

1. Functions that give a value are called _____ functions, and utilize a _____ statement, whereas functions that just display characters in the interactions pane are called _____ functions, and utilize a _____ statement.

2. Using the built-in Python method “round,” write and invoke a function that takes an integer n as a parameter and returns pi rounded to the n th decimal place.

Tip: No need to know what pi actually is! “math.pi” is also a useful built-in Python method.

3. What is the difference between an argument and a parameter?

4. What is an example of a function that would not require a parameter?

5. Write a function that takes an integer n and a string as a parameter and prints the string n times with n spaces in front of it. Ex: `func("hello world," 3)`

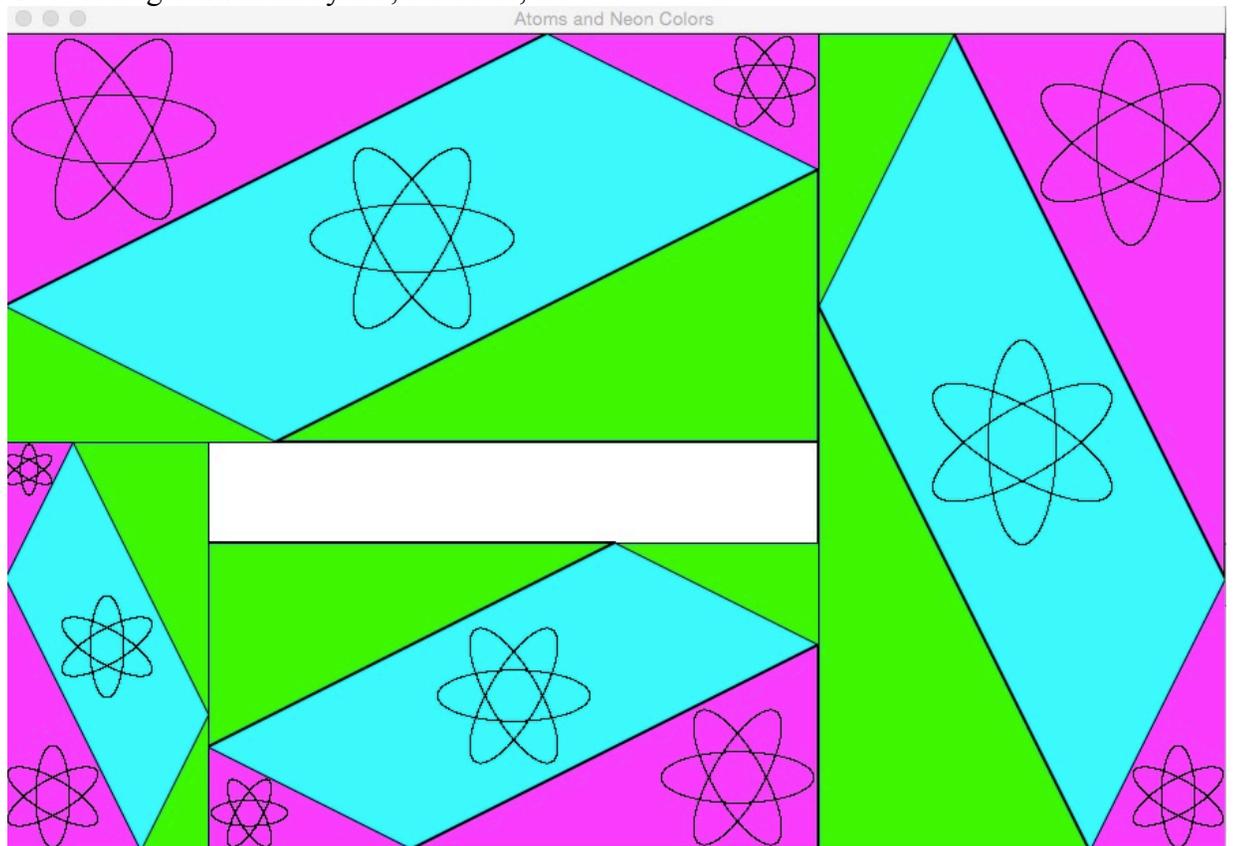
```
___ hello world  
___ hello world  
___ hello world
```

6. Write a function that takes an integer or float as a parameter and returns the number of characters in the integer or float. Then write a function that takes multiple numbers and returns the total number of characters. This second function should be one line.

For this problem, you will work in a group to write code for the image on the back.

Notes:

- The small uranium symbol is scaled by .5
- The uranium symbol is made up of three ellipses
- The rectangle in the middle is formed by the wedges covering a cyan layer
- The small wedges are scaled by .5
- One rectangle is scaled by .75; the other, .5



More Practice with Functions

1. What is wrong with the following code?

```
def converter(dollars):
    euros = dollars * 1.15
    return euros
    print (dollars + " US dollars converts to " + euros + " euros.")
```

2. Suppose you have the two following functions:

```
def multiplyByTen(num):
    return num*10
```

```
def multiplyByTwo(num):
    return num*2
```

What would the following invocation return?
`multiplyByTen(multiplyByTwo(2))`

3. What is wrong with the following code?

```
def printTwice(someString):
    print someString
    print someString
```

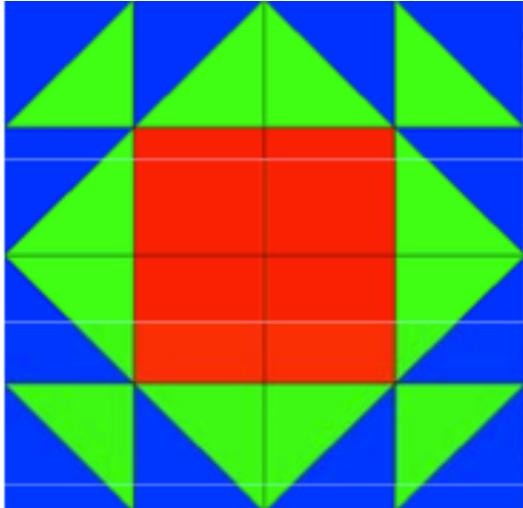
```
def printX(x):
    print someString*x
```

4. Write a function that takes a string as an argument and prints that string plus a space four times. Then write a function that takes a string as an argument and prints a space plus that string 4 times. Use the functions to produce the following:

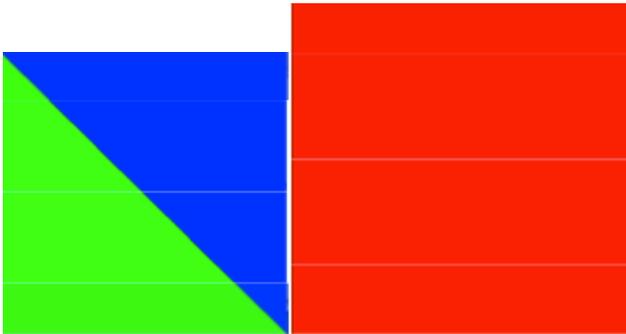
```
~~~~
 ~~~~
 ~~~~
 ~~~~
```

Divide, Conquer, and Glue

Write code to produce the following:



This looks like a complicated image to make, but it only consists of two things:



Booleans

What are Boolean values?

Write the symbol used in Python to symbolize the following:

Less than

Greater than

Less than or equal to

Greater than or equal to

Equals

Does not equal

Practice with Boolean Functions

1. Define a function that takes three lengths (a, b, and c) and determines if they could make a triangle. Hint: any given side of a triangle must be greater than the difference of the other two sides and less than the sum of the other two sides.

2. Define a function that takes three lengths and determines if one of the lengths could be the hypotenuse in a triangle consisting of the three lengths.

3. Write a function that takes an integer and returns a string of the letter grade that integer would convert to. For example, if this function were named “grade,” grade(90) should return ‘A-‘.

4. Write a function that takes a string and determines if the string is short (less than ten characters), medium (ten to twenty characters) or long.

5. Write a function that takes three strings and returns the longest string.

*6. Write a function that takes three arguments and randomly selects one to return.

*7. Say you have donuts, M&Ms, and pies, and you want to write a function that takes two of those and returns True if the first argument is bigger than the second. Ex:
bigger(“pies”, “donuts”) = True

More Boolean Practice

5. Write a function that takes a string as an input and returns whether its length is even, odd, or zero.

6. Write a function that takes a Wellesley dorm as an input and returns whether it is East or West Side. If you think there are dorms at Wellesley that don't fall into either category, write your function to include a third output.

7. Write a function that takes a number as an input and returns the number's absolute value.

Miscellaneous Practice

1. Write a function that returns a new string, identical to `s`, except that the character at the specified index is replaced with the specified new character.

For examples:

```
replaceCharacterAtIndexInString('READ', 0, 'L') return 'LEAD'
```

```
replaceCharacterAtIndexInString('READ', 2, 'E') return 'REED'
```

```
replaceCharacterAtIndexInString('READ', 3, 'P') return 'REAP'
```

2. Write a function that capitalizes the letters in the first half of the string if it is of even length.

3. Write a function that takes two numbers and returns true if the second number is a multiple of the first.

4. Write a function that takes a list of integers and returns a list containing only the integers divisible by ten.

5. Given a string, write a function that returns a list containing all incrementing substrings.

Example: `reduceString('computer')` should return `['c', 'co', 'com', 'comp', 'compu', 'comput', 'compute', 'computer']`

*6. If you have a list of two colors and you want to alternately print out the names of the colors `x` times, how would you do that?

Ex: `alt('cyan', 'yellow', 3)` \rightarrow `'cyan', 'yellow', 'cyan'`

*7. If you want to make your `cs1graphic`'s canvas title be the invocation of a function, how would you do that?

Miscellaneous Practice

1. Write a function that returns a new string, identical to *s*, except that the character at the specified index is replaced with the specified new character.

For examples:

```
replaceCharacterAtIndexInString('READ', 0, 'L') return 'LEAD'  
replaceCharacterAtIndexInString('READ', 2, 'E') return 'REED'  
replaceCharacterAtIndexInString('READ', 3, 'P') return 'REAP'
```

```
return s[0:index] + newChar + s[index+1:]
```

or

```
myList = list(s)  
myList[index] = newChar  
newString = ''.join(myList)
```

2. Write a function that capitalizes the letters in the first half of the string if it is of even length.

```
def capitalizeIfEven(string):  
    newString = ""  
    if len(string)%2 == 0:  
        for char in range(len(string)/2):  
            newString += string[char].upper()  
        return newString  
    else:  
        return string
```

3. Write a function that takes two numbers and returns true if the second number is a multiple of the first.

```
def multipleOf(int1, int2):  
    return int2%int1 == 0
```

4. Write a function that takes a list of integers and returns a list containing only the integers divisible by ten.

```
def divisibleByTen(listOfInts):  
    newList = []  
    for i in range(len(listOfInts)):  
        if listOfInts[i]%10 == 0:  
            newList.append(listOfInts[i])  
    return newList
```

5. Given a string, write a function that returns a list containing all incrementing substrings.

Example: reduceString('computer') should return ['', 'c', 'co', 'com', 'comp', 'compu', 'comput', 'compute', 'computer']

```
def reduceString(string1):  
    newList = []  
    for i in range(len(string1)):  
        newList.append(string1[:i])  
    return newList
```

*6. If you have a list of two colors and you want to alternately print out the names of the colors x times, how would you do that?

Ex: alt(['cyan', 'yellow'], 3) → 'cyan', 'yellow', 'cyan'

```
for i in range(x):  
    print (colorList[i%len(colorList)])
```

*7. If you want to make your cs1graphic's canvas title be the invocation of a function, how would you do that?

Say the function is someFunction(size,color)
myCanvas.setTitle('someFunction('+str(size) + str(color) + ')')

While Loops

Warm-up Questions:

1. In your own words, what is the difference between a for loop and a while loop?

2. What would cause a while loop to terminate? (Hint: there are three answers)

3. What would happen if I ran the following code?

```
b = True
while b:
    print(22)
```

4. What would happen if I ran the following code?

```
i = 5
while i>0:
    return i
```

5. What would happen if I ran the following code?

```
name = raw_input("What is your name?")
while name != " ":
    print("I'm sorry, I didn't catch that")
    name = raw_input("What is your name?")
```

Practice Questions:

1. Write a `makingChange` function that takes the total number of cents, the name of the denomination you're giving back, and the integer of the denomination as arguments and uses a while loop to print out how many times that denomination goes into the cents.

Example:

`makingChange(2032, "twenty", 2000)` would print out "We can make change for \$20.32 with 1 twenty."

2. Now let's say you want to make a list consisting of numbers 0 to 5, but instead of using `range` or a `for` loop, use a while loop.

3. "You have a cup of tea that is too hot. It is 115 degrees, and you want it to be 112. You add a chip of ice to it (which takes the temp down by 1 degree) until it is not too hot." Take this English expression and rewrite it in Python.

While Loops

Warm-up Questions:

1. In your own words, what is the difference between a for loop and a while loop?

A for loop iterates through a set list that the user tells it to, whereas a while loop iterates until a condition that the user sets is no longer met.

2. What would cause a while loop to terminate? (Hint: there are three answers)

Return statement, break, or condition no longer met

3. What would happen if I ran the following code?

```
b = True
while b:
    print(22)
```

22 would be printed infinitely

4. What would happen if I ran the following code?

```
i = 5
while i>0:
    return i
```

5 would be returned and the loop would terminate

5. What would happen if I ran the following code?

```
name = raw_input("What is your name?")
while name != " ":
    print("I'm sorry, I didn't catch that")
    name = raw_input("What is your name?")
```

The user would be asked for his/her name until he/she just types a space.

Practice Questions:

1. Remember the Making Change task from your first problem set? Oh, how times have changed! Write a new Making Change function that takes the total number of cents, the name of the denomination you're giving back, and the integer of the denomination as arguments and uses a while loop to print out how many times that denomination goes into the cents.

Example:

`makingChange(2032, "twenty", 2000)` would print out "We can make change for \$20.32 with 1 twenty."

```
def makingChange(total, denominationName, denomination):
    numOccurrences = 0
    while total >= denomination:
        total = total - denomination
        numOccurrences += 1
    print("We can make change for $" + str(total/100.00) + " with " +
        str(numOccurrences) + denominationName)
```

2. Now let's say you want to make a list consisting of numbers 0 to 5, but instead of using range or a for loop, use a while loop.

```
i = 0
result = []
while i < 6:
    result.append(i)
    i += 1
return result
```

3. "You have a cup of tea that is too hot. It is 115 degrees, and you want it to be 112. You add a chip of ice to it until it is not too hot."

Take this English expression and rewrite it in Python.

```
temperature = 115
while temperature > 112:
    temperature = temperature - 1
```