Software Engineering Bootcamp

Hyperiondev

Beginner Control Structures

Welcome

Your Lecturer for This Session

Yolandi Viljoen

RULES AND REGULATIONS FOR LIVE LECTURES

❑ The use of disrespectful language is prohibited.

❑ Please refer all non-academic queries to support@hyperiondev.com

❑ You are more than welcome to ask questions, however, please keep them related to the topic being discussed.

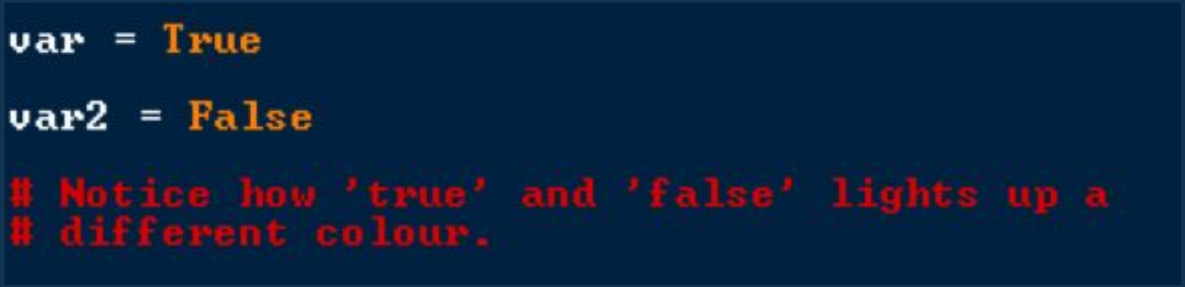
❑ There is a Q/A session at the end of each session, should you wish to ask any follow-up questions.

Objectives

- Learn how to use if, elif, and else statements to make decisions in your programs.
- Learning how booleans work.

Booleans

- ★ Booleans can only be stored as one of two things: True or False.
- ★ Mainly used for conditional checks.
- ★ Booleans should be declared in Python with capitals. Using lowercase for booleans will return an error in Python.
- ★ Example:



```
var = True

var2 = False

# Notice how 'true' and 'false' lights up a
# different colour.
```

Truthiness

In Python, all conditional checks resolve to True or False.

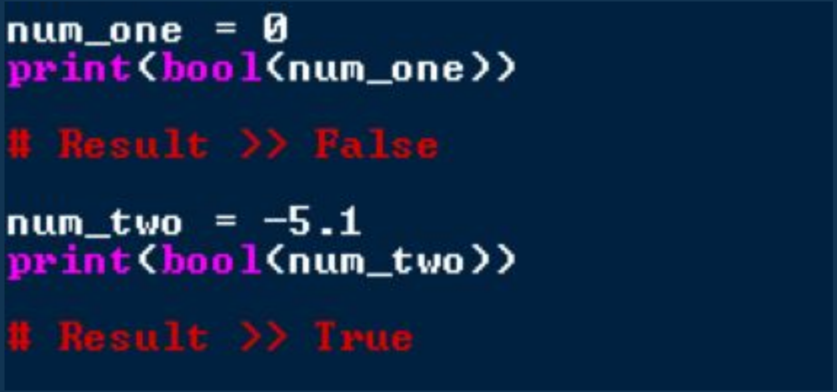
For Example:

X = 1
X is 1 >> TRUE
X is 0 >> FALSE

We can call values that result to True as "**truthy**" , or values that results to False as "**falsy**".
Besides False conditional checks, there are other things that are naturally falsy. These include: empty objects, empty strings, None and zero.

Integers and Floats as Booleans

- ★ Both integers and floating point numbers can be converted to boolean using the bool() function.
- ★ An int, float, or complex number set to zero returns as False.
- ★ An int, float, or complex number set to any value that is not zero, returns as True.



```
num_one = 0
print(bool(num_one))

# Result >> False

num_two = -5.1
print(bool(num_two))

# Result >> True
```

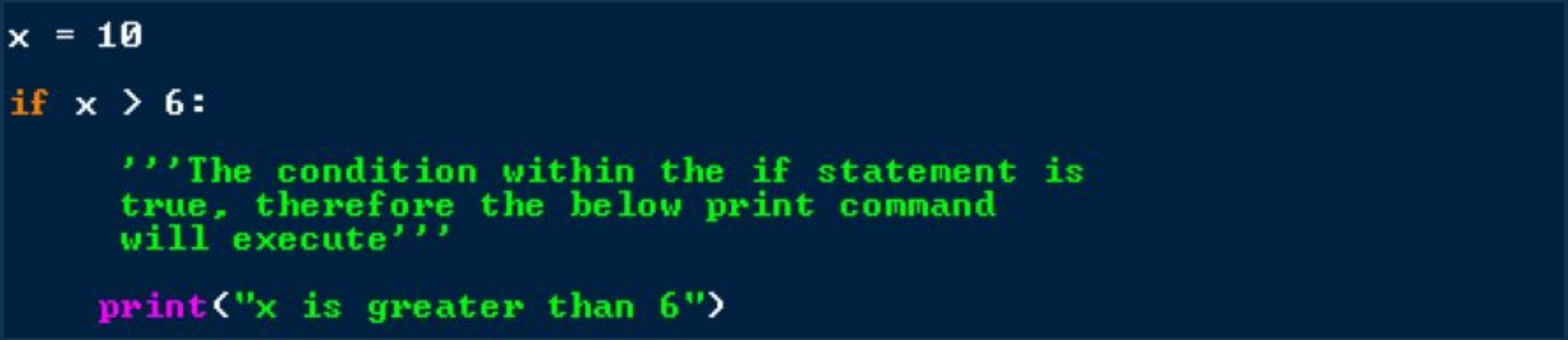
Control Structures

- ★ Control structures are code that will analyse variables and then choose a direction to follow based on the input provided.
- ★ Think of it as a form of branching: depending on the provided input, your program will have one of *x* branches to follow.
    - ○ e.g. “If I finish my work early, I will go to bed. Else, I will have to work through the night”.

If Statement & Syntax

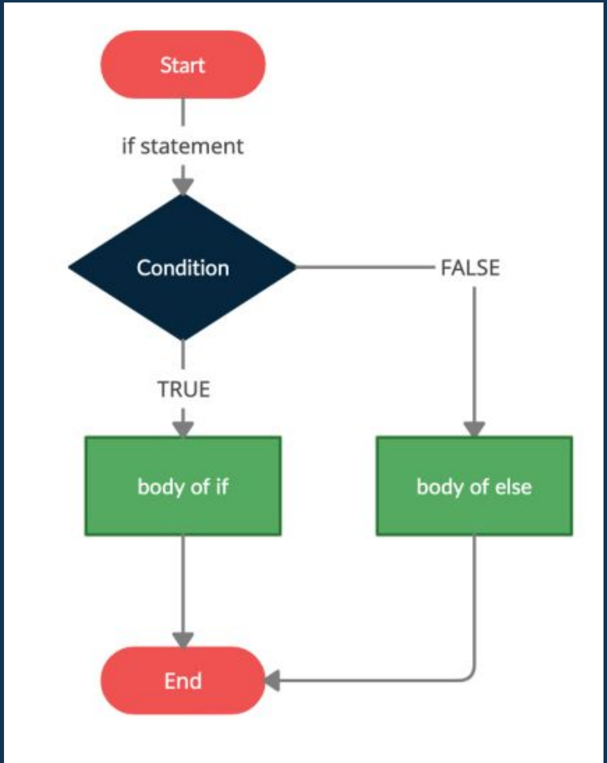
```
if <condition>:
    <statements>
```

Example:



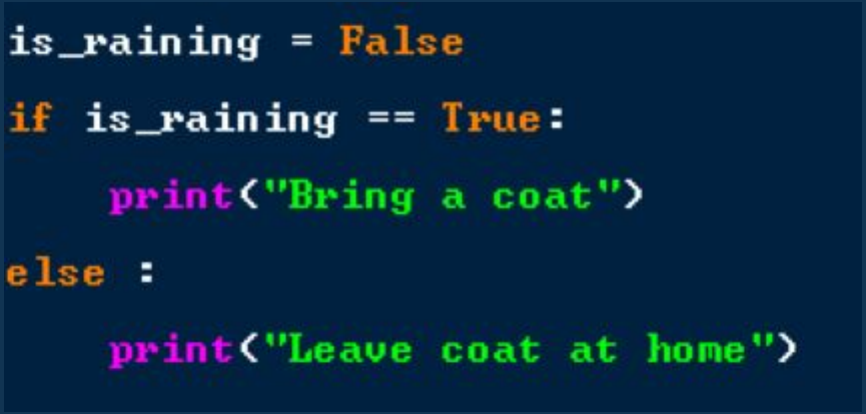
```
x = 10

if x > 6:

     '''The condition within the if statement is
     true, therefore the below print command
     will execute'''

    print("x is greater than 6")
```

The Structure of If-Else Statements



Else Statements

- We now that know we can use if statements to control the flow of our programs.
- What if we wanted an alternative outcome?
- This is where the else statement comes in.
    - e.g. if it is raining, I shall bring a coat, else I shall leave my coat at home.

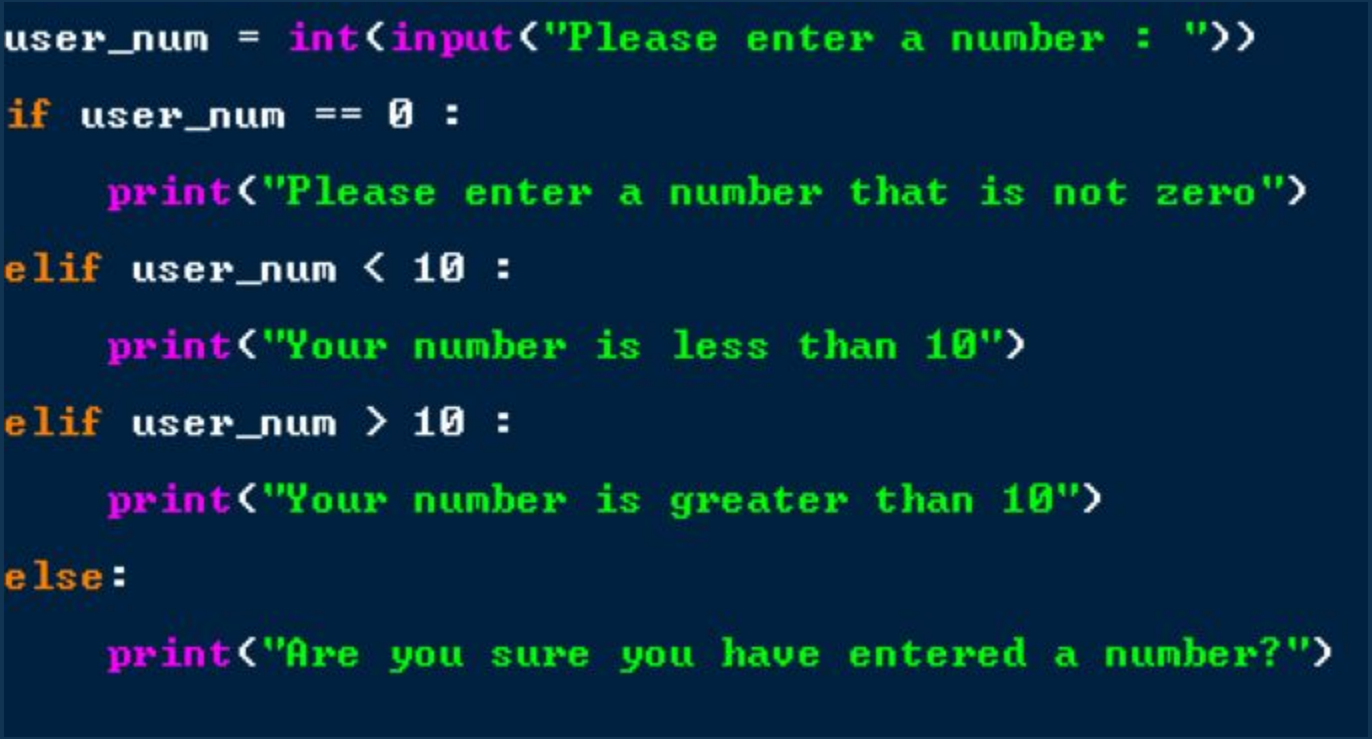


```
is_raining = False
if is_raining == True:
    print("Bring a coat")
else :
    print("Leave coat at home")
```

Elif Statements

- ★ What if there is a situation where we could have multiple statements that are True?
- ★ This is where elif comes into play: Else if → elif
- ★ Elif statements are mainly used to handle the case when multiple True statements are present.
- ★ Note that you can have multiple elif statements in an if-else block.

Elif Statement Example

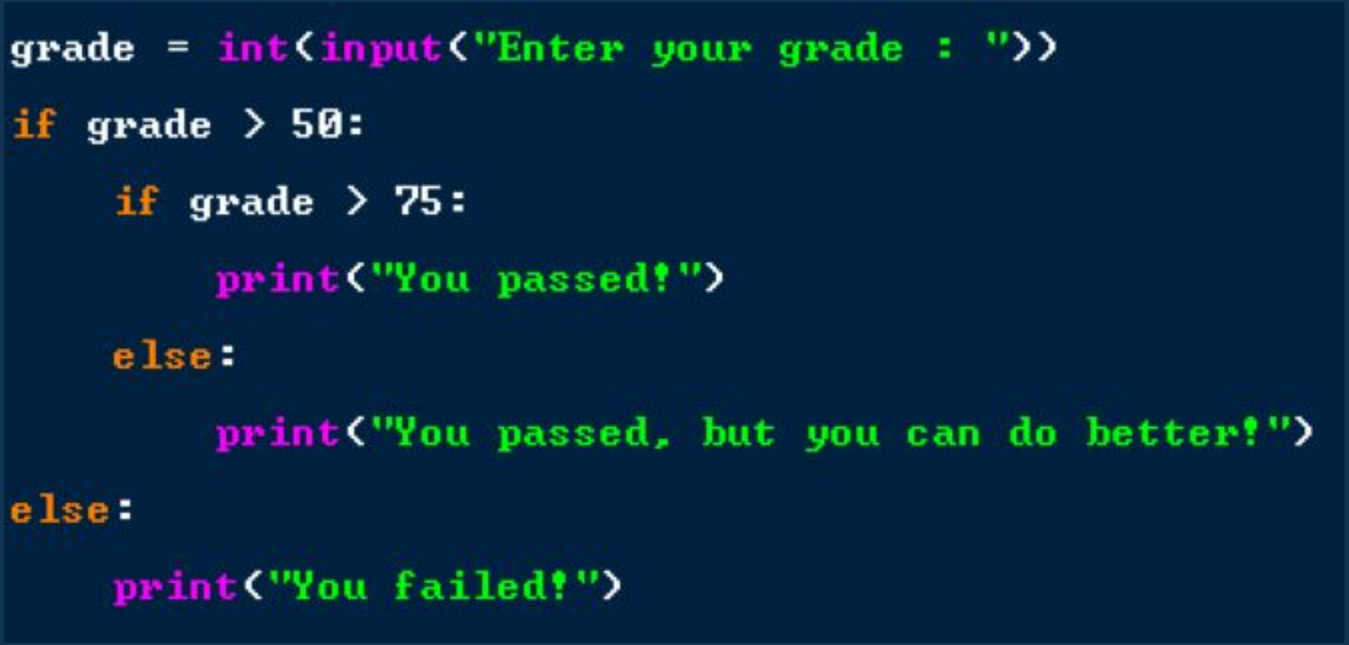


```
user_num = int(input("Please enter a number : "))
if user_num == 0 :
    print("Please enter a number that is not zero")
elif user_num < 10 :
    print("Your number is less than 10")
elif user_num > 10 :
    print("Your number is greater than 10")
else:
    print("Are you sure you have entered a number?")
```

Things to Note

- ★ There is no limit to the number of elif statement one could have in an if-else block.
- ★ Only one final else statement is allowed.
- ★ Each condition is checked in order.
- ★ If one condition is True, that branch executes, and the statement ends.
- ★ Even if there are multiple True conditions, only the first True branch will execute.

Nested If Statements

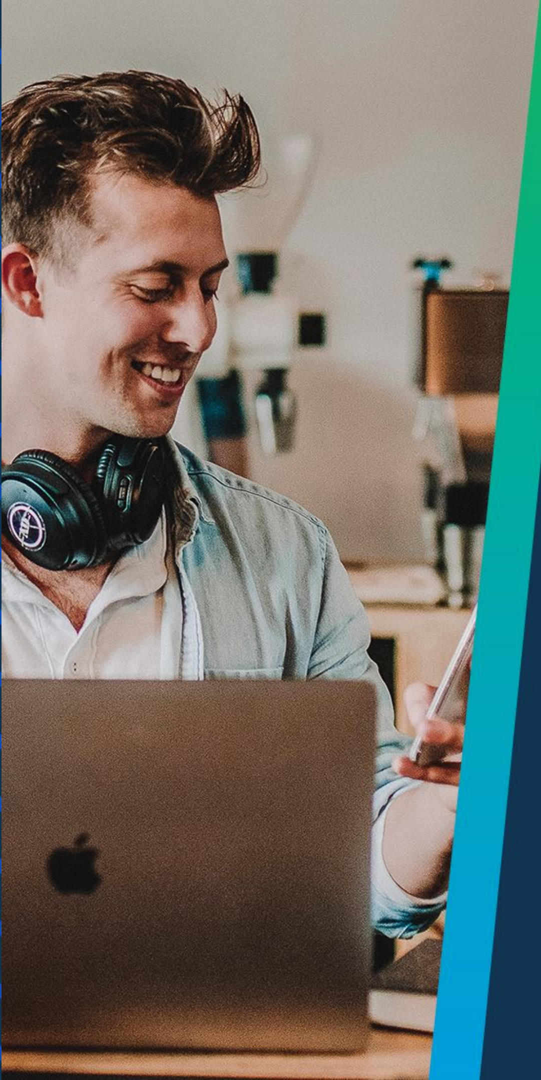


```
grade = int(input("Enter your grade : "))
if grade > 50:
    if grade > 75:
        print("You passed!")
    else:
        print("You passed, but you can do better!")
else:
    print("You failed!")
```

Hyperiondev

Q & A Section

Please use this time to ask any questions relating to the topic, should you have any.



Hyperiondev

Thank you for joining us