FinTech @ IU Python Session #2 – Basics

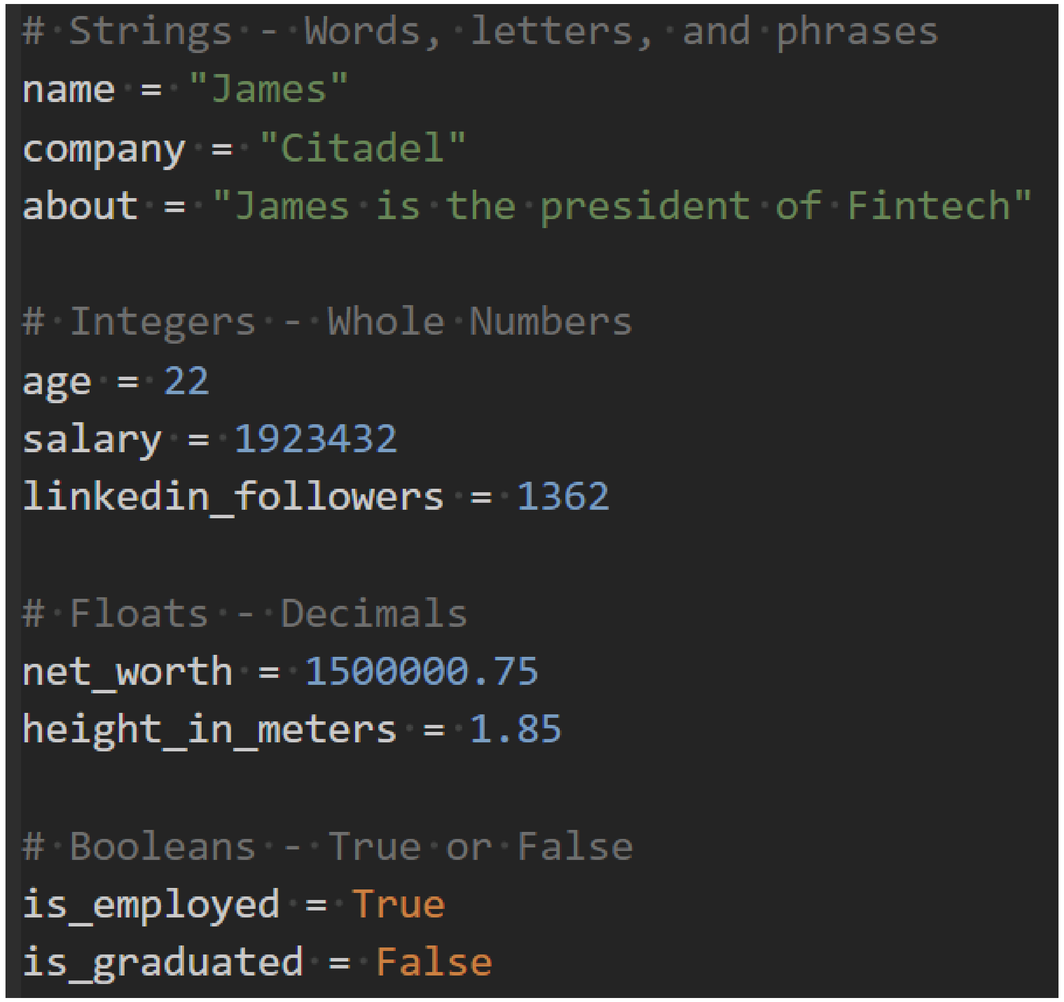
Prepared by

Gabriel Shores
Director of Technology – FinTech @ IU
https://www.linkedin.com/in/gabriel-
shores-379b81291/

09/27/2024

Variables

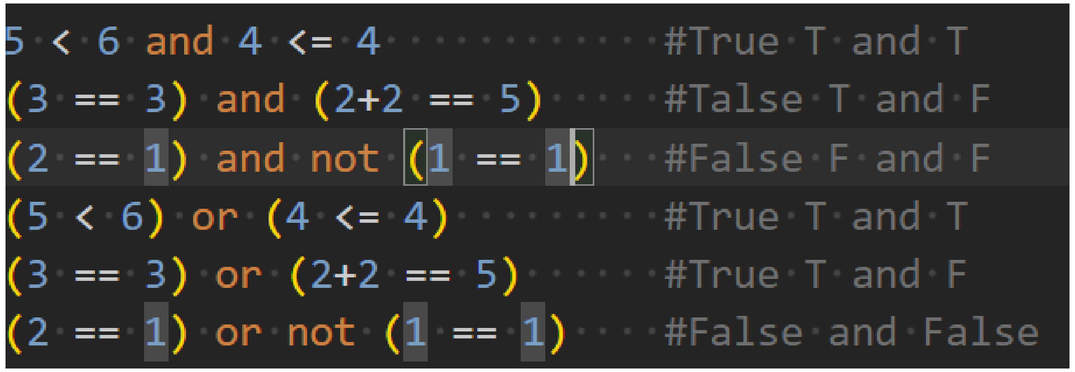
- Variables have two parts: their name and value
- Variables can be assigned and declared as such:
- Variables can be reassigned new values
- Python convention is to name_variables_like_this
- Four main types right now: Strings, Integers, Floats, Booleans
- Let’s dive in some more!



```python
# Strings - Words, letters, and phrases
name = "James"
company = "Citadel"
about = "James is the president of Fintech"

# Integers - Whole Numbers
age = 22
salary = 1923432
linkedin_followers = 1362

# Floats - Decimals
net_worth = 1500000.75
height_in_meters = 1.85

# Booleans - True or False
is_employed = True
is_graduated = False
```

More on Booleans/Conditionals

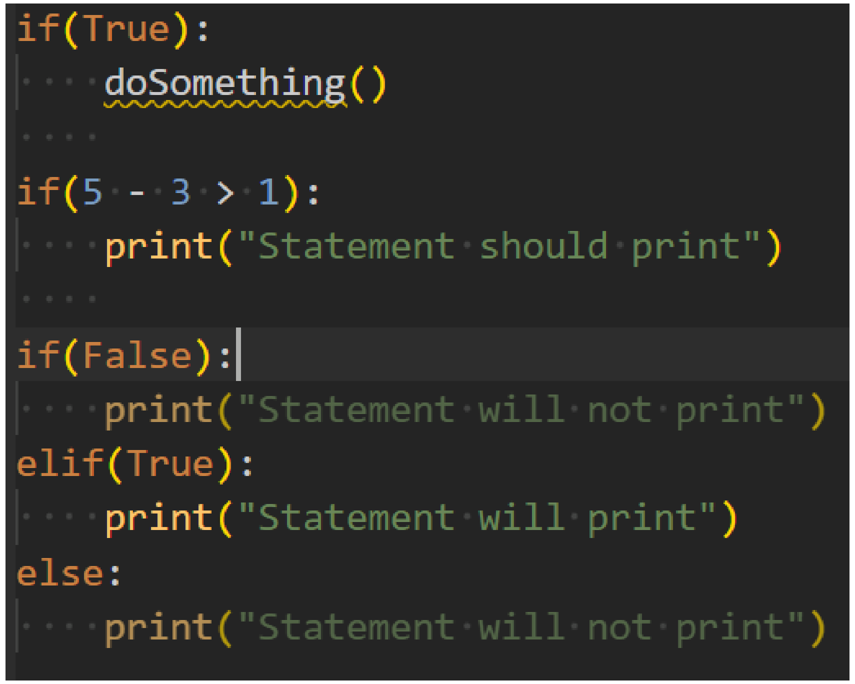
- Statements using >,>=,<,<=,==,evaluate to Boolean values, True or False
- Chaining multiple statements with and / or creates more complex logic
- Using not negates a statement: not True = False, not False = True



```
5 < 6 and 4 <= 4                #True T and T
(3 == 3) and (2+2 == 5)         #Talse T and F
(2 == 1) and not (1 == 1)       #False F and F
(5 < 6) or (4 <= 4)             #True T and T
(3 == 3) or (2+2 == 5)          #True T and F
(2 == 1) or not (1 == 1)        #False and False
```

More on Booleans/Conditionals

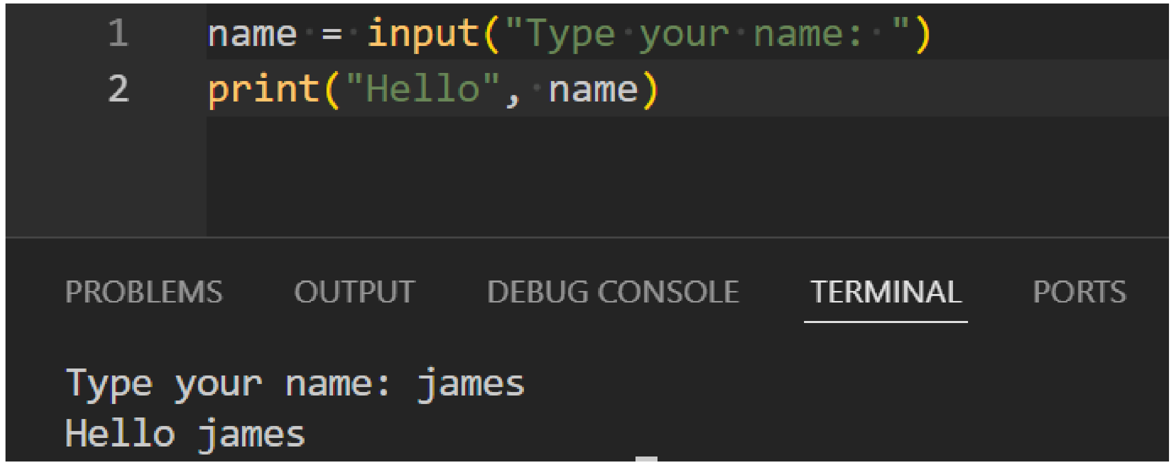
- If statements evaluate a block of code if the statement is true
- elif executes only if the previous statements were false, and its statement is true
- else executes if all previous statements evaluated to false
- Following logic needs to be indented



```
if(True):
    doSomething()
    
if(5 - 3 > 1):
    print("Statement should print")
    
if(False):
    print("Statement will not print")
elif(True):
    print("Statement will print")
else:
    print("Statement will not print")
```

Input!

- Much like with print(), we can use input("string")
- Input will print out the given string and then take in responses via the terminal
- Let's try it all out!

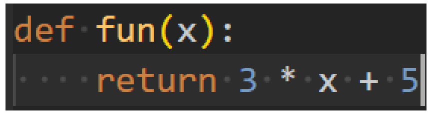


```python
name = input("Type your name: ")
print("Hello", name)
```

```
Type your name: james
Hello james
```

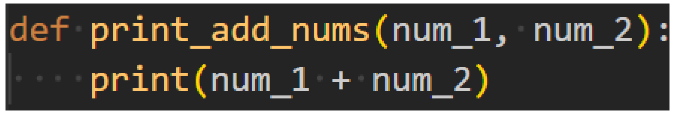
Functions!

- Think, back to math, functions can take in inputs (parameters) and return an output
- Let’s say we want to model f(x) = 3x+5 in code →



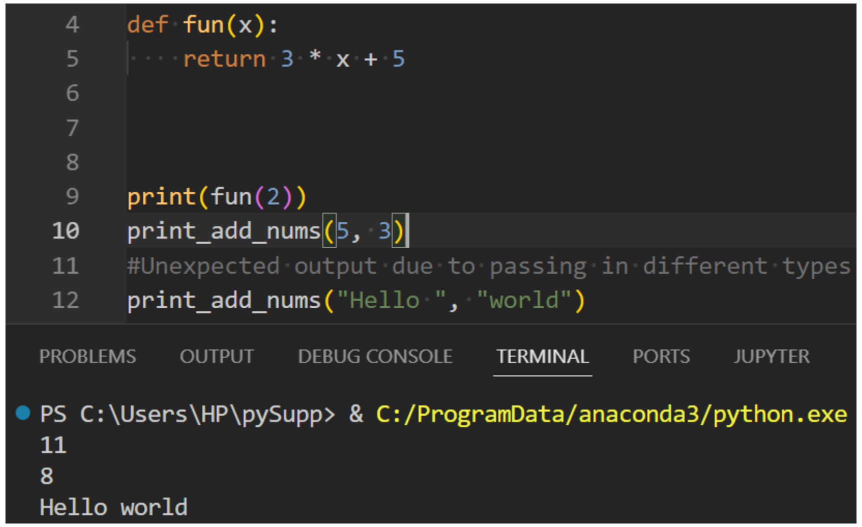
```python
def fun(x):
    return 3 * x + 5
```

- Functions also can have no variables or no return →



```python
def print_add_nums(num_1, num_2):
    print(num_1 + num_2)
```

- Output →



```python
def fun(x):
    return 3 * x + 5



print(fun(2))
print_add_nums(5, 3)
#Unexpected output due to passing in different types
print_add_nums("Hello ", "world")
```

```
PS C:\Users\HP\pySupp> & C:/ProgramData/anaconda3/python.exe
11
8
Hello world
```

The End!

- Next session we should be finishing up with Python basics, allowing us to dive into more workshop-styled sessions
- Sites like EdaBit allow you to practice the basics of Python through solving problems
- Other good resources include FreeCodeCamp and Codecademy
- Feel free to reach out if you ever need any help
- See you next time!