Python Flow Control:

for loop

REVIEW: while Loop

A while Loop executes a block of code a certain number of times as long as the condition is True

for Loop

A for Loop executes a block of code a certain number of times.

- You know when to start
- and when to stop
- and what to increase or decrease by.

for Loop:

- Used to iterate through a sequence
 - (like a "string", numbers, [lists], etc.)
- Iteration is a repeat of an action a predefined number of times.

for Loop SYNTAX:

```
for iteration_variable in range(start, stop, step):
   body of for loop
   to be repeated
```

range() method — takes up to 3 arguments

range(start, stop, step)

- start starting integer
- stop stopping integer 1
- step (Optional) increase or decrease integer value(can be a positive or negative number)

range() method -

if only 1 argument

for num in range(5):

body of for loop

to be repeated

#range(stop),
#num is
iteration_variable
If only 1 argument,
starts at 0,
STOPS at num -1
increments by 1

range() method -

if 2 arguments

for num in range(2, 5):
body of for loop
to be repeated

#range(start, stop),
#If 2 arguments,
start is first number,
STOP is at num -1,
increments by 1.

range() method -

if 3 arguments

#range(start, stop [,step])

for num in range(2, 10, 2):

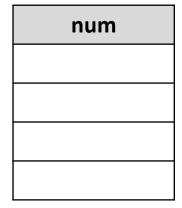
body of for loop to be repeated

Enter Code in: today's_date.py

for num in range(5):
print(num)

What is OUTPUT?

Iteration variable num starts at 0 up to **NOT** including 5.



for Loops and range method Syntax

num starts at 0 up to not including 5.

```
for num in range(5):

print(num, end = "")
```

for Loops Syntax

```
for num in 5:
print(num, end = "")
```

#Error 'int' object is not iterable.

#Iteration is a group of something

#(string or number) to go one after

#another.

for Loops Syntax (start and stop values)

```
for num in range(12, 16):

print(num, end = " ")
```

for Loops Syntax (start, stop and step values)

```
for num in range(3, 8, 2):

print(num, end = " ")
```

for Loops Syntax (range negative values)

```
for num in range(5, -1, -1):

print(num, end = " ")
```

break and continue

Can also be used in for Loops.

Enter Code in: forloop.py

1. Write a for loop to print the year [1 - 10]. Use year as your iteration variable.

2. Write a **for loop** to print the salary from [\$10,000 - \$50,000] in increments of 500. Use **salary** as your **iteration variable**.

3. Write a **for loop** to print temperature from [100 – 0], decreasing by 10. Use **tempC** as your **iteration variable**.

Search the Internet for more for loop Examples