

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.

Ans.

```
def pas(s):
```

```
    c=0
```

```
    l=0
```

```
    n=0
```

```
    if(len(s)==10):
```

```
        for i in s:
```

```
            if(i.isupper()):
```

```
                c=c+1
```

```
            elif(i.islower()):
```

```
                l=l+1
```

```
            elif(i.isnum()):
```

```
                n=n+1
```

```
    if(c>=2 and l>=2 and n>=1):
```

```
        return("Valid password")
```

```
    else:
```

```
        return("Invalid password")
```

```
password=input("Enter the password")
```

```
pass_check(list(password))
```

Output:

Enter the password ASHish!@#1

'Valid password'

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 1 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.

- Check if the string starts with a particular letter.

```
string="welcome to data science"
check=lambda value:value.startswith("we")
check(string)
```

OUTPUT:

True

- Check if the string is Numeric.

```
string="1234567890"
check_again=lambda value:value.isnumeric()
check_again(string)
```

OUTPUT:

True

- Sort a list of tuples having fruit names and their quantity.
[("mango",99),("orange",80), ("grapes", 1000)].

```
l=[("mango",99),("orange",80), ("grapes", 1000)]
check=lambda value:value.sort()
check(l)
l
```

OUTPUT:

[('grapes', 1000), ('mango', 99), ('orange', 80)]

- Find the squares of numbers from 1 to 10

```
l=[1,2,3,4,5,6,7,8,9,10]
```

```
def square(n):
```

```
    l=[]
```

```
    for i in n:
```

```
        l.append(i**2)
```

```
    return l
```

```
square(l)
```

OUTPUT:

[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]

- Find the cube root of numbers from 1 to 10

```
def cube(n):
```

```
    l=[]
```

```
    for i in n:
```

```
        l.append(i*(i**2))
```

```
    return l
```

```
cube(l)
```

OUTPUT:

[1, 8, 27, 64, 125, 216, 343, 512, 729, 1000]

- Check if a given number is even

```
number= int(input('Enter the number:'))
Even= lambda x: x%2==0
Even(number)
```

OUTPUT:

Enter the number: 8

True

- Filter odd numbers from the given list.

```
[1,2,3,4,5,6,7,8,9,10]
```

```
l=[1,2,3,4,5,6,7,8,9,10]
```

```
def odd(n):
    k=[]
    for i in n:
        if (i%2)!=0:
            k.append(i)
    return k
```

```
odd(l)
```

OUTPUT:

```
[1, 3, 5, 7, 9]
```

- Sort a list of integers into positive and negative integers lists.

```
[1,2,3,4,5,6,-1,-2,-3,-4,-5,0]
```

```
lists=[1,2,3,4,5,6,-1,-2,-3,-4,-5,0]
```

```
def sort(l):
    m=[]
    n=[]
    for i in l:
        if(i>=0):
            m.append(i)
        else:
            n.append(i)
    return m,n
```

```
k=list(sort(lists))
k
```

OUTPUT:

```
[[1, 2, 3, 4, 5, 6, 0], [-1, -2, -3, -4, -5]]
```