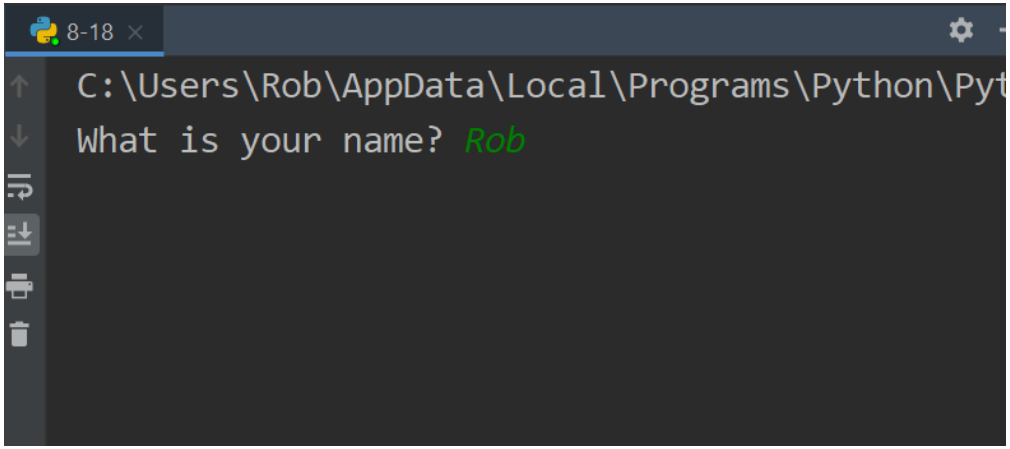
ITP 115
Programming in Python

Input

Input

- We can ask the user to type in information which we can store using the **`input(…)`** function
  - *"Getting input from the user"*
- Also called a **prompt**



Getting User Input

- Syntax

```
var = input(message)
```

- `message`
  - String that is displayed to the user
  - Should describe what they need to type
- `var`
  - Variable that contains what the user typed

Getting User Input

- Syntax

```
var = input(message)
```

- Examples

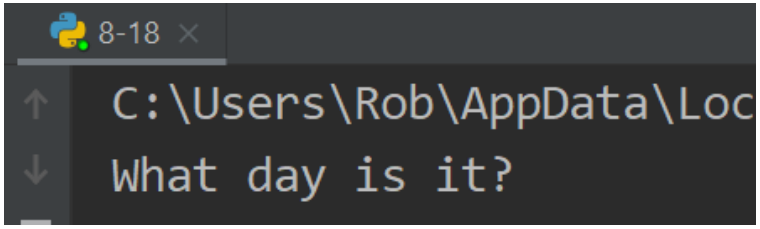
```
day = input("What day is it? ")

prompt = "What is your name? "
name = input(prompt)

print(input("What is your favorite color? "))
```

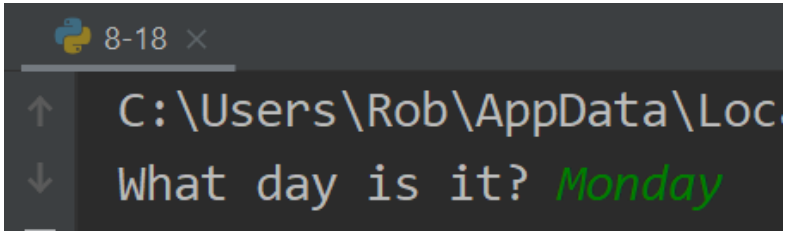
Example

```
day = input("What day is it? ")
```



day

- *Program pauses running until the user types something*
- *Once user presses enter, program continues*



day Monday

User types "Monday" and hits enter

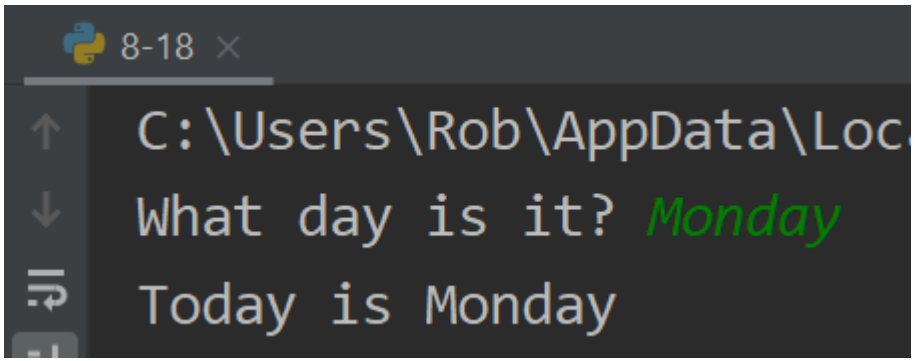
Example

```
day = input("What day is it? ")
```

- *Input function* ***returned*** *the value from user*
- *Value is* ***stored / assigned to variable***

day Monday

```
day = input("What day is it? ")
print("Today is", day)
```



Reading in Numbers

- Enter the following code

```
num1 = input("Please enter a number: ")
num2 = input("Please enter another number: ")
print(num1 + num2)
```

```
Please enter a number: 3
Please enter another number: 3
33
```

What is the output?

Reading in Numbers

- input() **always returns a string**
- **+** combines two strings together
- **Solution**
  - When reading in numbers, you need to **convert** the
    `string` → **`int`**
    or
    `string` → **`float`**

Reading in Numbers

- Enter the following code

```
num1 = int(input("Please enter a number: "))
num2 = int(input("Please enter another number: "))
print(num1 + num2)
```

```
Please enter a number: 3
Please enter another number: 3
6
```

What is the output?

Conversion Functions

Function	Description	Example	Returns
`float(x)`	Returns a floating-point value by converting x	`float("10.0")`	`10.0`
`int(x)`	Returns an integer value by converting x	`int("10")`	`10`
`str(x)`	Returns a string value by converting x	`str(10)`	`"10"`

Labs

- We suggest putting your labs in a folder named Labs under a folder named ITP115 somewhere on your computer.
- Use Finder (Mac) or File Explorer (Windows) to create folders if you have not already done so.

PyCharm – New File

- To create a new Python file, select **File → New...** from the main menu.
- In the little window, select the **Python File** option (not File).
- In the New Python File window, enter a name such as **ITP115_L1_1_*LastName_FirstName*** and press the return key.
  - Replace *LastName* with your last/family name and *FirstName* with your first name.

How to Compress / Submit Labs

- Create a zip file containing your Python code.
  - This cannot be done within PyCharm.
  - Find the file or folder on your computer and compress it.
  - Use File Explorer (Windows) or Finder (Mac).
  - Right click (two touch) and select the compress option.
- Upload the zip file to your Blackboard section:
  1. On Blackboard, click on the **Labs** item in the course menu on the left.
  2. Click on the specific item for this lab (Lab1-1).
  3. Click on the **Browse My Computer** button and select your zip file.
  4. Click the **Submit** button.

How to Compress / Submit Assignments

- We will go over this, but for reference
- Create a folder on your computer called **ITP115_A#_*LastName_FirstName*** where # is the assignment number and *LastName* is your last/family and and *FirstName* is your first name.
- Inside the folder, include your python source code
- Compress / zip the folder to create **ITP115_A#_*LastName_FirstName*.zip**
- Upload zip file to Blackboard