CSCI-UA-0002

Intro to Computer Programming (No Prior Experience)

Module 2: Types, Operators, Debugging

Professor Emily Zhao

Section 008	Section 012
T/R 12:30-1:45PM	T/R 4:55-6:10PM

Agenda

- Poll Everywhere Review Quiz
- Review Interactive vs Script mode
- Review Ed Questions
- Module 2 Review
- Assignment #1 Workshop

Module 2

- Commenting Your Code
- Data Types
- Data Type Conversion
- Math Operators + Mixed Type Expressions
- Error + Error Types
- Formatting Strings + Numbers
- Drawing Graphics in Python (Turtle)

→ *What is 'pseudocode'?*

→ *The difference between* `int` *and* `float`*?*

→ *What is the* `type()` *function?*

→ *How do nested functions work?*

→ *How does* `//` *round?*

→ *Can you give examples of using* `%`*?*

→ *What are formatting patterns?*

→ *How do you import a library?*

Commenting Your Code

Documentation
explaining how it works, what it does, and why certain decisions were made

Readability
organization, easier to understand, helps other developers understand your code

Debugging
remove lines of code without deleting them

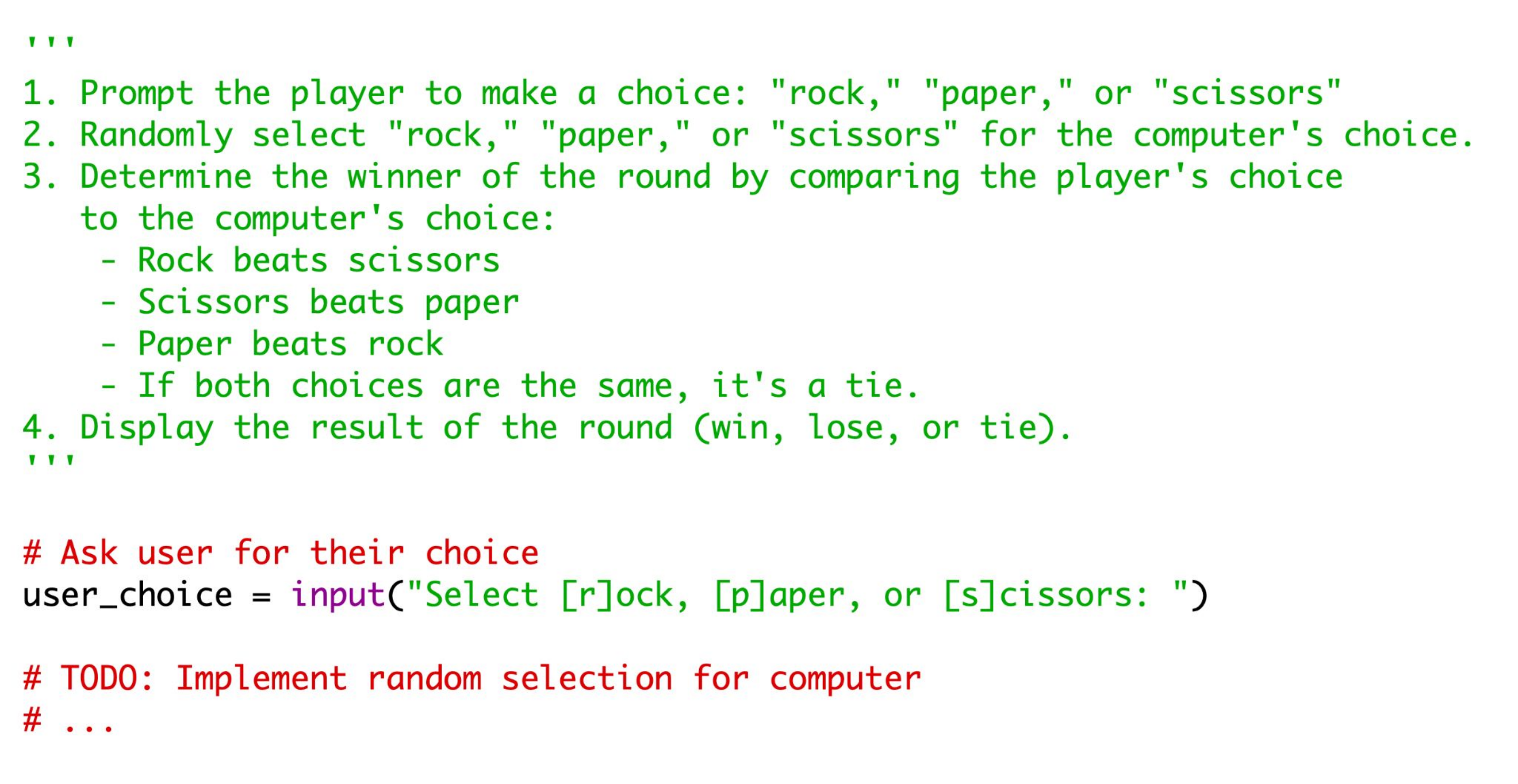
- → **Faster problem solving**
- → **Collaboration**
- → **Learning**
- → **Onboarding**
- → **Avoiding Misinterpretation**
- → **Future Planning**

Pseudocode

- allows programmers to plan and outline the steps of an algorithm or program in plain language
- not meant to be executed by a computer
- serves as an intermediate step between designing a program and writing the actual code in a programming language

```
Rock Paper Scissors Pseudocode

 1.   Prompt the player to make a choice:
      "rock," "paper," or "scissors"
 2.   Randomly select "rock," "paper," or
      "scissors" for the computer's choice.
 3.   Determine the winner of the round.
      Compare the player's choice to the
      computer's choice:
      - Rock beats scissors
      - Scissors beats paper
      - Paper beats rock
      - If both choices are the same, it's a
      tie.
 4.   Display the result of the round (win,
      lose, or tie).
```

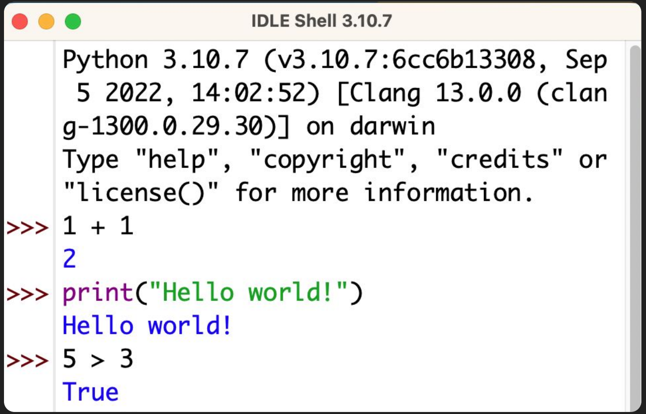
Rock, Paper, Scissors



```python
'''
1. Prompt the player to make a choice: "rock," "paper," or "scissors"
2. Randomly select "rock," "paper," or "scissors" for the computer's choice.
3. Determine the winner of the round by comparing the player's choice
   to the computer's choice:
    - Rock beats scissors
    - Scissors beats paper
    - Paper beats rock
    - If both choices are the same, it's a tie.
4. Display the result of the round (win, lose, or tie).
'''


# Ask user for their choice
user_choice = input("Select [r]ock, [p]aper, or [s]cissors: ")

# TODO: Implement random selection for computer
# ...
```

Interactive Mode

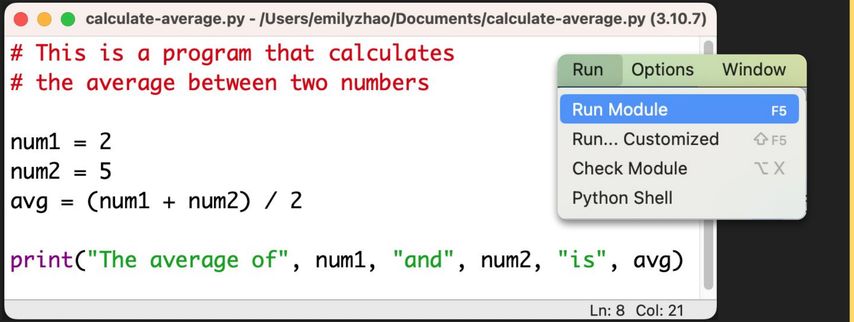
- Commands are typed directly in the Shell (the window that opens every time when you launch IDLE)
- Not meant to write multiple lines of code



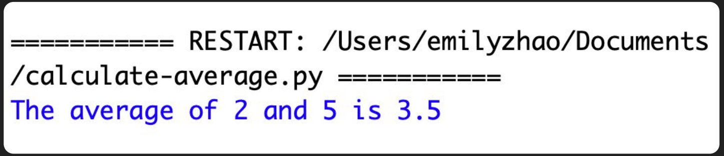
```
Python 3.10.7 (v3.10.7:6cc6b13308, Sep
 5 2022, 14:02:52) [Clang 13.0.0 (clan
g-1300.0.29.30)] on darwin
Type "help", "copyright", "credits" or
"license()" for more information.
>>> 1 + 1
    2
>>> print("Hello world!")
    Hello world!
>>> 5 > 3
    True
```

Script Mode

- Create a new script (File > New File)
- Meant for writing long programs
- Save file before running
- Output shows up in the Shell



```
# This is a program that calculates
# the average between two numbers

num1 = 2
num2 = 5
avg = (num1 + num2) / 2

print("The average of", num1, "and", num2, "is", avg)
```



```
=========== RESTART: /Users/emilyzhao/Documents
/calculate-average.py ===========
The average of 2 and 5 is 3.5
```

Data Types

Data Types

String Literals (character-based data):

```
greeting = "Hello, World!"
```

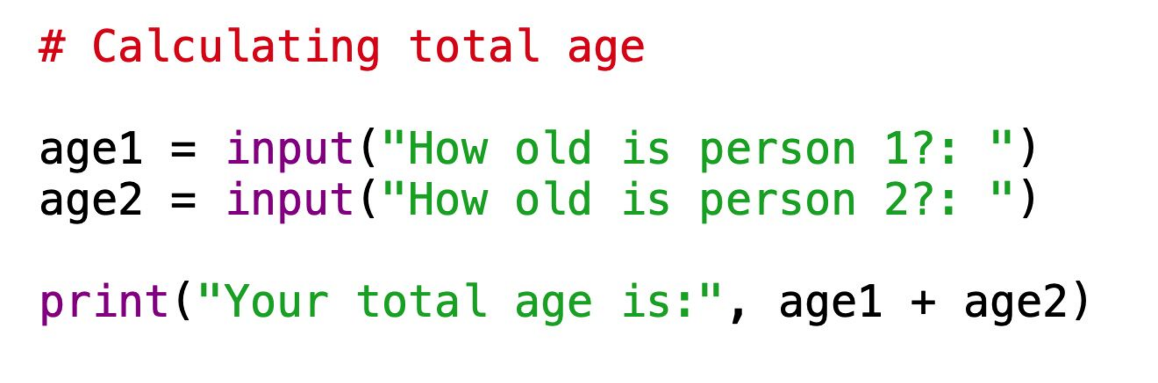
Numeric Literals:

```
num = 5
pi = 3.1415
```

Logical Values (booleans):

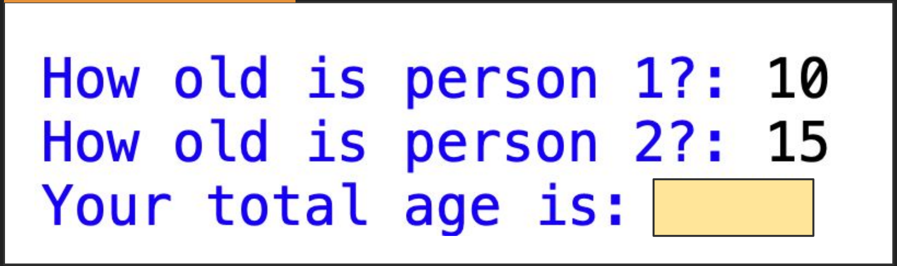
```
isThursday = True
```

Source:



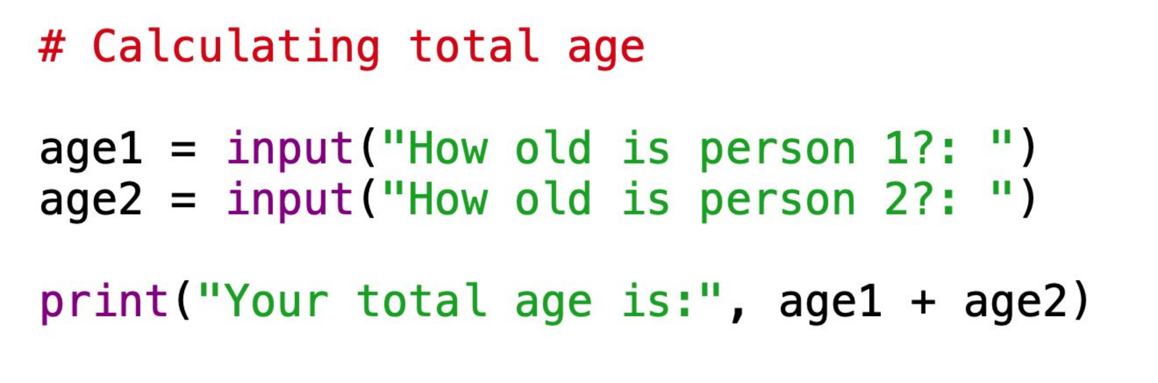
```python
# Calculating total age

age1 = input("How old is person 1?: ")
age2 = input("How old is person 2?: ")

print("Your total age is:", age1 + age2)
```

Execution:



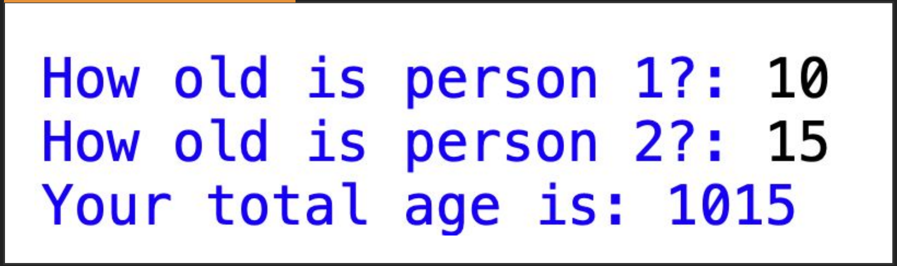
```
How old is person 1?: 10
How old is person 2?: 15
Your total age is: 
```

Source:



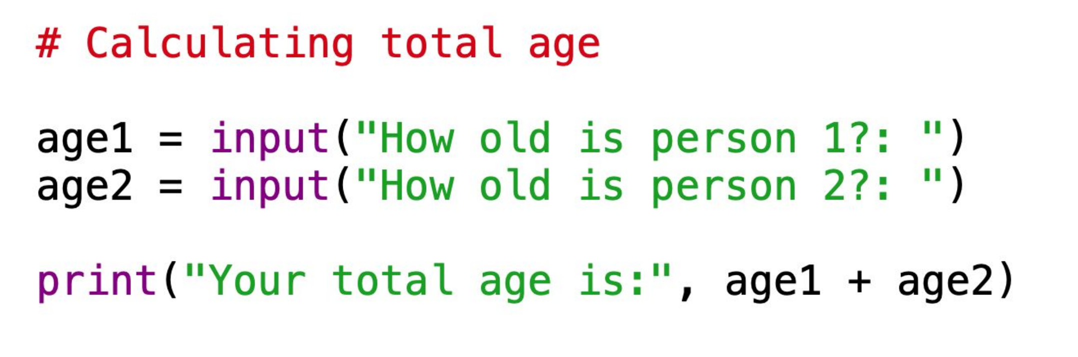
```python
# Calculating total age

age1 = input("How old is person 1?: ")
age2 = input("How old is person 2?: ")

print("Your total age is:", age1 + age2)
```

Execution:



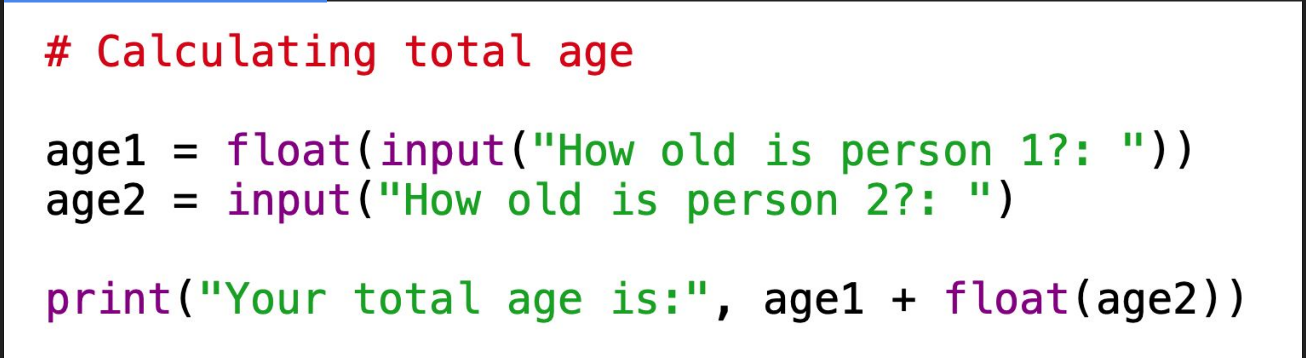
```
How old is person 1?: 10
How old is person 2?: 15
Your total age is: 1015
```

Old Source:



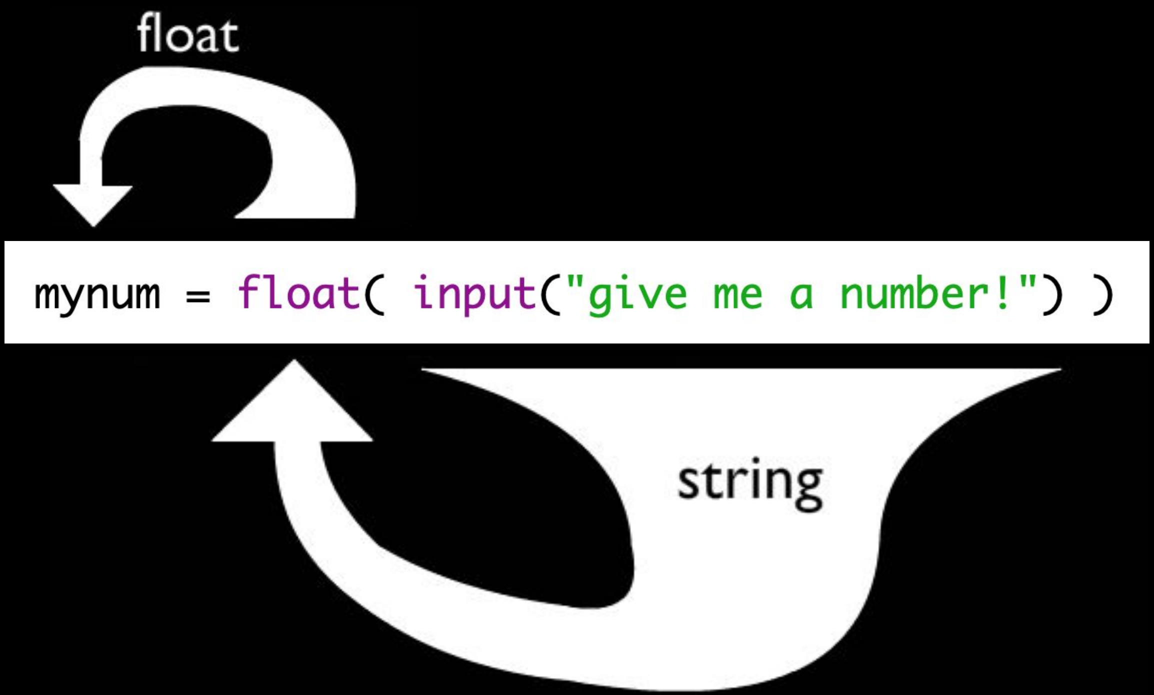
```python
# Calculating total age

age1 = input("How old is person 1?: ")
age2 = input("How old is person 2?: ")

print("Your total age is:", age1 + age2)
```

New Source:



```python
# Calculating total age

age1 = float(input("How old is person 1?: "))
age2 = input("How old is person 2?: ")

print("Your total age is:", age1 + float(age2))
```

Nesting data type conversions



```
mynum = float( input("give me a number!") )
```

Conversion functions

- **To String**
  `str()`
- **To Float**
  `float()`
- **To Integer**
  `int()`

What’s the difference between / and //?

/	floating point division
//	integer division
\	escape character

What’s the output?

`2 + 3` → **5**

`2 + 3.0` → **5.0**

`2.0 + 3.0` → **5.0**

`5 / 2` → **2.5**

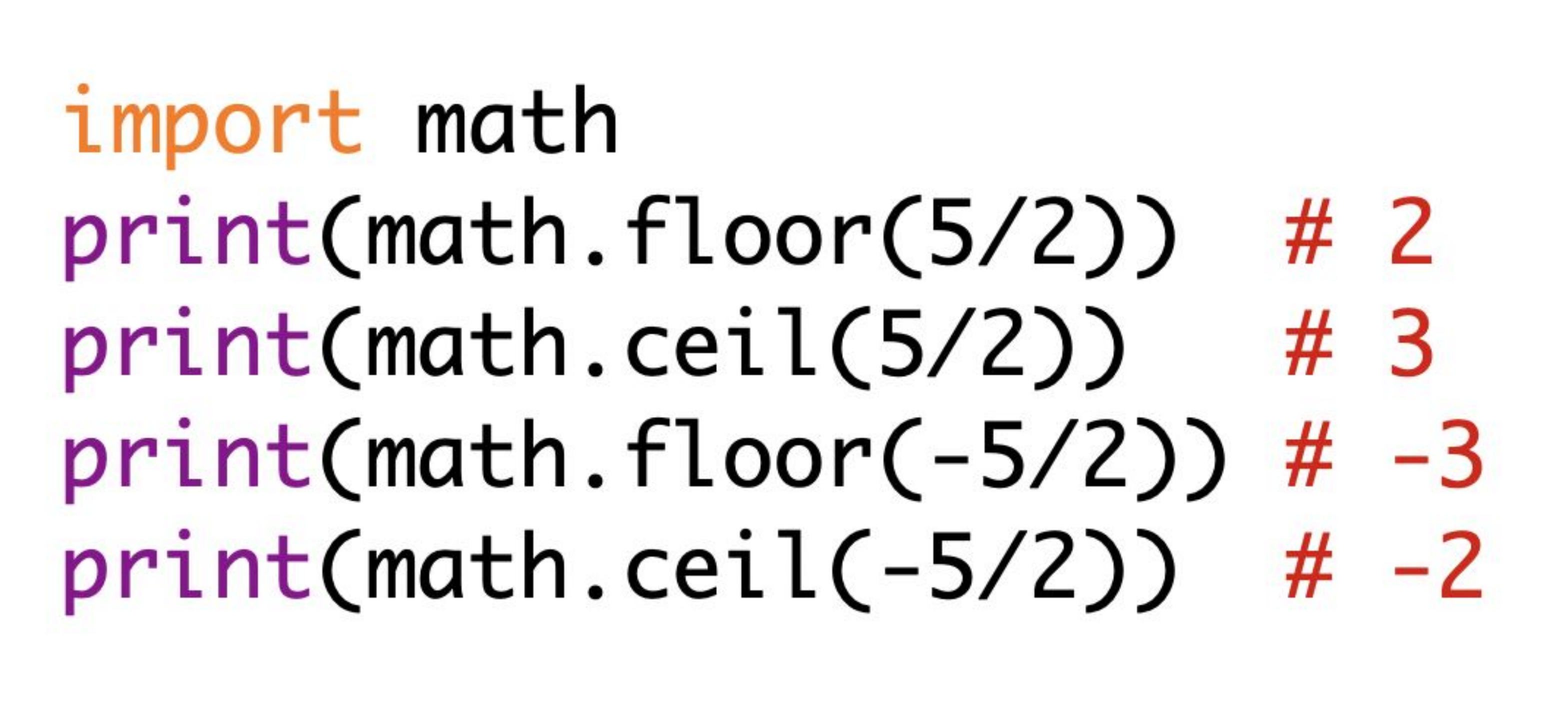
`6 / 2` → **3.0**

`5 // 2` → **2**

`- 5 // 2` → **-3**

`6 // 2` → **3**

`6.0 // 2.0` → **3.0**



```
import math
print(math.floor(5/2))  # 2
print(math.ceil(5/2))   # 3
print(math.floor(-5/2)) # -3
print(math.ceil(-5/2))  # -2
```

Programming Challenge: Days, Hours, Minutes

Write a Python program that takes an input integer representing a total number of minutes and calculates and prints the equivalent number of days, hours, and remaining minutes.

```
Enter minutes: 245678

>> 170 days, 14 hours, and 38 minutes
```

Formatting Strings + Numbers

The `format` function

```
format(value, format_spec="")
```

* only takes in one value
* always returns a string

The formatting spec

```
">10.2f"
```

Align modifier

`>`	right-aligned
`<`	left-aligned
`^`	center-aligned

Width modifier

<u>TOTAL</u> width of formatted string

Precision modifier

`.2`	round to 2 decimal places

Type modifier

`f`	float
`d`	integer
`s`	string

```
PI = 3.1415926
format(PI, ">10.2f")

>>       3.14
   ----------
```

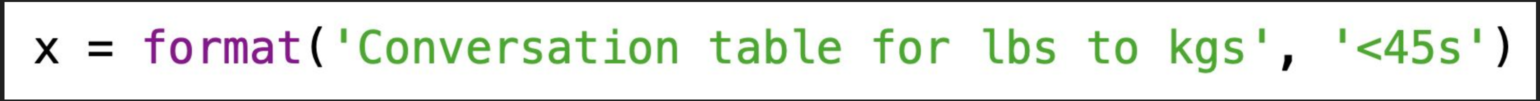
The `format` function

Input	Format Spec	Output	Description
3.1415926	".2f"		
3141.5926	",.2f"		
0.52	"%"		
0.52	".0%"		
11	">10d"		
11	"<10d"		
11	"^10d"		
11	"0>10d"		
39.3947234	">10.3f"		

The `format` function

Input	Format Spec	Output	Description
`3.1415926`	`".2f"`	`3.14`	2 decimal places
`3141.5926`	`",.2f"`	`3,141.59`	2 decimal places with comma
`0.52`	`"%"`	`52.000000%`	Convert to percentage
`0.52`	`".0%"`	`52%`	No decimal places with %
`11`	`">10d"`	`        11`	Right aligned integer, length: 10
`11`	`"<10d"`	`11        `	Left aligned integer, length: 10
`11`	`"^10d"`	`    11    `	Center aligned integer, length: 10
`11`	`"0>10d"`	`0000000011`	Padded with zeros on the left
`39.3947234`	`">10.3f"`	`    39.394`	Right aligned, rounded 3, 10 wide

The `format` function

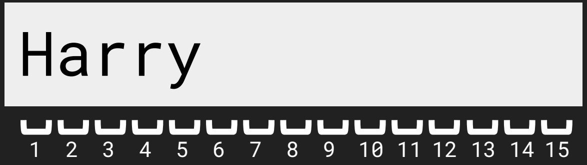


```
x = format('Conversation table for lbs to kgs', '<45s')
```

- * Make sure that your padding length (45) is actually longer than the length of your string.
- You can calculate the length of strings by using `len("string")`.

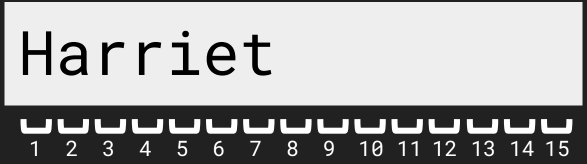
The `format` function

```
print(format('Harry', '<15s'))
```



```
Harry
```

```
print(format(Harriet, '<15s'))
```



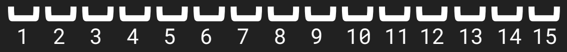
```
Harriet
```

“Harry” and “Harriet” are now both 15 character spaces long, justified to the left.

The `format` function

```
print(format('Apple', '<15s'), end="")
print(0.75)
print(format('Banana', '<15s'), end="")
print(0.25)
```

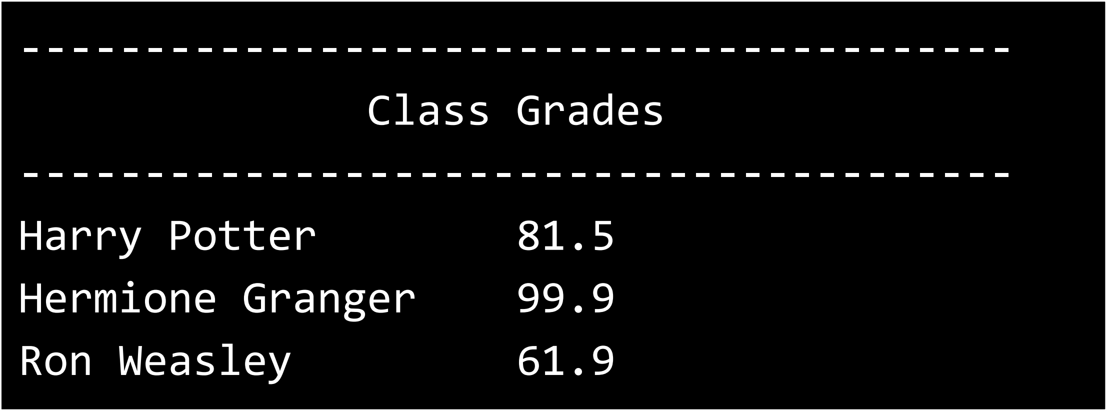
```
Apple          0.75
Banana         0.25
```



Setting “Apple” and “Banana” both to 15 characters long to create a two column layout

Programming Challenge: Formatting a Table

Reproduce the following table using format (40 spaces wide)



```
----------------------------------------
              Class Grades
----------------------------------------
Harry Potter        81.5
Hermione Granger    99.9
Ron Weasley         61.9
```

Homework

- Assignment #1 (due tonight @11:59PM)
- Self-Paced Learning Module #3 (due next class)
- Quiz #3 (due next class by 12:30PM)
- Ask a question on Ed