CONDITIONAL EXECUTION: IF STATEMENTS

By Rachael Sera

For Junior Knights

CONDITIONAL EXECUTION

So far, our programs do the same operations every time we run them

Conditional execution allows us to perform different actions based on conditions

IF STATEMENT

We use the "if-statement" to execute code *if* a condition is true (and not execute the code if it's not true)

INDENTATION

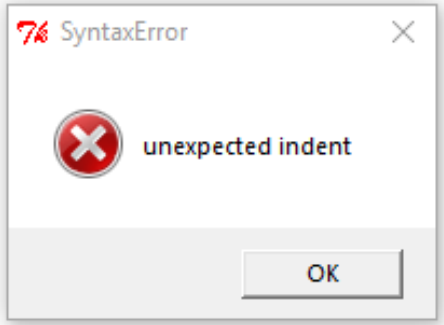
Indentation is used to group lines of code together

- So far, our programs have all been left-justified

Inappropriately aligning lines will produce errors

File Edit Format Run Options Windows Help

```
# This statement is left-justified
print("Hello")
    #This statement is tabbed in and will give an error
    print("oops")
```



INDENTAITON

BOOLEAN EXPRESSIONS

Statements that evaluate to either true of false

<operand> <operator> <operand>

2 + 3

- 2 and 3 are the operands and + is the operator

Operator in if-statements: a relational operator

RELATIONAL OPERATORS

==	equal to
!=	not equal to
>	greater than
>=	greater than or equal to
<	less than
<=	less than or equal to

BOOLEAN EXPRESSION EXAMPLES

Are these expressions true or false?

- 5 == 5
- 5 == 4
- 5 > 4
- 5 < 4
- 5 != 5
- 5 != 4
- 5 >=4
- 5 <= 4
- 5 <= 5
- 4 > 5
- 4 < 5
- 4 <= 5

IF-STATEMENT

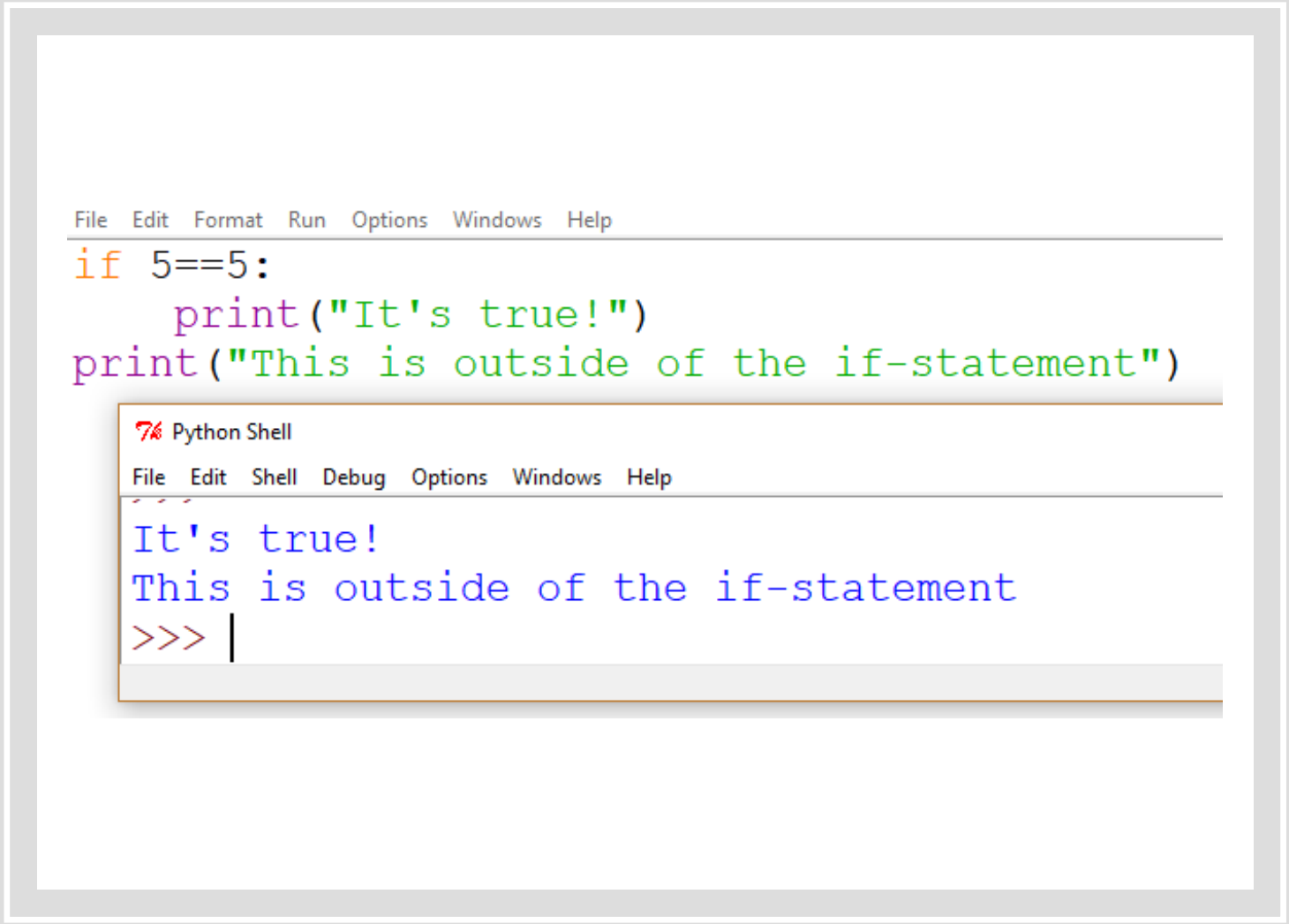
The if-statement will execute code *if* a condition is true (and not execute the code if it's not true)

Indentation is used to group lines of code together

Code that is indented under the if-statement is part of the if-statement and is what will or won't be executed based on whether the condition is true or not

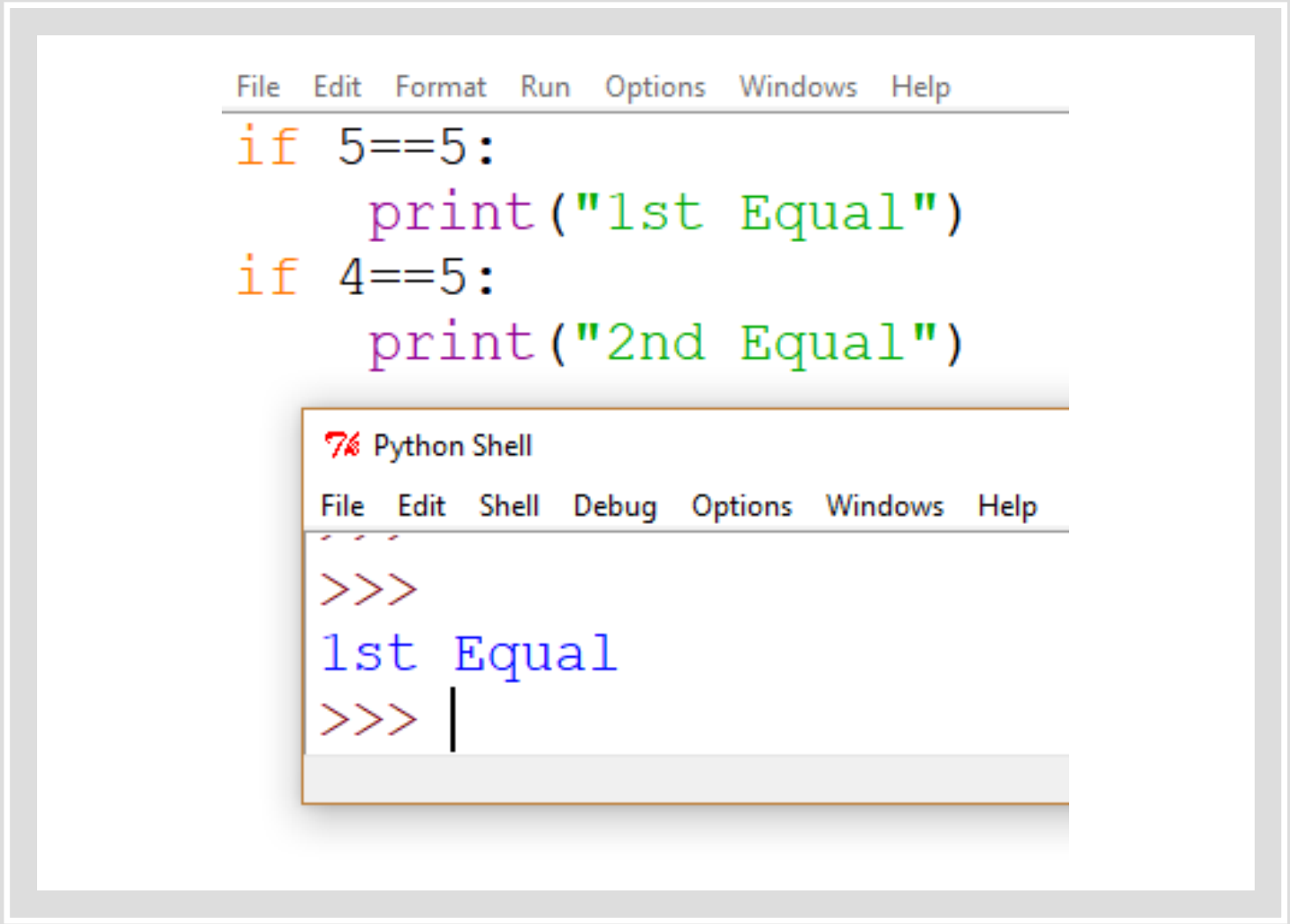
IF-STATEMENT EXAMPLE

- Everything indented under the if-statement will execute only if the condition is true (in this example it is)
- Anything left-justified is outside of the if-statement and will execute regardless, like our programs before
- You can have multiple lines inside the if-statement



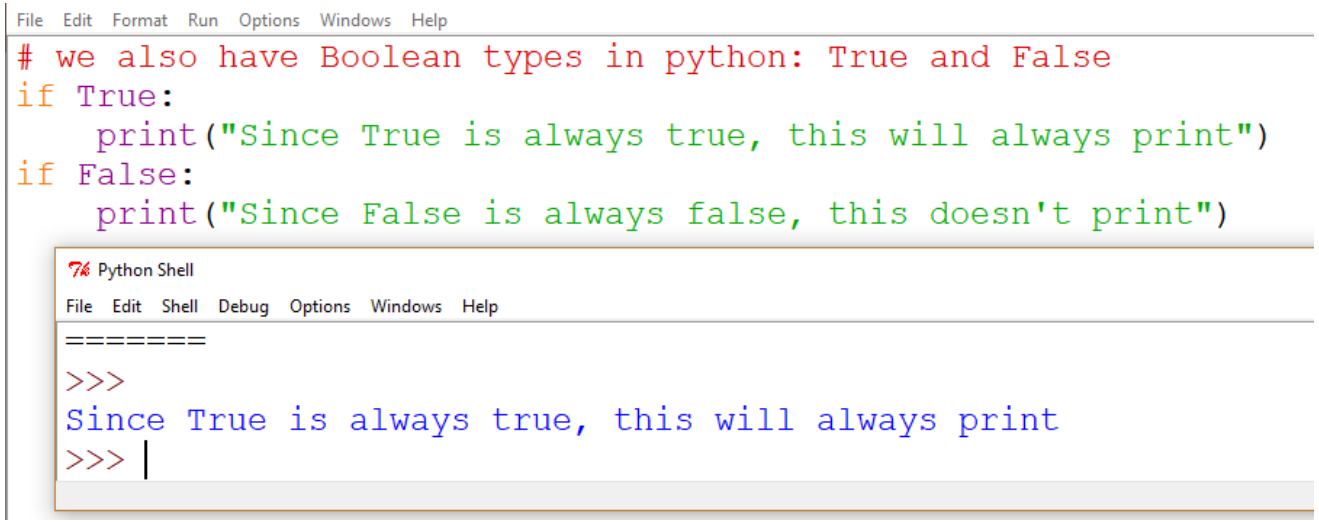
IF-STATEMENT EXAMPLE

- The first statement is true, so "1st Equal" prints
- But the second statement is not true, so "2nd Equal" does not print



BOOLEAN DATA TYPE

- This is to illustrate that if-statements work by evaluating a condition and to introduce the Boolean type



ELSE

- Use else to make something "else" happen when the statement is false

```
File  Edit  Format  Run  Options  Windows  Help
num = int(input("Enter a number: "))
if num < 10:
    print("The number is less than 10")
else:
    print("The number is greater than or equal to 10")
```

```
Python Shell
File  Edit  Shell  Debug  Options  Windows  Help
>>>
Enter a number: 15
The number is greater than or equal to 10
>>>
```

File Edit Format Run Options Windows Help

```
# Celsius or Fahrenheit conversion

startScale = input("Would you like to convert from Fahrenheit or Celsius (Enter F or C)?")
startTemp = float(input("What is the starting temp?"))

if startScale == "C":
    F = 1.8*startTemp+32
    print(F)
else:
    C = (startTemp-32)/1.8
    print(C)
```

TEMPERATURE CONVERSION IF/ELSE EXAMPLE

ELIF

- "Elif" is like "else" but instead of simply catching the opposite of the "if" condition, it can check for other, specific conditions
- The first condition that is true will execute, and the rest will be skipped
- The way this is structured, order matters. If percent >= 90, then we print, and we don't check to see if it's also >= 80.
- If we move one of the other cases up, it will give misleading output.
- Remove the "el" and then multiple statements can print

File Edit Format Run Options Windows Help

```
percent = int(input("What is your percentage in class?\n"))

if percent >= 90:
    print("You got an A!")
elif percent >= 80:
    print("You got a B!")
elif percent >= 70:
    print("You got a C.")
else:
    print("Sorry, you didn't pass.")
```

Python Shell

File Edit Shell Debug Options Windows Help

```
>>>
What is your percentage in class?
84
You got a B!
>>>
```

AND

- Use “and” when multiple conditions are required

File Edit Format Run Options Windows Help

```
GPA = float(input("Enter your GPA: "))
SAT = float(input("Enter your SAT score: "))

if GPA >= 3.4 and SAT >= 1500:
    print("Hoorya! You qualify for the scholarship!")
else:
    print("Sorry, you don't qualify")
```

Python Shell

File Edit Shell Debug Options Windows Help

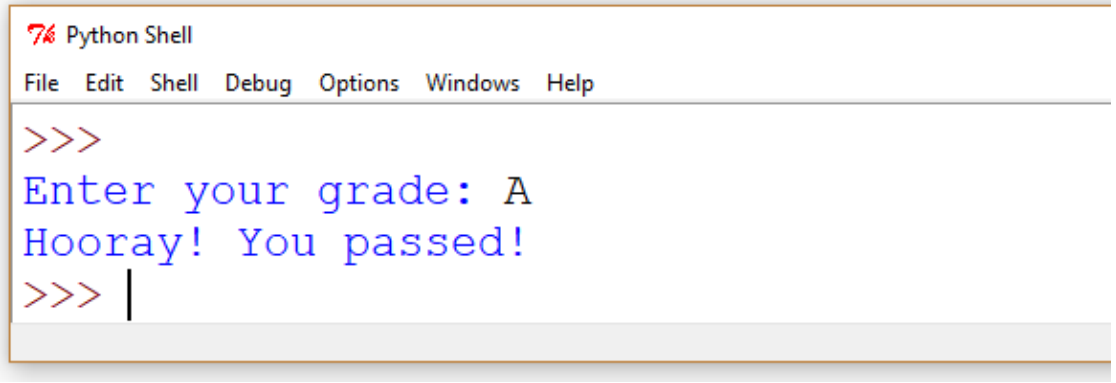
```
>>>
Enter your GPA: 3.5
Enter your SAT score: 1600
Hoorya! You qualify for the scholarship!
>>>
```

OR

Use “or” when only one (at least) of multiple conditions should cause the code to execute

File Edit Format Run Options Windows Help

```
grade = input("Enter your grade: ")
if grade == "A" or grade == "B" or grade == "C":
    print("Hooray! You passed!")
else:
    print("Boo, you didn't pass")
```



MATERIALS

http://www.eecs.ucf.edu/JuniorKnights/material/

- Book for additional reading
- Meeting dates

github.com/rsera/junior-knights

- These slides are
  - *Python Lesson 4 If Statement.pptx*
- The problems for today are: goo.gl/aUipex