Lecture Notes:Week 3

From JOCWiki

Contents

[hide]

Creating Lists[edit]

Code :

Evenings = ["Tea","Coffee","Samosa"]

for snacks in Evenings :

```
   print (snacks)
```

OUTPUT:

Tea

Coffee

```
Samosa
```

Merging 2 lists:[edit]

```
list1 = ['curd','butter','chocolate'] list2 = ['wheat','pulses','burger'] list3 = list1+list2

print(list3)
```

output

```
['curd', 'butter', 'chocolate', 'wheat', 'pulses', 'burger']
```

Reversing a List[edit]

```
Evenings = ["Tea","Coffee","Samosa"]

new_lst = Evenings[::-1]

print(new_lst)

or

use reverse() predefined function

Evenings = ["Tea","Coffee","Samosa"]

Evenings.reverse()

print(Evenings)
```

Appending Lists[edit]

```
Evenings = ["Tea","Coffee","Samosa"]

print (Evenings)

Evenings.append("Chocolate")

print (Evenings)
```

OUTPUT:

['Tea', 'Coffee', 'Samosa'] ['Tea', 'Coffee', 'Samosa', 'Chocolate']

Inserting into a List[edit]

Syntax:

insert(position,item)

Example:

names=["Ankit","Sujata","Ravi","Roshni"]

print(names[2])

names.insert(2,"Rishabh") #inserting item at position 2

print(names[2])

Output: Ravi Rishabh

Finding the length of a List[edit]

Syntax: *len(list_name)*

Example: *names=["Sudarshan","Gupta","sam","umapathy","magesh"] length=len(names) print(length)*

Output: 5

Removing an item from the list[edit]

Items can be removed from the list using pop() and remove() functions.

pop()

pop() function deletes the element at a specified index or position of a list.

Syntax:

list_name.pop(index)

Example:

l=[12,32,423,23]

l.pop(1) #the item at position 1 will be pushed from the list.In this case 32 is at position 1.

print(l)

Output:

[12, 423, 23]

remove()

remove() function deletes the specified item from the list.

Syntax:

list_name.remove(item)

Example:

Note:using the same list as in pop() example.

l.remove(12) #it removes the item 12 from the list

print(l)

Output:

[423, 23]

Please take note that using remove() only the first occurrence of the item will be removed. example : mylist =[12,423,23,12] mylist.remove(12) print(mylist) the output will be [423,23,12]

Sorting the List[edit]

The list can be sorted in ascending or descending order using the method sort(). **Syntax:**

list_name.sort()

Example:

cities=["Nagpur","Wani","Mumbai","Hyderabad"]

cities.sort() #by default sort() prints the list in ascending order.

print(cities)

Output: *['Hyderabad', 'Mumbai', 'Nagpur', 'Wani']*

Reversing the List using *sort()* method

cities.sort(reverse=True) #reverse parameter is use to sort the list in descending or ascending order.By default its value is False.

print(cities)

Ouput:

['Wani', 'Nagpur', 'Mumbai', 'Hyderabad']

Fibonacci Series Using For Loop[edit]

a=0

b=1

n= int(input())

print(b)

for i in range(n):

```
    c=a+b
    print(c)
    a=b
    b=c
```

Passed after ignoring Presentation Error[edit]

To avoid this presentation error use below code

CODE: list1=[1,2,3,4,5]

for i in range(0, len(list1)):

```
    if(i < len(list1)-1):
        print(list1[i],end=" ")
    else:
        print(list1[i],end="")
```

OUTPUT: 1 2 3 4 5

EXPLANATION: (1) if(i < len(list1)-1): this code will help you to find the last variable.

(2) print(list1[i],end=" "):code helps give space after variable printing

(3) print(list1[i],end=""): code does not give space after variable printing

(4) if condition true means it is not last variable (2) code will execute or false means it is last variable (3) will execute

Check whether first and last element of an array is same[edit]

n = list(map(int,input().split()))

if n[0] == n[-1]:

```
    print(True)
```

else:

```
    print(False)
```

File operations[edit]

Opening a file: *A file can be opened using the open().*

Syntax: *open(file_name,file_mode)*

Example: *with open("file1.txt","r") as file_1:*

1. *Remember if the file is opened using the read mode then file must exist. Else it will give "No such file or directory:".*

2.In "w"(write) mode,if the file does not exist then a new file is created.It does not happen in "r" mode.

Reading from a file:

To read from a file,open the file in "r" or "r+" mode.Then use the function read() to read the contents of the file.

Syntax:

file_object.read()

Example:

with open("file1.txt","r+") as file_1:

```
    a=file_1.read()
```

Writing to the file

To write to the file use the write() function.

1. *Remember the write() function first erases the contents of the file, then writes to it.*

Syntax:

file_object.write(content)

Example:

```
file_1.write("This is a new line added to the file")
```

Library[edit]

A library is a collection of subroutines and functions which are ready to be executed. These subroutines/functions provide added functionality.They can be imported by the user in the program.

Syntax for importing the library:

import library_name

Syntax for using the functions of library:

library_name.function_name(argument)

Some of the examples of libraries are as follows:

statistics

```
      This library is use for mathematical statistical operations.
```

Some of the functions included:

statistics.mean():To find the mean value of data.

statistics.median():To find the median(middle) value of the data.

Example:

import statistics

data=[12,23,13,43,22,333,65,44,33,98,656,45,454,86,30,28,63]

data.sort()

ar=statistics.median(data)

print(ar)

ar=statistics.mean(data)

print(ar)

Output *: 44 120.47058823529412*

random

```
   This library is use for selecting random value from a sequence or from a range of values.
```

Some of the functions included:

random.randint(start,end) : It returns a random integer from a range of integers.

random.randrange(start,end): It returns a random value from a range of values. Remember the end value is excluded.

random.choice(list): It returns a random value from the list.

Example:

import random

data=[12,32,123,323,2,23,2,32323,23,2323,23,2323,2323]

ar=random.choice(data)

print(ar)

ar=random.randint(1,10) #10 will be included.

print(ar)

ar=random.randrange(1,6) #6 will be excluded.

print(ar)

Output: *23 10 5*

- *Note: For more information on libraries in Python visit the following link: https://docs.python.org/3/library*

Second max and second min[edit]

```
a=list(map(int,input().split()))
print(sorted(a))
a.sort()
if(a[::-1].index(min(a))):
```

```
   print("Min2= ",a[len(a)-a[::-1].index(min(a))]," Max2= ",a[a.index(max(a))-1],end="")
else:

   print("Min2= ",a[0]," Max2= ",a[0],end="")
```

Jumbled word game[edit]

```
#Jumbled word game(2 players)
import random

def choose():



words=["computer","science","zoology","chemistry","mathematics","rainbow","kolkata","banaras","university","souvenir","p
up"]
   return random.choice(words)         #choose random word from words_list
def jumble(word):

   jumbled="".join(random.sample(word,len(word)))#"delimeter".join(list_name) && sample(list_name,no_of
elements_selected randomly)
   return jumbled
def play():

   player=[input("Enter name of player 1:"),input("Enter name of player 2:")]
   turn=0      #turn= 0 for 1st player and 1 for 2nd player
   point=[0,0] #counting points of players
   life=[3,3]  #three lives for each player
   while(1):
       #printing score status of both players
       print(player[0]+"----> Point: ",point[0]," ; Lives: ",life[0])
```

```
      print(player[1]+"----> Point: ",point[1]," ; Lives: ",life[1])
      picked_word=choose()    #a word is chosen
      question="What word is meant by this jumbled word: "+jumble(picked_word)+" ?"    #word is jumbled and question is
preapared
      if(life[turn]==1): print(player[turn]," has last life left")
      answer=input(player[turn]+"'s turn to play\n"+question)    #Question asked for proper player and answer is
recorded
      if(answer==picked_word):            #checked whether the answer is correct
          point[turn]=point[turn]+1       #point of proper player is increased
          print("Correct answer!! Great going!!")
      else:
          life[turn]=life[turn]-1         #decreasing life of appropriate player if answer is wrong
          print("Oops!! Wrong answer... You lost one life...\nCorrect answer is ",picked_word)
      #setting turn for the next player
      if(turn):
          turn=turn-1
      else:
          turn=turn+1
      #if both player has lives the continue the loop
      if(life[0]==0 or life[1]==0):
          break
  #determining winner
  if(point[0]>point[1]):
      print(player[0]," won the match")
  elif(point[1]>point[0]):
      print(player[1]," won the match")
  else:
```

```
        if(life[0]>life[1]):
            print(player[0]," won the match")
        elif(life[1]>life[0]):
            print(player[1]," won the match")
        else:
            print("Match is drawn")
    #thanking both players
    print("Thanks "+player[0]+" and "+player[1]+" for playing Jumanji jumbled word game")
#main starts here
print("=====================================================")
print("|| JUMANJI JUMBLE WORD GAME ||")
print("=====================================================")
print("RULES:\n1)Two players will play the game\n2)Each player has three lives")
print("3)Every question will have a jumbled version of a word, each player has to type the original version of the word in his/her turn")
print("4)Each correct answer will award 1 point to the appropriate player")
print("5)Each wrong answer will cost a life of the appropriate player")
print("6)Game will end if one of the player looses all lives")
print("7)Player with maximum points (or in case of equal points player with max. lives) will win")
print("--------------------GOOD LUCK-----------------------")
play()
```

Print those numbers in list which are not divisible by 5: IN A NEW WAY[edit]

```
a=list(map(int,input().split()))
a=[a[i] for i in range(len(a)) if a[i]%5]
```

```
print(a)
```