CS 1112: Introduction To Programming

Loops: Break and Continue

Dr. Nada Basit // basit[*at*]Virginia[*dot*]edu

Friendly Reminders

- Your ***safety*** and ***comfort*** is important!
  - If you choose to wear a mask you are welcome to do so
  - *We will interpret wearing a mask as being considerate and caring of others in the classroom (not that you are sick), and realize that some may choose to mask to remain distanced*
- Remember to always be **kind, respectful, supportive, compassionate and mindful of others!** ☺
- Be an ***active*** participant in your learning!
  You're welcome and ***encouraged*** to ask questions during class!
- If you feel ***unwell***, or think you are, please stay home
  - *Contact us!* *We will work with you!*
  - Get some rest ☺
  - View the recorded lectures – *please allow 24-48 hours to post*

Announcements

- **Quiz 4** is being graded
- **PA03** is due by 11:00pm on Wednesday (***tonight***)!
  - Submit on Gradescope: your .py file
- **Exam 1** is coming up… on February 26, 2025!
  - If you have **SDAC** *time and/or distraction-free accommodations*, please **book** a time slot with SDAC to take the exam at their facility (*any time* on Feb. 26, not another day, please!)
  - Review session: February 24 (the class before the exam)

Python Demonstration

Let's jump on PyCharm!

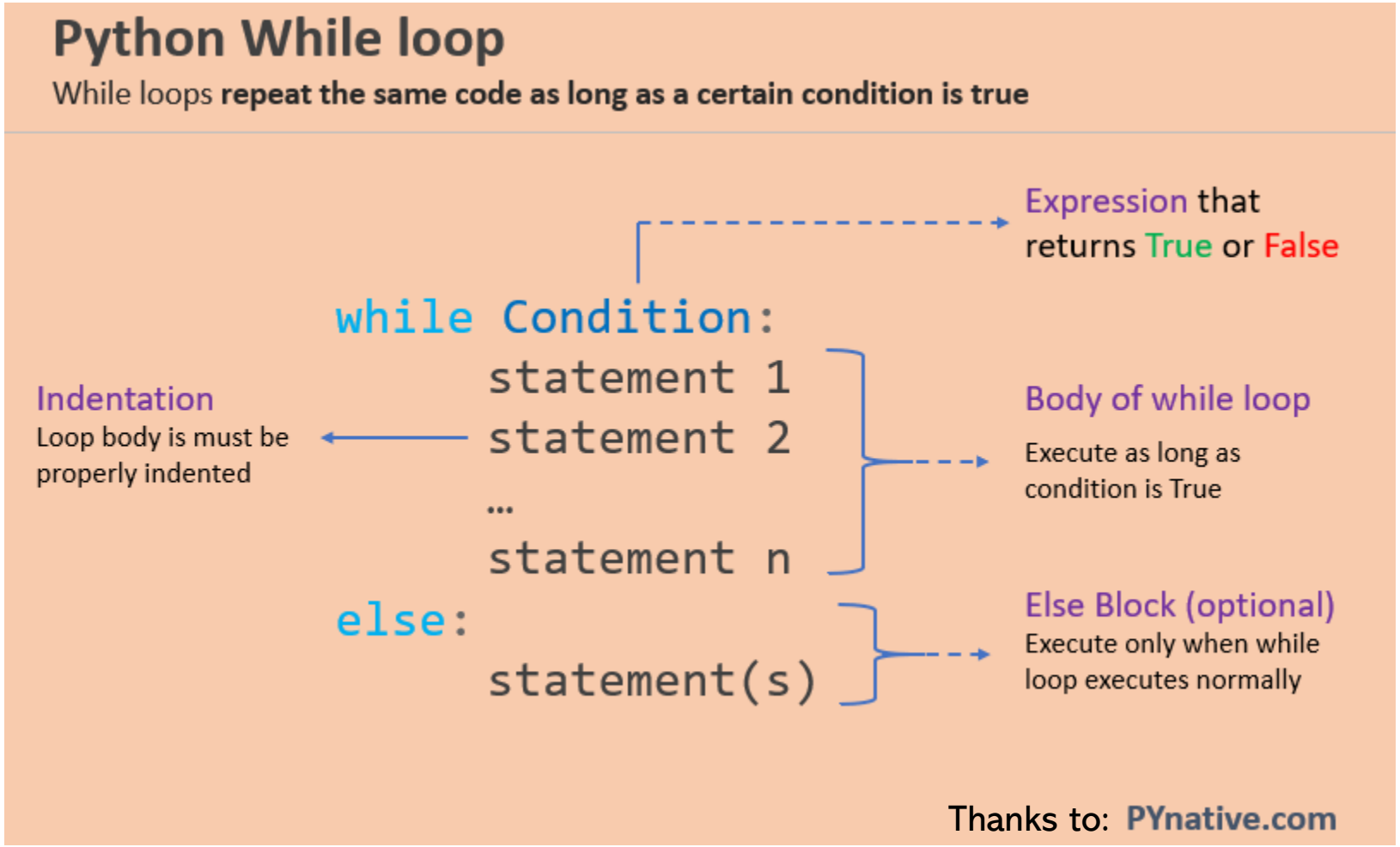
`loops_ica_sol.py`

Let's review the solution to the last in-class "lab" activity.

Quick Review

For-loop and While-loop

Review: While-loop



While-loop Example: Assure proper user input

```
number = int(input('Enter any number between 100 and 500 '))
# number greater than 100 and less than 500
while number < 100 or number > 500:
    print('Incorrect number, Please enter correct number:')
    number = int(input('Enter a Number between 100 and 500 '))
else:
    print("Given Number is correct", number)
```

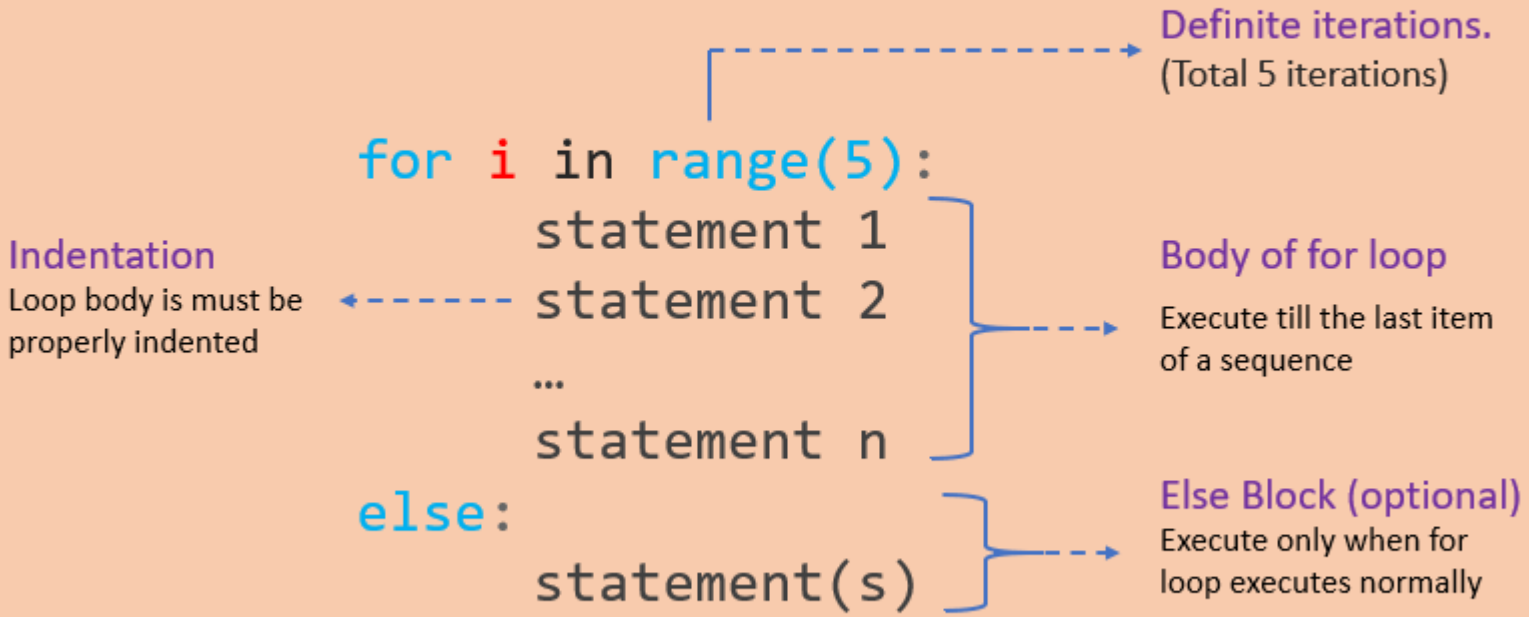
Run of Program:

Enter any number between 100 and 500 700
Incorrect number, Please enter correct number:
Enter a Number between 100 and 500 98
Incorrect number, Please enter correct number:
Enter a Number between 100 and 500 300
Given Number is correct 300

Review: For-loop

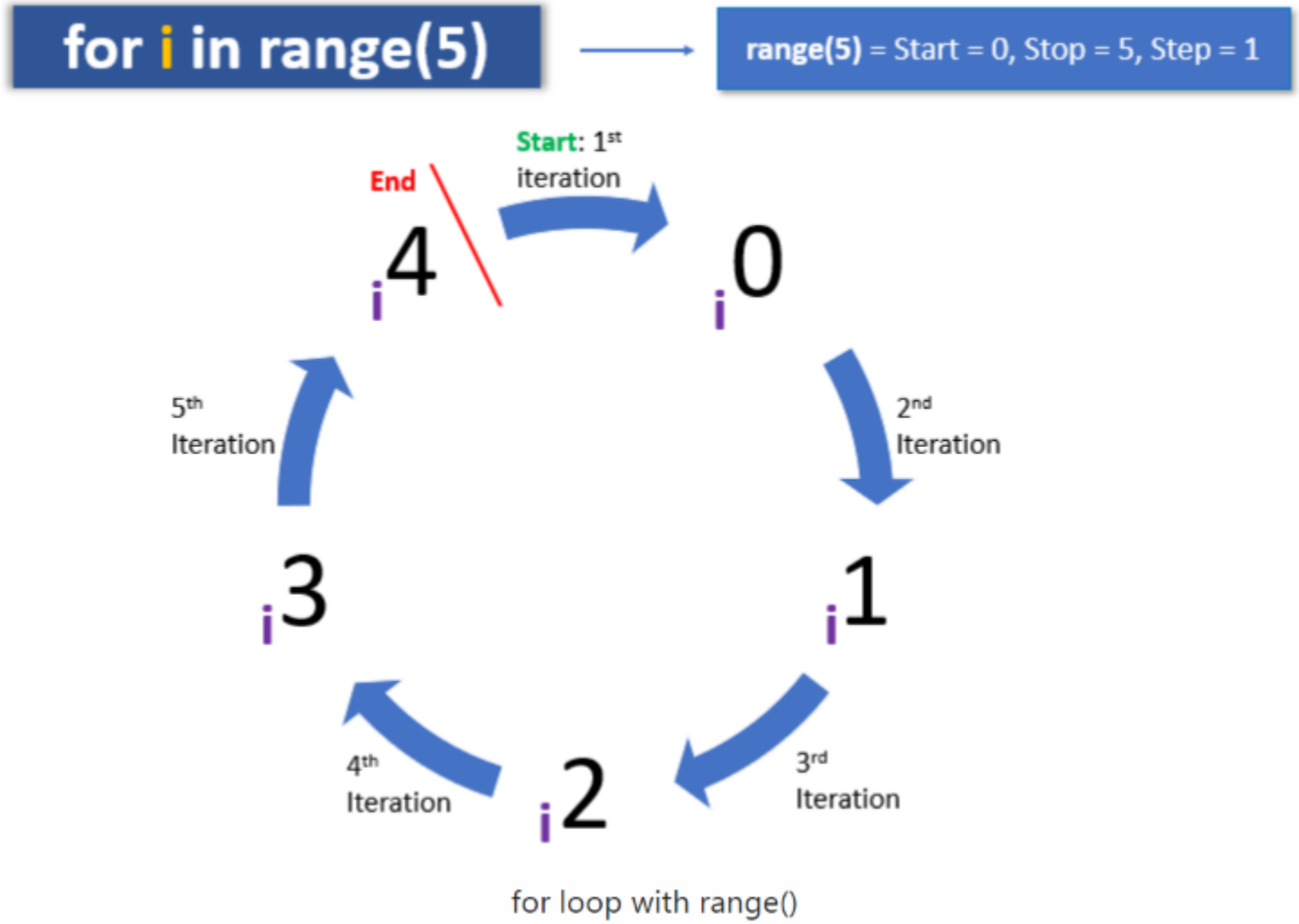
Python for loop

A for loop is **used for iterating over a sequence and iterables** (like range, list, a tuple, a dictionary, a set, or a string).



Thanks to: **PYnative.com**

For-loop Example: for i in range ( )



For-loop Example: with a list

```
numbers = [1, 2, 3, 4, 5]
# iterate over each element in list num
for i in numbers:
    # ** exponent operator
    square = i ** 2
    print("Square of:", i, "is:", square)
```

Run of Program:

Square of: 1 is: 1
Square of: 2 is: 4
Square of: 3 is: 9
Square of: 4 is: 16
Square of: 5 is: 25

Python Demonstration

Let's jump on PyCharm!

`loop_conversion.py` - converting between for-loops and while-loops

Quick & Fun Survey Questions

Get to know your peers! ☺

Starbucks or Dunkin?

Loop Control Statements: Break and Continue

Useful and interesting statements used in conjunction with loops

Loop control statements **change the execution** of the normal functioning of the loop

They are used if you want to **exit** a loop or **skip** a part of the loop based on a **condition**

Thanks to PYnative for some of these examples

Loop Control Statement
"break" – *exit the loop*

- You can use the **break** statement whenever you want **to stop (or quit) the loop early**, if some condition is met. ➔ Uses an **if-statement**!

- When the **break** statement is encountered, Python stops the current loop, and the control flow is transferred to the following line of code immediately following the loop.

- **Example**:
  This loop displays each character from a string and if the character is a number, then stop the loop

```
name = 'Jesaa29Roy'
size = len(name)
i = 0
# iterate loop till the last character
while i < size:
    # break loop if current character is number
    if name[i].isdecimal():
        break;
    # print current character
    print(name[i], end=' ')
    i = i + 1
```

Output:

```
J e s a a
```

Loop Control Statement
"break" – *exit the loop*

- Another example (first, regular for-loop example, then one using break):

```
cities = ['Charlottesville','New York','SF','Portland','LA']
for city in cities:
    city = city.lower()
    print(city)
```

```
# Now, instead of iterating over all the cities, quit early if SF is reached
cities = ['Charlottesville','New York','SF','Portland','LA']
for city in cities:
    if city == 'SF':   # Notice the condition
        break
    city = city.lower()
    print(city)
```

Output:

```
???
```

Loop Control Statement
"break" – *exit the loop*

- Another example (first, regular for-loop example, then one using break):

```
cities = ['Charlottesville','New York','SF','Portland','LA']
for city in cities:
    city = city.lower()
    print(city)
```

```
# Now, instead of iterating over all the cities, quit early if SF is reached
cities = ['Charlottesville','New York','SF','Portland','LA']
for city in cities:
    if city == 'SF':   # Notice the condition
        break
    city = city.lower()
    print(city)
```

Output:

```
charlottesville
new york
```

```
for val in sequence:
    # code
    if condition:
        break

    # code
```

```
while condition:
    # code
    if condition:
        break

    # code
```

“break” –

exit the loop

Loop Control Statement
"continue" – *stop the current iteration*

- Sometimes you want to introduce **skipping behavior** in the loop.
- The **continue** statement skips the current iteration of a loop and immediately jumps to the next iteration, if some condition is met. ➔ Uses an **if-statement**!
  - In other words, when the **continue** statement is encountered inside the loop, the Python interpreter skips the remaining code and moves to the beginning of the next iteration

```
for i in range(5):
    if i == 3:  # skip the number 3!
        continue
    print(i)
```

Output:

```
0
1
2
4
```

Loop Control Statement
"continue" – *stop the current iteration*

- Here is an example program to print odd numbers from 1 to 10
- When the number is even, the **continue** statement skips the current iteration and starts the next iteration. As a result, the even numbers do not get printed!

```
num = 0

while num < 10:
    num += 1

    if (num % 2) == 0:
        continue

    print(num)
```

Output:

```
1
3
5
7
9
```

Loop Control Statement

"continue" – *stop the current iteration*

```
name = 'Jesaa29Roy'

size = len(name)
i = -1  <-------- Variable "i" is initialized to -1
# iterate loop till the last character
while i < size - 1:
    i = i + 1 <-------- Variable "i" initially starts at 0
    # skip while loop body if current character is not alphabet
    if not name[i].isalpha():
        continue
    # print current character
    print(name[i], end=' ')
```

Output:

J e s a a R o y

```
for val in sequence:
    # code
    if condition:
        continue

    # code
```

```
while condition:
    # code
    if condition:
        continue

    # code
```

“continue” –

stop the current iteration

Python Demonstration

Let's jump on PyCharm!

`break_and_continue.py` - example code using "break" and "continue"

In-Class "lab" Activity!

Activity for Today!

- In pairs or groups up to three work on the following activity.
- **password_ica.py**
- *Practice writing a solution that requires you to use "break" and "continue"*

Remember to check-in with a TA before leaving class today!

Reminder: CS Laptop Loaner Program

- This course requires students to have a **laptop**
- I realize that not everybody might have one (nor necessarily need one for their desired major / path…)
- If you do not have a laptop for any reason… ***not to worry!***
- The CS department's Systems staff has a notebook / laptop loaner program and will be able to loan you a notebook / laptop computer for the duration of the semester if you don't have one or if you cannot afford one.
  - Also available if your laptop is broken and under repair, we can arrange for you to receive a loaner laptop for a week or two until your own laptop is fixed

Interested? Link: https://www.cs.virginia.edu/wiki/doku.php?id=cs_laptop_loaner

I am happy to be your sponsor. Please let me know.