARTING:

1. Open your assignment 1 code from last week
    a. You will be using the same base and the loading in the same CSV files for data

2. Make sure that your code from assignment 1 is still functional
    a. Run it as is, with all the points plotting as one color

EDITING THE CODE:

In this assignment, and all future assignments, the places where you have to edit the code will be denoted in the comments as follows:

```
##### EDIT HERE ######
```

In this assignment, and all future assignments, the places where you have to STOP editing the code will be denoted in the comments as follows:

```
##### STOP EDITING HERE #####
```

This way, you know which parts you can change without losing functionality in the code or adding additional errors.

> *Reminder: SPACING MATTERS IN PYTHON. You will get errors if the code is not formatted correctly.*

WRITING IF STATEMENTS:

The goal of this assignment is to write some if statements to change how markers look based on different conditions. This way, not all the markers will be the same color and you will have more flexibility in the output of your code

1. First, navigate to the set_color_assignment_2() function. We want to use if (or if/else or if/elif/else) statements to have different points plot in different colors based on some conditions. Pick some of the marker variables we changed last week in the set_color_assignment_1() function and change how the markers look based on different conditions.
    a. Options for variables to change in your if statements:
        i. marker_fill_color

ii. marker_edge_color
iii. marker_size
iv. marker_type
v. marker_fill_color2
vi. marker_fill_style

2. Next, decide what you want your conditions to be for changing the color. Here are some examples:
    a. Change the color based on the distance of the point from the center of the plot
    b. Change the color based on the angle of the point around a circle
    c. Change the color based on the distance of the current point from the first point plotted
    d. For this assignment, we are looking for something a little more complex than just changing the point based on the index number or the x or y value. We would like you to get practice writing mathematical equations in python, so your conditions should be based on some mathematical expression (for ex, the distance formula).

3. Choose multiple conditions based on the output of your mathematical expression and write some combination of if, if/else, if/elif/else statements that changes the color of the point.

4. Navigate to the main function and play around with your background color using the set_facecolor function

5. Remember in the main code to change set_colors_assignment_1 to set_colors_assignment_2

6. Play around with the conditions of the If statements and background colors!

7. Turn in your two favorite combinations as pngs
    a. Don't forget to change the file name of your pngs when you save so they don't overwrite

INCLUDE IN YOUR PDF:

1. Imagine you are at the farmers market and need to choose a fruit. However, you only have enough room in your bag for one type of fruit. You like pears best, followed by oranges, followed by apples. You prefer to have pears, but will settle for oranges, followed by apples if need be.
    a. Will this problem require multiple if statements, or an if/elif/else statement?
    b. Write out these statements

2. Now, imagine you are at the farmers market and need to choose a fruit and this time you have room in your bag for as many fruits as are available. You walk around to look for pears, oranges, and apples.
    a. Will this problem require multiple if statements, or an if/elif/else statement?
    b. Write out these statements

3. Include two pngs of your final plot.
    a. PNG 1: Save a png of the plot that is created without changing the color variable.
    b. PNG 2: Save a png of the plot in a new color of your choosing by changing the color variable.

4. If you were using this base code for a larger art project, how could you incorporate if statements? Especially consider how you may use if statements for an *interactive* art exhibit.