PYTHON

Lecture - 05

Recap

- Print formats
- User Input
- Control Statements (If-elif-else)
- **Homework (Problem Based on if-else statements)**
- **Class work**

Write a Python program that takes the *age* of a person as input and classifies them into one of the following age groups:

Child: 0-12 years
Teen: 13-19 years
Adult: 20-64 years
Senior: 65 years and above

Contents

- Loop statements

- Write a python code that prints your name 10 times.

Sample Output:

Donald Trump
Donald Trump
Donald Trump
Donald Trump
Donald Trump
Donald Trump
Donald Trump
Donald Trump
Donald Trump
Donald Trump

Loops in Python

- Python primarily offers two types of loops:
  - While: while loop in Python repeatedly executes a block of code as long as a specified condition remains True.
  - For: for loop in Python is used to iterate over a sequence *(such as a list, tuple, string, or range)* and execute a block of code for each element in the sequence

While in Python

- *syntax*

```
initialization
while expression:
    statement(s)
        inc/dec
```

```
i = 1
while i < 6:
  print(i)
  i += 1
```

While in Python (Example)

- *Print your name 10 times:*

```
i = 1
while i <=10:
  print(“Donald Trump”)
  i += 1
```

- *Now try to print all name in a single line with separated by comma [hints: use end=“”]*
- *Print number 1 to 20*

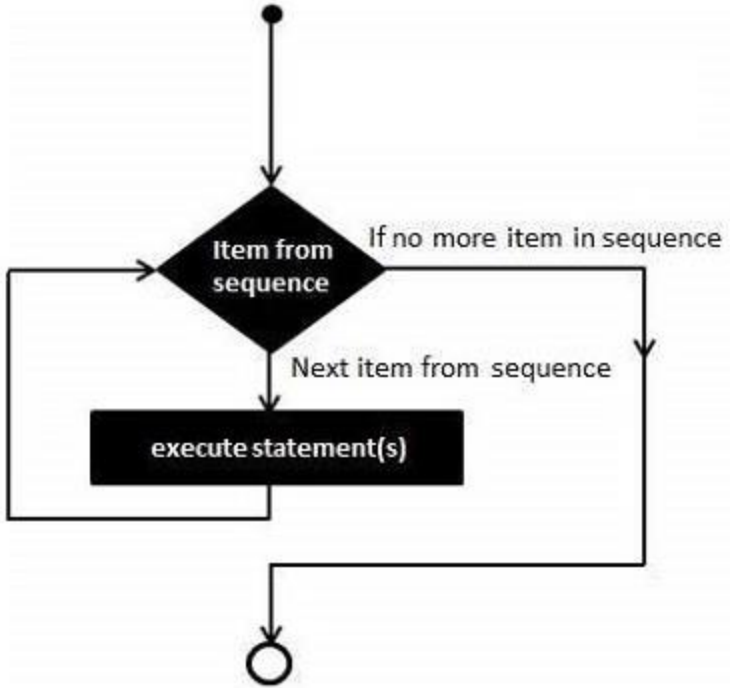
for in Python (Example)

- A `for` loop is used for iterating over a sequence (that is either a ***list, a tuple, a dictionary, a set, or a string***).
- This is less like the `for` keyword in other programming languages, and works more like an iterator method as found in other object-orientated programming languages.
- With the `for` loop we can execute a set of statements, once for each item in a list, tuple, set etc.

```
fruits = ["apple", "banana", "cherry"]
for x in fruits:
  print(x)
```

Python For Loop (Syntax)

```
for iterating_var in sequence:
    statements(s)
```



Python For Loop (Example - 1)

Iterate over a List and print it's elements.

```
cities = ['Barisal', 'Dhaka', 'Khulna',
'Sylhet']
for city in cities:
   print(city)
```

OUTPUT

Barisal
Dhaka
Khulna
Sylhet

Python For Loop (Example - 2)

Given last 5 days temperature in a list. Find its average.

```
temp = [35.8, 39.6, 38.2, 40.8, 41.2]
sum = 0.0
for x in temp:
    sum+=x

avg = sum/len(temp)
print(avg)
```

OUTPUT

39.12

Python For Loop (Example - 3)

Print all chars of a given string

```
s = "university"
for i in s:
   print(i)
```

OUTPUT

u
n
i
v
e
r
s
i
t
y

Python For Loop with range() Function

range(start, stop, step) → returns a List of int datatype

Start – Starting value of the range. **Optional**. Default is 0

Stop – The range goes upto stop-1

Step – Integers in the range increment by the step value. **Optional**, default is 1.

range(2,10,2) → 2, 4, 6,8
range(6) → 0, 1, 2, 3, 4, 5

Python For Loop (Example-01)-Using range

Iterate over a List and print it's elements.

```
cities = ['Barisal', 'Dhaka', 'Khulna',
'Sylhet']
for i in range(len(cities)):
  print(cities[i])
```

OUTPUT

Barisal
Dhaka
Khulna
Sylhet

Python For Loop (Example - 2)

Given last 5 days temperature in a list. Find its average.

```
temp = [35.8, 39.6, 38.2, 40.8, 41.2]
sum = 0.0
for i in range(len(temp)):
  sum+=temp[i]
 avg = sum/len(temp)
print(avg)
```

OUTPUT

39.12

Python For Loop (Example - 3)

Print all chars of a given string

```
s = "university"
for i in range(len(s)):
  print(s[i])
```

OUTPUT

u
n
i
v
e
r
s
i
t
y

Python Loop Control Statement

Sr.No.	Control Statement & Description
1	**break statement** Terminates the loop statement and transfers execution to the statement immediately following the loop.
2	**continue statement** Causes the loop to skip the remainder of its body and immediately retest its condition prior to reiterating.
3	**pass statement** The pass statement in Python is used when a statement is required syntactically but you do not want any command or code to execute.

Python Loop Control Statement (break)

Find if the given list has an even value or not.

```
List = [3, 5, 6, 7, 10]
even = 0
for x in List:
   if x%2==0:
       even = 1
       break

if even:
   print("Found")
else:
   print("Not Found")
```

OUTPUT

Found

Python Loop Control Statement (continue)

Given a list. Print all odd numbers from the list.

```
List = [3, 5, 6, 7, 10]
for x in List:
  if x%2==0:
     continue
  print(x)
```

OUTPUT

```
3
5
7
```

Python Loop Control Statement (pass)

It is a null operation; nothing happens when it executes. Python pass statement is also useful in places where your code will eventually go, but has not been written yet.

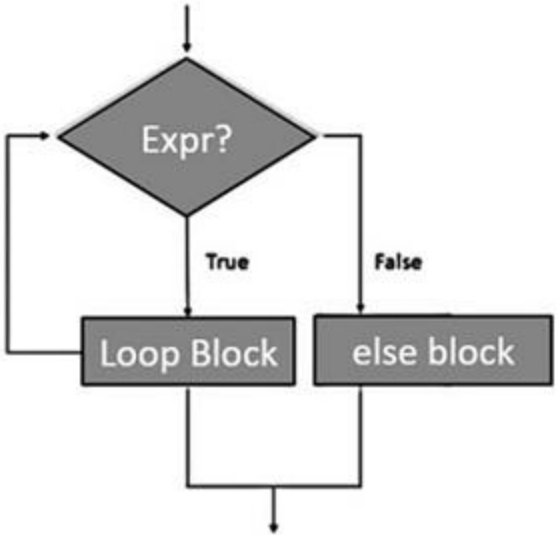
```
s = "university"
for i in range(len(s)):
  pass
```

OUTPUT

#The code print nothing.

Python for-else Loop (Syntax)

```
for variable_name in iterable:
 #stmts in the loop
 .
 .
else:
 #stmts in else clause
 .
 .
```



Python for-else Loop (Example)

Find if the given list has an even value or not.

```
List = [3, 5, 11, 7, 9]
for x in List:
  if x%2==0:
     print("Found")
     break
else:
  print("Not Found")
```

OUTPUT

Not Found

Python for-else Loop (Example)

Find if the given list has an even value or not.

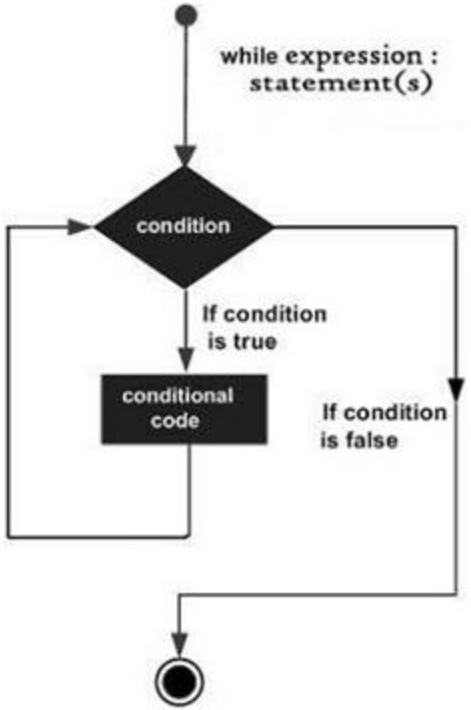
```
List = [3, 5, 10, 7, 9]

for x in List:
  if x%2==0:
     print("Found")
     break
else:
  print("Not Found")
```

OUTPUT

Found

Python While Loop (Syntax)

```
initialization
while expression:
    statement(s)
        inc/dec
```



Python While Loop (Example - 1)

Iterate over a List and print it's elements.

```
cities = ['Barisal', 'Dhaka', 'Khulna',
'Sylhet']
i = 0
while(i<len(cities)):
  print(cities[i])
  i+=1
```

OUTPUT

Barisal
Dhaka
Khulna
Sylhet

Python While Loop (Example - 2)

Given last 5 days temperature in a list. Find its average.

```
temp = [35.8, 39.6, 38.2, 40.8, 41.2]
sum = 0.0
i = 0
while i<len(temp):
  sum+=temp[i]
  i+=1
avg = sum/len(temp)
print(avg)
```

OUTPUT

39.12

Python While Loop (Example - 3)

Print all chars of a given string

```
s = "university"
i = 0
while i<len(s):
  print(s[i])
  i+=1
```

OUTPUT

u
n
i
v
e
r
s
i
t
y

Python Nested Loop (Syntax)

```
for iterating_var in sequence:
    for iterating_var in sequence:
        statements(s)
    statements(s)
```

Python Nested Loop (Example - 1)

Print all prime numbers upto 100

```
num = 2
while(num <= 100):
  j = 2
  while(j <= (num/j)):
     if not(num%j):
        break
     j = j + 1
  if(j > num/j):
     print(num,"is prime")
  num = num + 1
```

OUTPUT

2 is prime
3 is prime
5 is prime
…
…
89 is prime
97 is prime

Exercise – Bonus_01

- Find the sum of numbers from **1** to **5**.
- Find the sum of numbers from **1** to **10**.
- Given **N**. Find the sum of numbers from **1** to **N**.
- Print all numbers for 10 to 1.

Exercise-Homework

- Write a Python program that prints the multiplication table of a given number n.
- Write a Python program that calculates the factorial of a given number n. The factorial of a number is the product of all positive integers up to that number.
- Write a Python program that prints all even numbers from 1 to a given number n.
- Write a Python program that prints the numbers from 1 to 50. For multiples of 3, print "Fizz" instead of the number, and for multiples of 5, print "Buzz". For numbers which are multiples of both 3 and 5, print "FizzBuzz".
- Write a Python program that takes a list of numbers and finds the largest number in the list using a loop.

Exercise-Homework (Age Group)

Write a Python program that takes a list of ages as input and classifies each age into one of the following age groups:

Child: 0-12 years
Teen: 13-19 years
Adult: 20-64 years
Senior: 65 years and above

The program should then count and print the number of individuals in each age group.

Example:

Given the list of ages= [5, 17, 24, 13, 45, 67, 89, 15, 33, 12, 18, 64, 65, 70]

Sample Output:

Children: 3
Teens: 4
Adults: 5
Seniors: 3

Exam Announcement

- Quiz Exam (10%):
  - -Date: Next Saturday (02 Nov)
  - -Quiz: mcq and short question
  - -lab Test: you have to write program for a given problem.
  - -Syllabus: Till today
- Midterm (20%):
  - -Date: Saturday (09 Nov)
  - -Quiz: mcq and short question
  - -lab Test: you have to write program for given problems.
  - -Syllabus: Upto previous class of exam day.

Resources

- https://www.tutorialspoint.com/python/index.htm
- https://www.w3resource.com/python/python-tutorial.php
- https://www.w3resource.com/python-exercises/string/
- https://www.w3schools.com/python/
- https://www.geeksforgeeks.org/python-programming-language/
- https://youtu.be/t2_Q2BRzeEE?si=OO6J_YNCZykedqsT
- https://realpython.com/
- Head First Python, 3rd Edition by Paul Barry
- Automate the Boring Stuff with Python By Al Sweigart.

Thank You