Day - 5

**Loops**

For loop, range()

**5.1 For loop**

**for** item **in** list\_of\_item:

#do something to each item

Here "***item***" is just a variable which is assigned to each elements of the object one by one (until end). This variable will be used in code-block of ***for***.

char = ['a', 'b', 'c', 'd']

**for** ch **in** char:

**print**(ch)

#*python loop.py*

* Exercise 5.1: Average of Heights of some students.

#*following will be*spitted*with respect to*space

#*using split() with input() directly will generate a list*

student\_height = **input**("Enter the list of students height : ")**.split**()

**for** i **in** **range**(0, len(student\_height)):

    student\_height[i] = **int**(student\_height[i])

**print**(student\_height)

sum\_1 = 0

count = 0

#*Don't change code above*

#*Cannot use len() or sum()*

**for** n **in** student\_height:

    sum\_1 += n

    count += 1

average = **round**(**float**(sum\_1)/**float**(count), 3)

**print**("Average of the given lengths : ", average)

#*Shortest way: use the sum() function and len()*

**print**("Use the sum() and len() : ", sum(student\_height)/len(student\_height))

#*python avg\_higt.py*

* However ***sum()*** is built-in function in Python to calculate sum of given values.
* Exercise 5.2: Maximum of numbers.

vaLues = **input**("Enter the list of values : ")**.split**()

**for** i **in** **range**(0, len(vaLues)):

    vaLues[i] = **int**(vaLues[i])

**print**(vaLues)

mAx = vaLues[0]

#*Cannot use max()*

**for** vAl **in** vaLues:

**if** vAl **>** mAx :

        mAx = vAl

**print**(f"hihgest value : {mAx}")

#*Shortest way: use the built-in function- max()*

**print**("Use the max() function : ", max(vaLues))

#*python high.py*

* Note we could set " **mAx = 0**". But this is not a good idea if we are working with **negative –ve** numbers.
* However ***max()*** the built-in function can do the exact job.

**5.2 Range function**

It is similar to classic C/C++ for loops range: , or  **, ie *[a, b)***

**for** number **in** **range**(a, b, step\_size)

**print**(number)

It is used with loop control variable. Notice the following examples:

|  |  |
| --- | --- |
| **for** x **in** **range**(1, 10):  **print**("x : ", x)  **for** y **in** **range**(1, 50, 3):  **print**("y : ", y) | **for** x **in** **range**(1, 11):  **print**("x : ", x)      toTal += x  **print**("total : ", toTal) |

* Exercise 5.3: Summation using loop.

toTal = 0

**for** x **in** **range**(2, 101, 2):

    toTal += x

**print**("total : ", toTal)

#*python loop\_100\_sum.py*

* Exercise 5.4: Fizz Buzz. Divisible by 3 *fizz*; Divisible by 5 *buzz*; Divisible by 3 & 5 *fizzbuzz*

|  |  |
| --- | --- |
| nUm = **int**(input("Enter the terminal value : "))  **for** x **in** **range**(1, nUm+1):  **if** (x % 3)  **==** 0 :  **if** (x % 5) **==** 0 :  **print**("FizzBuzzz")  **print**("Fizz")  **elif** (x % 5) **==** 0 :  **print**("Buzzz")  **else**:  **print**(x) | Joint condition must appear first  #*Other way*  **print**("\n\n\nOther way\n")  **for** y **in** **range**(1, nUm+1):  **if** ((y % 3)  **==** 0) **and** ((y % 5)  **==** 0):  **print**("FizzBuzzz")  **elif** (y % 3)  **==** 0 :  **print**("Fizz")  **elif** (y % 5) **==** 0 :  **print**("Buzzz")  **else**:  **print**(y)  #*python fizzbuzz.py* |

* Exercise 5.5: Password generator.

#*Password generator project*

**import** random

letters = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z', 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z']

numbers = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']

symbols = ['!', '#', '$', '%', '&', '(', ')', '\*', '+']

**print**("Welcome to the PyPassword Generator!")

nr\_letters = **int**(input("How many letters would you like in your password?\n"))

nr\_symbols = **int**(input(f"How many symbols would you like?\n"))

nr\_numbers = **int**(input(f"How many numbers would you like?\n"))

lTr = **len**(letters)-1

sYm = **len**(symbols)-1

nMr = **len**(numbers)-1

#*creating string of random characters*

pw = ''

**for** i **in** **range**(0, nr\_letters):

    pw += letters[random**.randint**(0, lTr)]

**for** i **in** **range**(0, nr\_symbols):

    pw += symbols[random**.randint**(0, sYm)]

**for** i **in** **range**(0, nr\_numbers):

    pw += numbers[random**.randint**(0, nMr)]

**print**("\t\t Un-suffled final Password  : ", pw)

#*========== Suffling the characters ==========*

**if** **len**(pw) **>** 0 :

    sUfl = [pw[0]]

    #*Creating list of password characters*

**for** i **in** **range**(1, len(pw)) :

        sUfl**.append**(pw[i])

**print**("\n\n\t\t List of password characters  Before suffle : ",sUfl)

    #*suffling*

    random**.shuffle**(sUfl)

    #*creating final password string*

    paswrd = ''

**for** i **in** **range**(0, len(sUfl)) :

        paswrd += sUfl[i]

    #*random.shuffle(sUfl) itself nothing; just changes the given list.*

    #*so print("suffoled : ", random.shuffle(sUfl)) not applicable*

**print**("\n\t\tsuffoled password characters : ", sUfl)

**print**("\n\t\tFinal Password : ", paswrd)

**print**("\n\t\tLength Password : ", len(paswrd))

#*python password\_generator.py*

Instructor solution:

#*Password generator project*

**import** random

letters = ['a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i', 'j', 'k', 'l', 'm', 'n', 'o', 'p', 'q', 'r', 's', 't', 'u', 'v', 'w', 'x', 'y', 'z', 'A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J', 'K', 'L', 'M', 'N', 'O', 'P', 'Q', 'R', 'S', 'T', 'U', 'V', 'W', 'X', 'Y', 'Z']

numbers = ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']

symbols = ['!', '#', '$', '%', '&', '(', ')', '\*', '+']

**print**("Welcome to the PyPassword Generator!")

nr\_letters = **int**(input("How many letters would you like in your password?\n"))

nr\_symbols = **int**(input(f"How many symbols would you like?\n"))

nr\_numbers = **int**(input(f"How many numbers would you like?\n"))

#*======   Easy level  ======*

creating string of random characters

passWord = ''

**for** char **in** **range**(1, nr\_letters + 1):

    #*randomly choose letters and add to string*

    passWord += random**.choice**(letters)

**for** char **in** **range**(1, nr\_symbols + 1):

    passWord += random**.choice**(symbols)

**for** char **in** **range**(1, nr\_numbers + 1):

    passWord += random**.choice**(numbers)

**print**("\n\t\tEasy Level password : ", passWord)

#*======     Hard level    ========*

passWord\_list = []

**for** char **in** **range**(1, nr\_letters + 1):

    #*randomly choose letters and add to string*

    passWord\_list**.append**(random**.choice**(letters))

**for** char **in** **range**(1, nr\_symbols + 1):

    passWord\_list**.append**(random**.choice**(symbols))

**for** char **in** **range**(1, nr\_numbers + 1):

    passWord\_list**.append**(random**.choice**(numbers))

**print**(passWord\_list)

random**.shuffle**(passWord\_list)

**print**(passWord\_list)

#*creating final password string*

paswrd = ''

**for** i **in** passWord\_list :

    paswrd += i

**print**(f"\n\n\t\tPassword is : {paswrd} \n")

#*python password\_generator\_angela.py*