# Unit4_Assessment

December 18, 2023

# 1 Unit 4 Career Preparation: Technical Assessment

## 1.1 Problem 1

Write a script that: * Reads the file `problem1.txt`. * Adds each line to a new list. * Prints the new list.

```python
[67]: def reader(file_1):
          new_list = []
          with open (file_1, "r") as f:
              for line in f:
                  line = line.replace("\n","")
                  new_list.append(line)
          return new_list

      def main():
          file_1 = "/voc/public/problem1.txt"
          print(reader(file_1))



      main()
```

```
['item1', 'item2', 'item3', 'item4', 'item5']
```

## 1.2 Problem 2

Write a script that: * Reads the file `problem2.txt`. * Counts how many times 192.168.1.1 appears in the file. * Prints the result.

```python
[83]: def count_ip(file_path):
          ip = input("Enter IP address to count: ")
          with open (file_path, "r") as f:
              count = 0
              for line in f:
                  line = line.strip("\n")
                  if line == ip:
```

```
                count += 1
            else:
                pass
    print(ip + " appears: " + str(count) + "times")

def main():
    file_path = input("enter file path:")
    count_ip(file_path)

main()
```

```
enter file path: /voc/public/problem2.txt
Enter IP address to count:  192.168.1.1

192.168.1.1 appears: 5times
```

## 1.3  Problem 3

Write a script using a function (dedupe) that: * Takes a list l = [1,5,7,2,4,3,5,1,6,2,6]. * Returns a new list that contains all of the elements from the first list, excluding duplicates.

[105]:
```
def dedupe(list_1):
    list_a = set()
    list_b = []
    for x in list_1:
        if x not in list_a:
            list_a.add(x)
            list_b.append(x)
    return list_b

def main():
    list_1 = [1,5,7,2,4,3,5,1,6,2,6]
    new_list = dedupe(list_1)
    print("New list: " + str(new_list))

main()
```

```
New list: [1, 5, 7, 2, 4, 3, 6]
```

## 1.4  Problem 4

Write a program (using a function) that: * Asks the user for a long string containing multiple words. * Prints back the same string, except with the words in reverse order.

For example, if the user types the string: 'My name is robert', it will print 'robert is name My'.

[121]:
```
def boomhauer(string_1):

    words = string_1.split()
```

```
        reverse_words = words[::-1]
        reverse_string = ' '.join(reverse_words)
        return reverse_string

def main():
    string_1 = input("enter string of words to reverse: ")
    result = boomhauer(string_1)
    print("BoomHauer said: " + result)

main()
```

enter string of words to reverse:  wind the in dust ol dang

BoomHauer said: dang ol dust in the wind

## 1.5   Problem 5

Write a script that: * Opens the file `problem5.txt`. * Counts each port and puts the results in a dictionary.

```
[133]: import re
def organize_port(file_name):
    with open (file_name) as f:
        dict_1 = {}
        for port in f.readlines():
            match = re.search("\d{1,4}", port)
            if match:
                port_count = match.group(0)
                if port_count in dict_1:
                    dict_1[port_count] += 1
                else:
                    dict_1[port_count] = 1
            else:
                pass
        return dict_1

def main():

        dict_port_count = organize_port("/voc/public/problem5.txt")
        print(dict_port_count)

main()
```

{'80': 7, '443': 3, '22': 5, '21': 2, '25': 3, '389': 1, '3389': 1, '445': 3}

```
[ ]:
```