Week 1 Quiz

Name: _____

Directions

- **This quiz contains 10 questions and will last 30 minutes.**
- For questions with *square checkboxes*, you may select *multiple* choices.
  - ☐ You could select this choice.
  - ☐ You could select this one too!

Staff use only.

Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Total

1. Write the type of the following values.

```
"Jamaica!!!!!!!!!!!!!!!!!!!"      # Type: _____________________
52                                 # Type: _____________________
52.0                               # Type: _____________________
["Coder"]                          # Type: _____________________
True                               # Type: _____________________
[False, "a", "tomato"]             # Type: _____________________
```

2. What do the following expressions evaluate to?

```
53 > 40                            # Result: _____________________
((5 % 2) * (1 + 7)) / 2.0          # Result: _____________________
str(3) == "3"                      # Result: _____________________
(1 + 2) * (1 + 7 // 2)             # Result: _____________________
int("123") != 123                  # Result: _____________________
```

3. What is the proper way of commenting in Python? Tick all that apply.

- ☐ // This is a comment.
- ☐ # This is a comment.
- ☐ % This is a comment.

4. Tick all the valid variable declarations.

```
int = 5                            # A ☐
5 = var                            # B ☐
addis-coder = "fun"                # C ☐
addis_coder = 1                    # D ☐
uwi_campus1 = "pretty"             # E ☐
```

5. Consider the following code and fill in the blanks next to each print statement.

```
w = "Harer"
x = [12, 15, 16]

print(type(w))        # What is printed: _____________________

print(type(x))        # What is printed: _____________________

print(w[1:3])         # What is printed: _____________________

print(x[1:2])         # What is printed: _____________________
```

6. What is the printed value? Read the code carefully! Scratch work may be written in the column on the left. In the column on the right, include **only** the printed values.

a)
```
i = 1
i + 1
print(i)
```

b)
```
my_var = 10
if 0 <= my_var and my_var <= 3:
    print("Jam")
elif 4 <= my_var and my_var <= 10:
    print("Coder")
```

c)
```
my_var = 10
if my_var > 0:
    print("Jam")
if my_var > 4:
    print("Coder")
```

7. Trace through the following blocks of code. After each block, write the final values of x, y, and z.

```
x = 12
y = x + 10
x = x / 3
z = y - x

# Final x value: ___________

# Final y value: ___________

# Final z value: ___________
```

```
x = [1,2,3]
x[0] = 10
y = x
y[1] = 20
z = "Hello"

# Final x value: ___________

# Final y value: ___________

# Final z value: ___________
```

```
x = "MontegoBay"
y = x[0:3]
z = x[2:6]

# Final x value: ___________

# Final y value: ___________

# Final z value: ___________
```

8. What is the printed output of the following code snippets? If there is an infinite loop, write "Infinite". If there's an error, write "Error" and also the reason for the error.

a)

```
num = 0
while num < 6:
    if num % 2 == 0:
        print(num)
    num += 1
```

b)

```
num = 0
if num % 2 == 0:
    while num < 6:
        print(num)
    num += 1
```

c)

```
counter = 0
def increment_counter(n):
  n = n + 1

increment_counter(counter)
print(counter)
```

9. Writing code! Let's write a function called `blastoff`, that takes a value `n`, and prints out the integers from `n` to `0`, then prints out "Blastoff"

For example,

```
blastoff(3)
```

will print:

```
3
2
1
0
Blastoff
```

Write your code below. You may not need to use all of the lines.

10. ***Optional*** **Challenge Problem:** Finish all other problems before doing this one.

We're going to write a `get_jamclass` function to create a list of names of all the students in our class.More specifically,

1. Start with an empty list.
2. Prompt the user to provide an integer input that will be used as the number of students in the class.
3. Repeatedly prompt the user for the names of the students and then append those to the list created in step 1. *Hint:* You can use a loop for doing this as many number of times as the class size.
4. In the end, return the entire list.

For example, if after calling `get_jamclass()`, the user inputs the following:

```
4
Yara
Tim
Jonathan
Samantha
```

The function will return:

```
['Yara', 'Tim', 'Jonathan', 'Samantha']
```

Write your code below. You may not need to use all of the lines.