Assignment 08: 07 Feb 2023 (Task 2)

Q1. You are writing code for a company. The requirement of the company is that you create a python function that will check whether the password entered by the user is correct or not. The function should take the password as input and return the string "Valid Password" if the entered password follows the below-given password guidelines else it should return "Invalid Password".
Note:

1. The Password should contain at least two uppercase letters and at least two lowercase letters.
2. The Password should contain at least a number and three special characters.
3. The length of the password should be 10 characters long.

Q2. Solve the below-given questions using at least one of the following:
1. Lambda function
2. Filter function
3. Map function
4. List Comprehension

- Check if the string starts with a particular letter.
- Check if the string is numeric.
- Sort a list of tuples having fruit names and their quantity. [("mango",99), ("orange",80), ("grapes", 1000)]
- Find the squares of numbers from to 10. Find the cube root of numbers from to 10.
- Check if a given number is even.
- Filter odd numbers from the given list.
  [1,2,3,4,5,6,7,8,9,10]
- Sort a list of integers into positive and negative integers lists.
  [1,2,3,4,5,6,-1,-2,-3,-4,-5,0]

Q1. You are writing code for a company. The requirement of the company is that you create a python function that will check whether the password entered by the user is correct or not. The function should take the password as input and return the string "Valid Password" if the entered password follows the below-given password guidelines else it should return "Invalid Password".
Note:

1. The Password should contain atleast two uppercase letters and atleast two lowercase letters.
2. The Password should contain atleast a number and three special characters.
3. The length of the password should be 10 characters long.

Ans:

```
import re
import string
def validate_password():
    """Validation for A Given PassWord """ # doc string
    punct=string.punctuation # making a list of all special chr
    while True:
        password = input("Enter a password: ")
        if len(password) < 10: # for accept minimum 10 len password
            print("Invalid Password")
        elif re.search('[0-9]',password) is None: # for atleast 1 digit
```

```
        print("Invalid Password")
    elif len(re.findall('[A-Z]',password)) <2: # for atleast 2 capital letter
        print("Invalid Password")
    elif len(re.findall('[a-z]',password) )<2: # for atleast 2 small letter
        print("Invalid Password")
    elif len([i for i in password if i in punct])<3: # for atleast 3 special character
         print("Invalid Password")
    else:
        print("Valid Password")
        break
validate_password()          # for checking validation
```

Q2. Solve the below-given questions using at least one of the following:

1. Lambda function
2. Filter function
3. Map function
4. List Comprehension

- Check if the string starts with a particular letter.
- Check if the string is numeric.
- Sort a list of tuples having fruit names and their quantity. [("mango",99), ("orange",80), ("grapes", 1000)]
- Find the squares of numbers from 1 to 10. Find the cube root of numbers from to 10.
- Check if a given number is even.
- Filter odd numbers from the given list. [1,2,3,4,5,6,7,8,9,10]
- Sort a list of integers into positive and negative integers lists. [1,2,3,4,5,6,-1,-2,-3,-4,-5,0]

Ans:

```
# a) Check if the string starts with a particular letter
text= "Pwskills"
# checking if the word "Pwskills" has starts with a particular letter P
filter_words=map(lambda x: x[0]=="P", text)
next(filter_words)
#checking with different word same function
text1= "UpGrad"
filter_words=map(lambda x: x[0]=="P", text1)
next(filter_words)
# checking with a list of string
list1=["PomPom", "Promotion", "Good", "Baad"]
filter_words_list=filter(lambda x: x[0]=="P", list1)
for i in filter_words_list:
    print(i)

# b) Check if the string is numeric
text3="123456"
filter_number =map(lambda x: x.isdigit(), text3)
```

```
next(filter_number)

# checking with a list of string
list2=["1234", "a123", "223", "3234a"]
filter_number1 =map(lambda x: x.isdigit(), list2)
for i in filter_number1:
    print(i)

# c) Sort a list of tuples having fruit names and their quantity. [("mango",99),("orange",80), ("grapes", 1000)]
list3=[("mango",99),("orange",80), ("grapes", 1000)]
print("acending sort::::::\n", sorted(list3, key=lambda x: x[1]), "\n" )
print(80*"-")
print()
print("decending sort::::::\n", sorted(list3, key=lambda x: x[1], reverse=True) )

# d.1) Find the squares of numbers from 1 to 10Y
list4=[1,2,3,4,5,6,7,8,9,10] # get the list
sql_list1=map(lambda x: x**2, list4) # make the logic
[i for i in sql_list1] #print the final list
# d.2) Find the cube root of numbers from 1 to 10
sql_list2=map(lambda x: x**3, list4) # make the logic
[i for i in sql_list2] #print the final list

# e) Check if a given number is even
n=[]
num= int(input("enter the number\n")) #int(input("Enter the Number"))
print()
n.append(num)
x=map(lambda y: "even" if y%2==0 else "odd", n)
for i in x:
    print(num, " is a ", i, "number")

# f) Filter odd numbers from the given list. [1,2,3,4,5,6,7,8,9,10]
l10=[1,2,3,4,5,6,7,8,9,10]
print("The Odd Numbers are:::")
[i for i in filter(lambda x: x%2 !=0, l10)]

# g) Sort a list of integers into positive and negative integers lists.[1,2,3,4,5,6,-1,-2,-3,-4,-5,0]
def separation():
    list_x=[1,2,3,4,5,6,-1,-2,-3,-4,-5,0]
    positive_num=sorted([i for i in filter(lambda x: x>=0, list_x)]) #filter the positive number
    negative_num=sorted([i for i in filter(lambda x: x<0, list_x)]) #filter the negative number
    print(positive_num)
    print(negative_num)
separation()
```