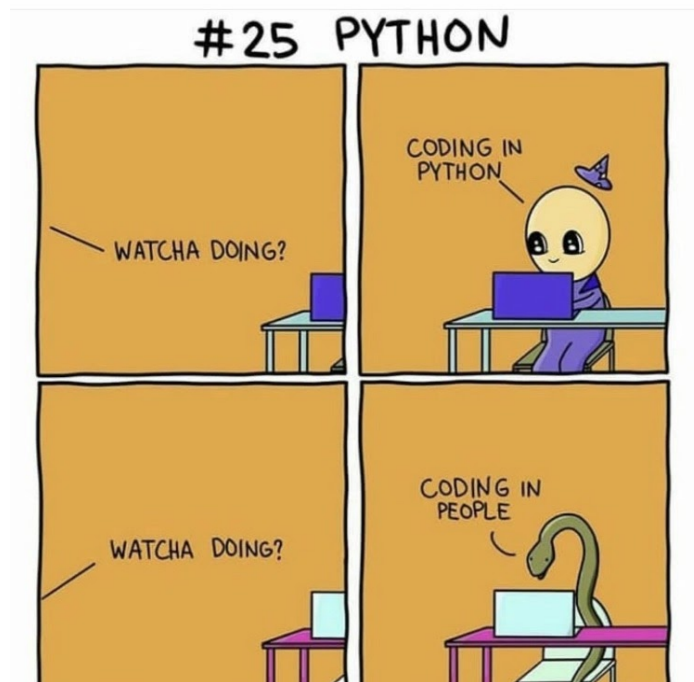**Key Concepts:**

Warning: this list is not comprehensive!

- **Parameters:** mutability, keywords, default values

- **Returning more than 1 argument**

- **String and List Slicing:** [int], [:int], [int:], [int:int], negatives

- **Some String Methods:** replace(), find(), split(), join()

- **Some List Methods:** append(), insert(), pop(), remove(), index()

- **A note on using google, documentation, and zyBooks**

Discussion Questions:

1. What will the following code output?

```
lyst = ["the", "tribe", "has", "spoken"]
element = "Brookey"
def push(lyst, element = "Yul"):
    lyst.append(element)
    print(lyst)
print(push(lyst, element))
print(push(lyst))
print(lyst)
```

2. What will the following code output?

```
lyst = list(range(0, 10, 2))
print(lyst[-2])
print(lyst[1:-1])
print(lyst[10])
print(lyst[10:])
lyst.insert(6, 4)
lyst.pop(2)
lyst.remove(2)
lyst.append(2)
print(lyst.count(2))
print(lyst)
```

3. Write a function, parse(), which takes 1 required argument, csv, which represents the contents of a comma separated values file in a string. Additionally, parse() takes two optional arguments, sep and col. Sep is the character that separates each value; default to a comma if not specified. Col is the number of columns of a 2D List the function should return; default to 3 columns if not specified. In the case of empty cells of the 2D list, add the NaN value using float("NaN").

```
#Example
csvstr = "8;3;1;4;6;2;6;8;3;1;2"
output = parse(csv=csvstr, sep=";", cols=5)
#output = [ [8, 3, 1, 4, 6], [2, 6, 8, 3, 1], [2, NaN, NaN, NaN, NaN]]
```

4. Pig Latin is a language where you mutate every word in a sentence by taking the first letter and moving it to the end of the word and adding the suffix "-ay." If the first letter is a vowel, the first letter is omitted and the suffix becomes "-way" instead. **Write a function, convertWord( ), which takes a string argument and returns a string translated into Pig Latin.**

```
#Example
string = "stars"
converted = convertWord(string)
#converted = "tarssay"
```