Module 02

Types, Operators, Debugging

Emily Zhao
CSCI–UA.0002

Agenda

- Tutoring Resources
- Review Module 2 / Go over Quiz 2
- Practice Problems

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)

Data Types

Data Types

String Literals (character-based data):

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

Numeric Literals:

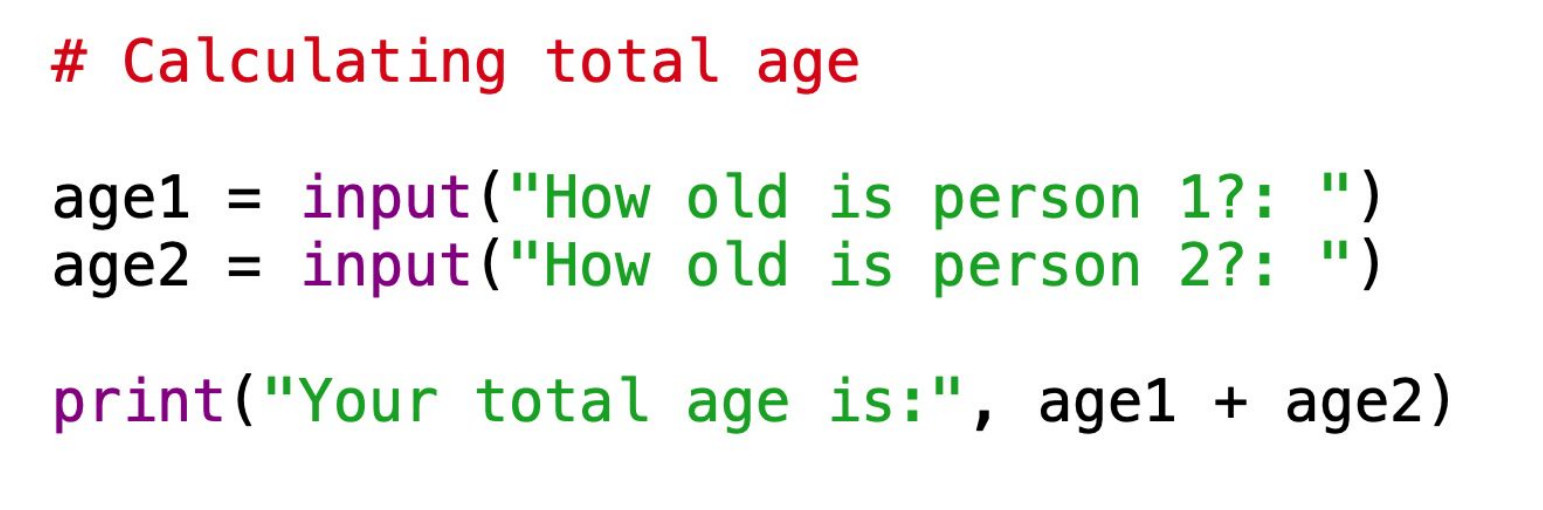
```
num = 5
pi = 3.1415
```

Logical Values:

```
isThursday = True
```

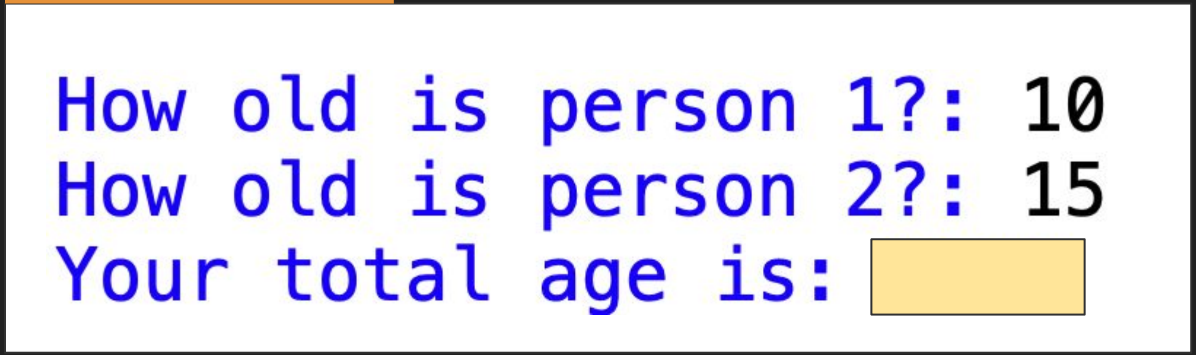
Data Type Conversions

Source:



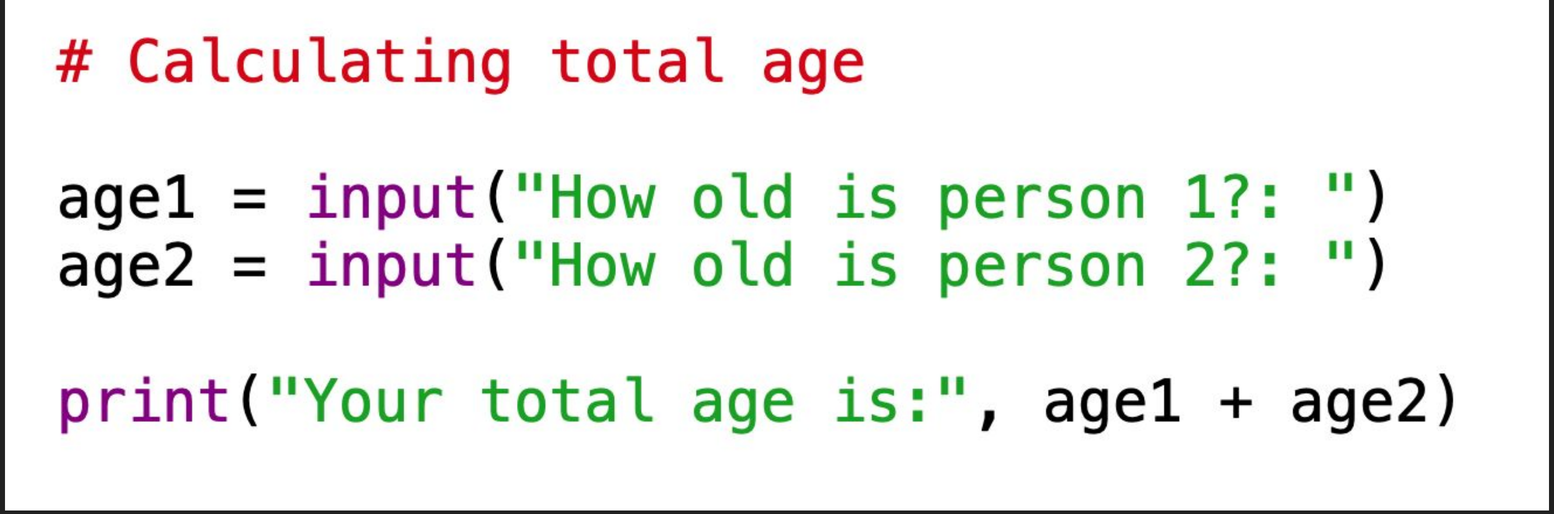
```python
# Calculating total age

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

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

Execution:



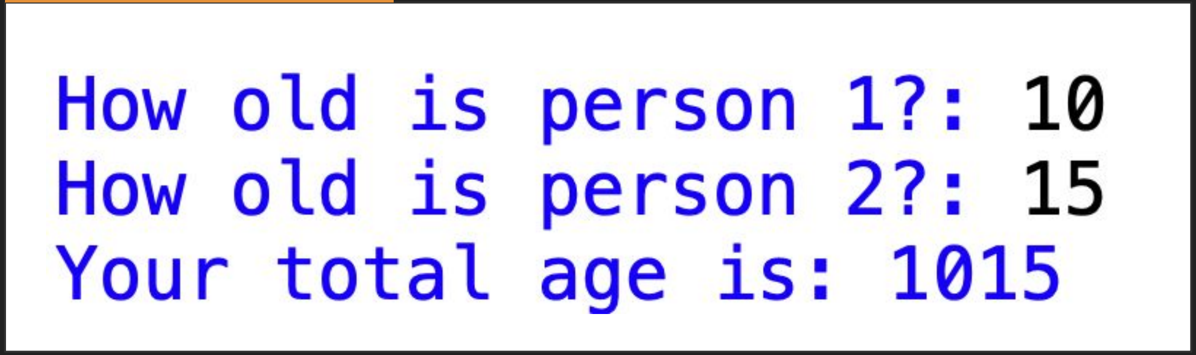
```
How old is person 1?: 10
How old is person 2?: 15
Your total age is:
```

Source:



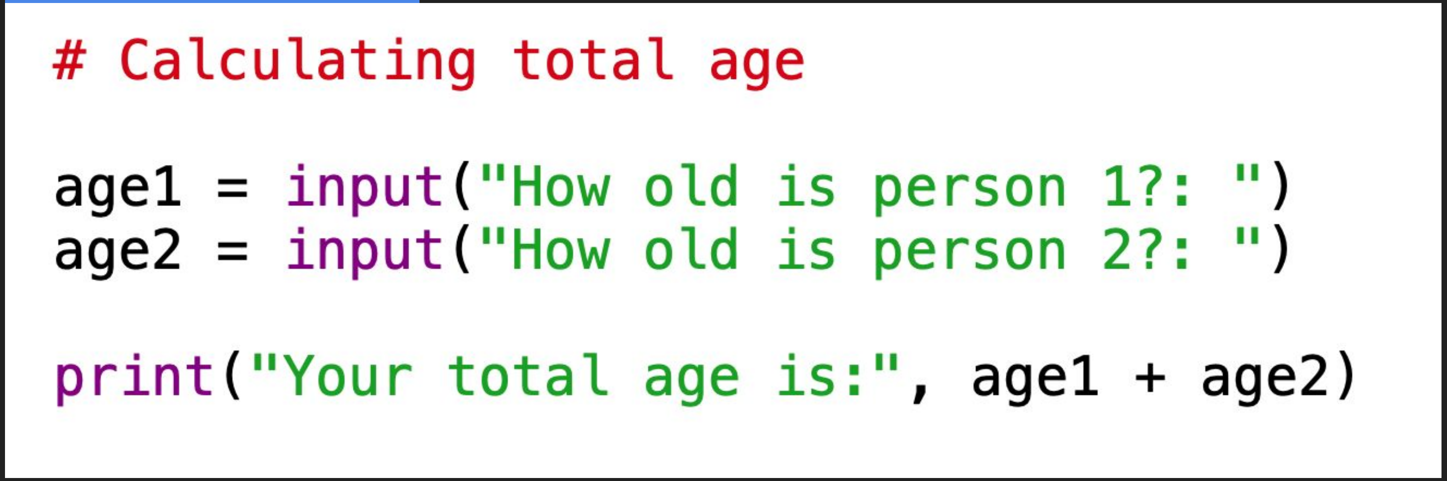
```
# Calculating total age

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

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

Execution:



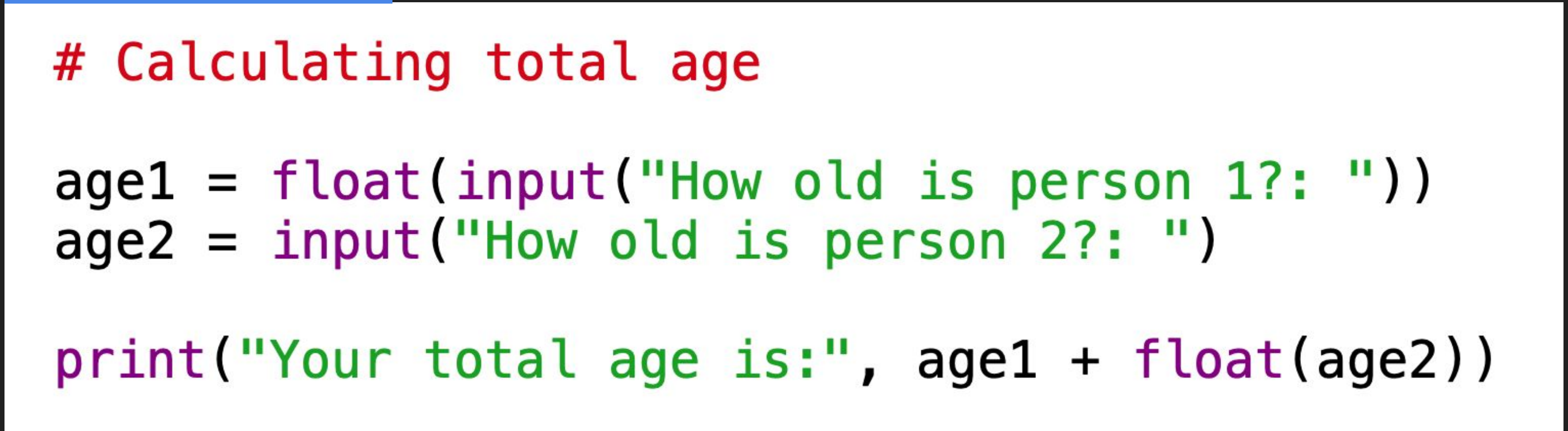
```
How old is person 1?: 10
How old is person 2?: 15
Your total age is: 1015
```

Old Source:



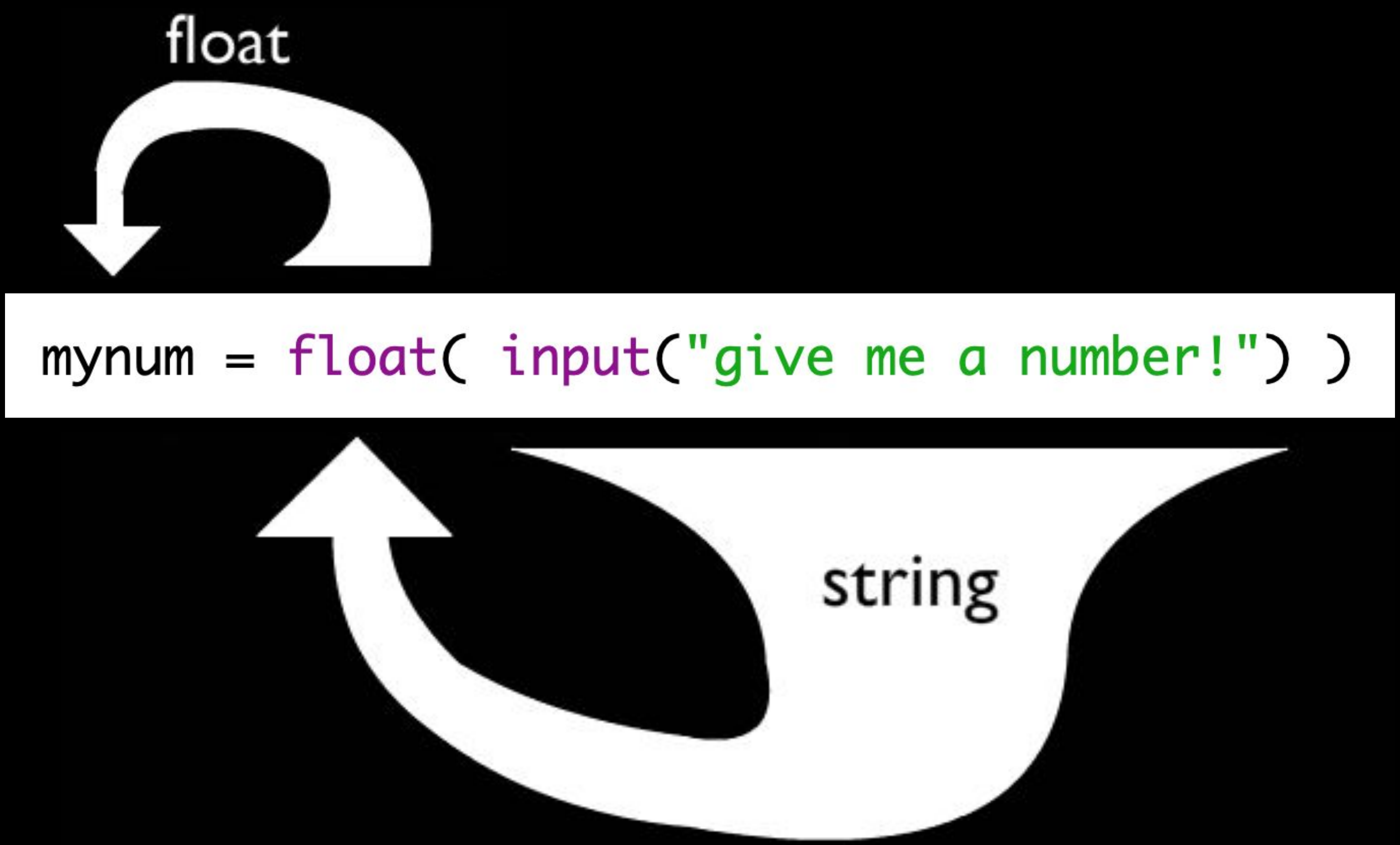
```
# Calculating total age

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

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

New Source:



```
# 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()`

Math Operators + Mixed Type Expressions

What’s the difference between / and //?

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**

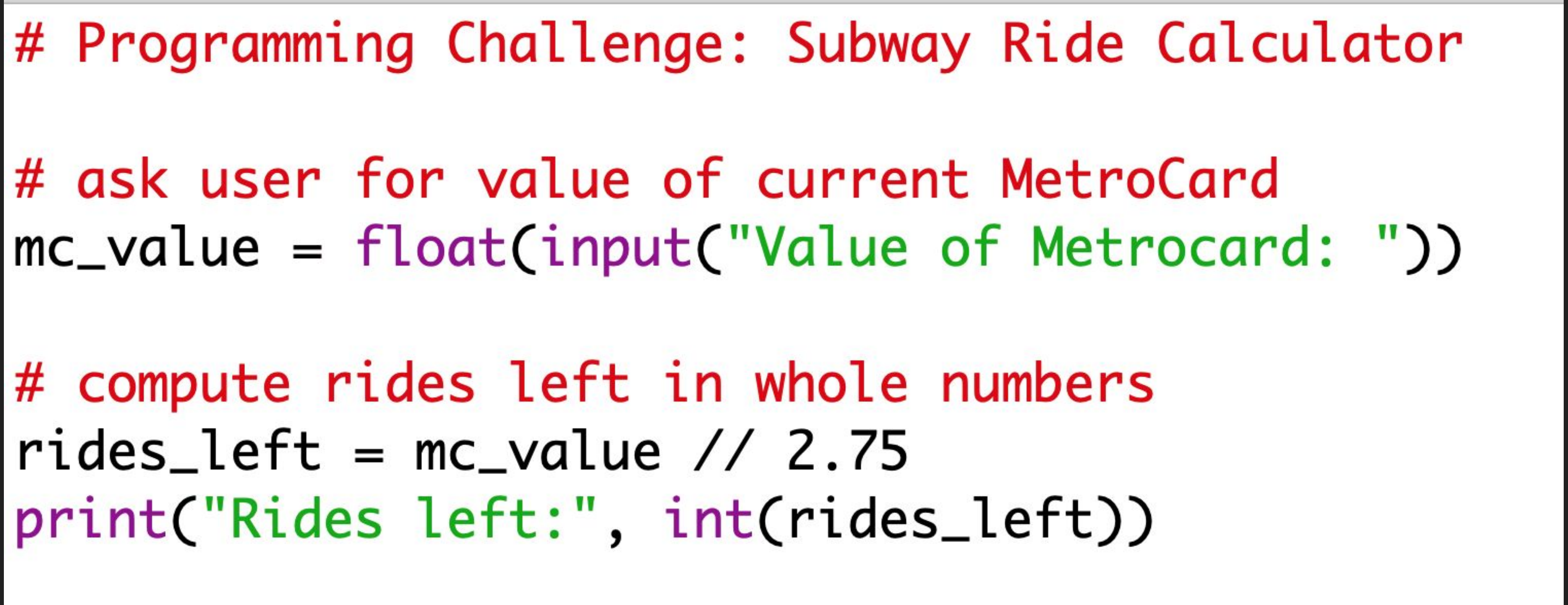
`6 // 2` → **3**

Programming Challenge: Subway Ride Calculator

- Write a program that asks the user for the value of their current Metrocard
- Compute how many rides they have left on their card. Only provide whole number results (i.e. you cannot have 3.5 rides left on a card)
- The fare for one ride is $2.75

```
Value of Metrocard: 25.50

> Rides Left: 9
```

Programming Challenge: Subway Ride Calculator



```
# Programming Challenge: Subway Ride Calculator

# ask user for value of current MetroCard
mc_value = float(input("Value of Metrocard: "))

# compute rides left in whole numbers
rides_left = mc_value // 2.75
print("Rides left:", int(rides_left))
```

Convert the expression

2^4 → **2**4**

$\sqrt{4}$ → **4 ** (1 / 2)**

What does the `%` operator do?

What does the `%` operator do?

Remainder (Modulo) Operator

`12 % 7` → **5**

`34 % 3` → **1**

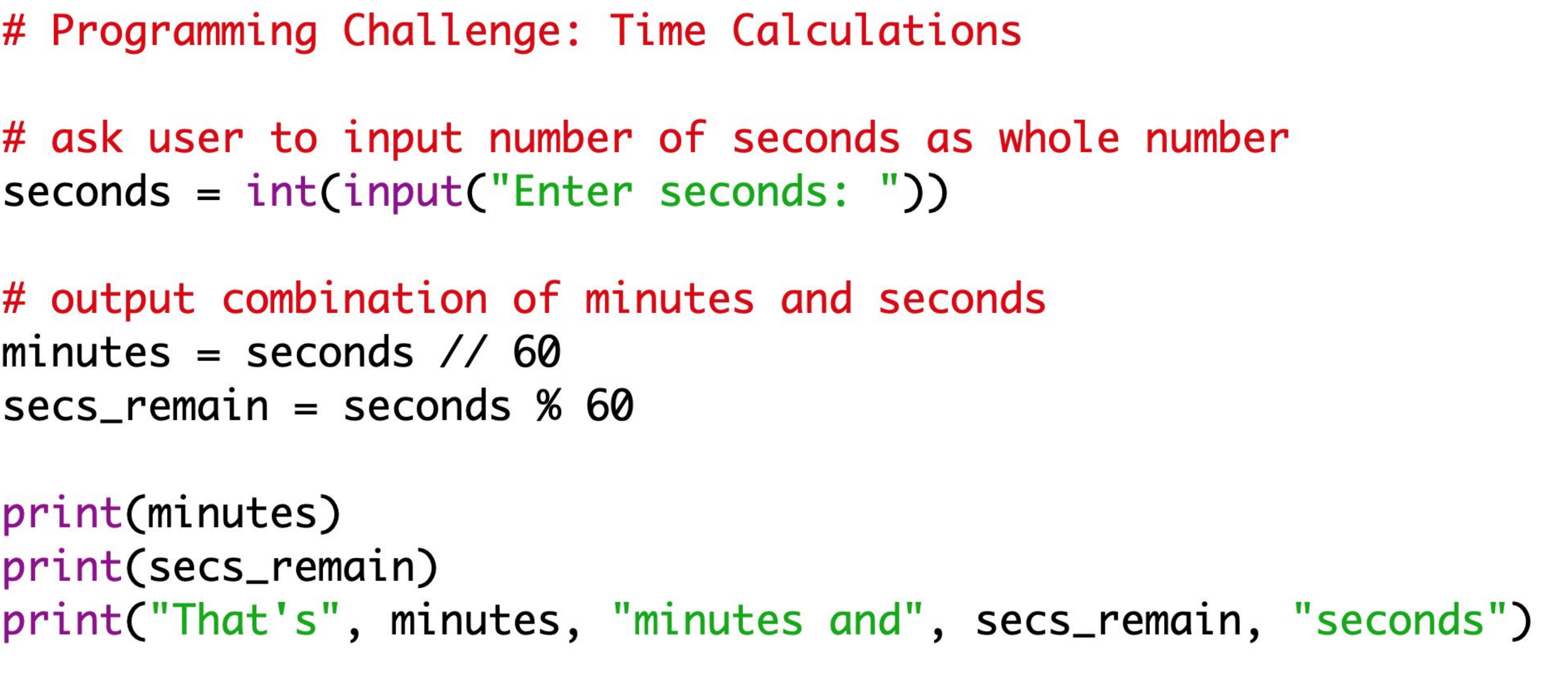
`10 % 5` → **0**

Programming Challenge: Time Calculations

- Ask the user to input a number of seconds as a whole number. Then express the time value inputted as a combination of minutes and seconds.

```
Enter seconds: 110

> That’s 1 minute and 50 seconds
```

Programming Challenge: Time Calculations



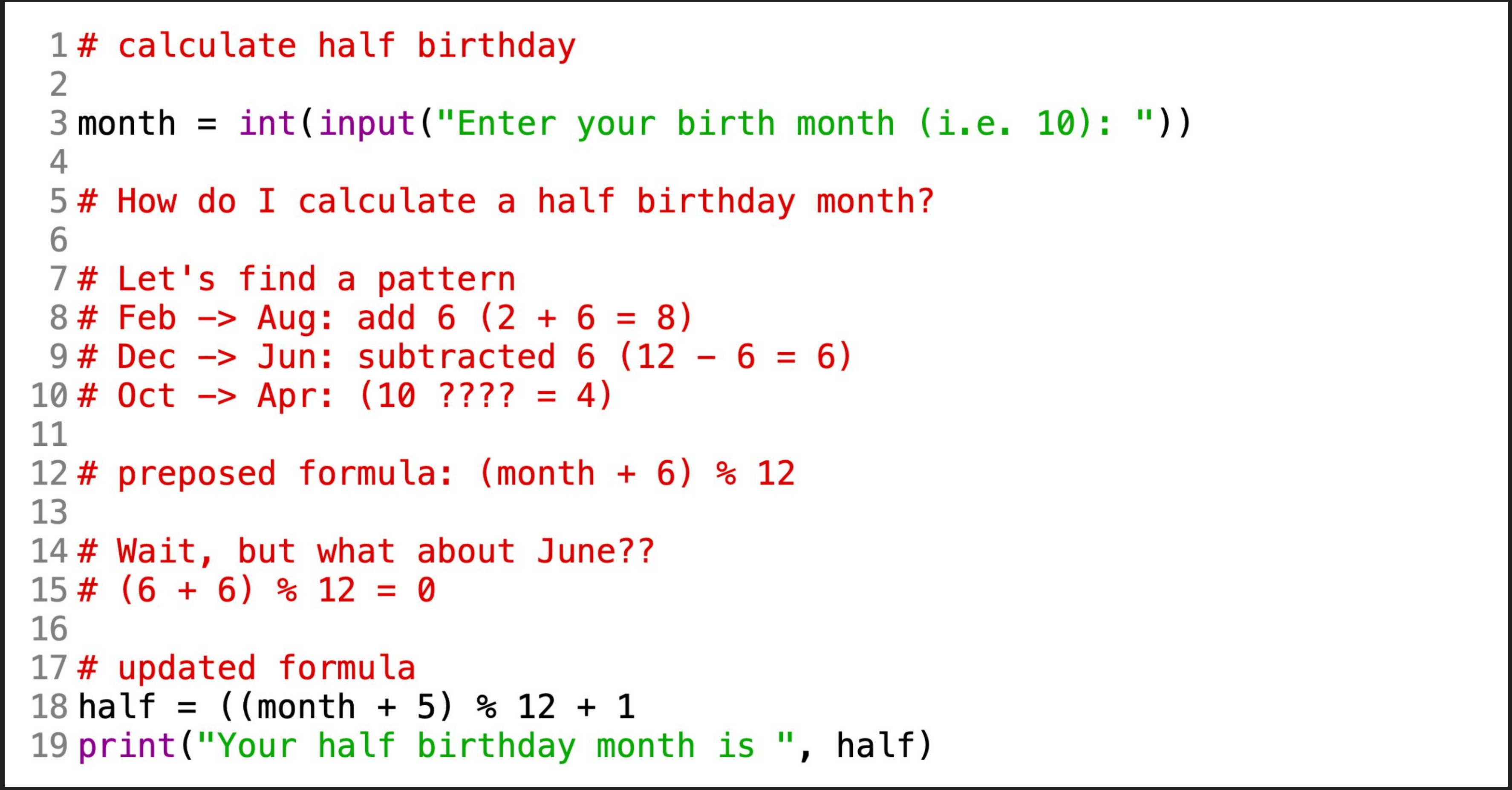
```
# Programming Challenge: Time Calculations

# ask user to input number of seconds as whole number
seconds = int(input("Enter seconds: "))

# output combination of minutes and seconds
minutes = seconds // 60
secs_remain = seconds % 60

print(minutes)
print(secs_remain)
print("That's", minutes, "minutes and", secs_remain, "seconds")
```

Programming Challenge: Calculate Half Birthday Month

- Write a program that takes a user's birth month and returns their half birthday month

```
Enter your birth month (i.e. 10): 10

> Your half birthday month is: 4
```

Programming Challenge: Calculate Half Birthday

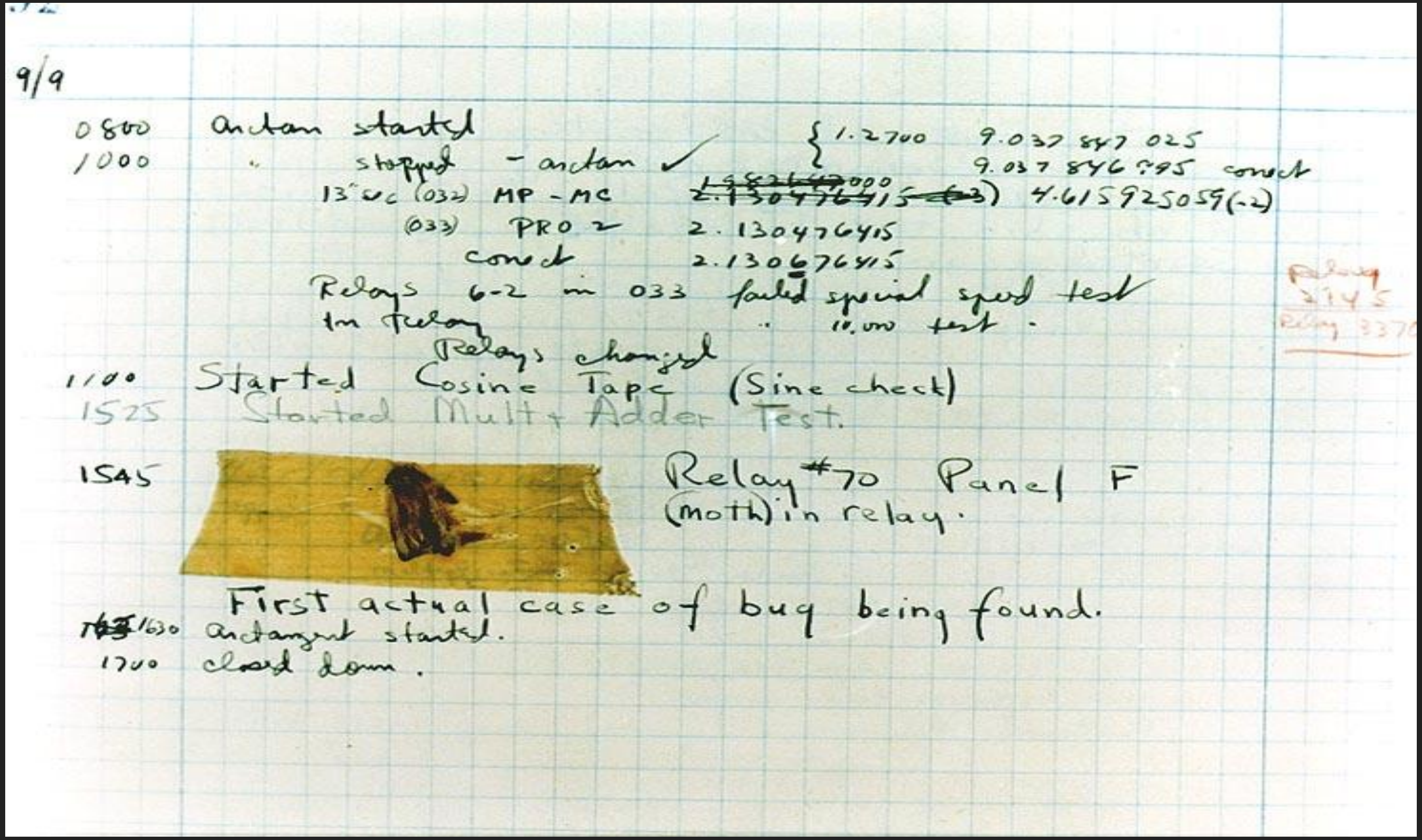


```python
# calculate half birthday

month = int(input("Enter your birth month (i.e. 10): "))

# How do I calculate a half birthday month?

# Let's find a pattern
# Feb -> Aug: add 6 (2 + 6 = 8)
# Dec -> Jun: subtracted 6 (12 - 6 = 6)
# Oct -> Apr: (10 ???? = 4)

# preposed formula: (month + 6) % 12

# Wait, but what about June??
# (6 + 6) % 12 = 0

# updated formula
half = ((month + 5) % 12 + 1
print("Your half birthday month is ", half)
```

Error + Error Types

“Debugging”

1947, Harvard Mark II Computer, Grace Hopper’s log book:



9/9

0800 arctan started

1000 " stopped - arctan ✓ {1.2700 9.037 847 025 / 9.037 846 995 correct

13" uc (032) MP - MC ~~1.982147000~~ ~~2.130476415 (-3)~~) 4.615925059(-2)

(033) PRO 2 2.130476415

correct 2.130676415

Relays 6-2 in 033 failed special speed test

In relay " 10,000 test .

Relays changed

1100 Started Cosine Tape (Sine check)

1525 Started Mult + Adder Test.

1545 Relay #70 Panel F

(moth) in relay.

First actual case of bug being found.

~~1630~~ 1630 antangent started.

1700 closed down.

Relay 2145

Relay 3370

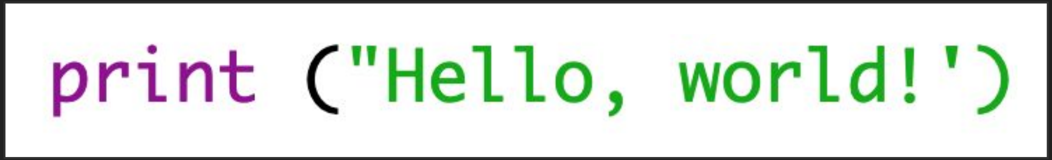
Types of Errors

- **Syntax Errors:**
  Does not follow the rules of the language

- **Runtime Errors:**
  Code is fine, but program crashes when it runs

- **Logic Errors:**
  Hardest to find; program is correct syntactically, but the output is unanticipated or outright wrong

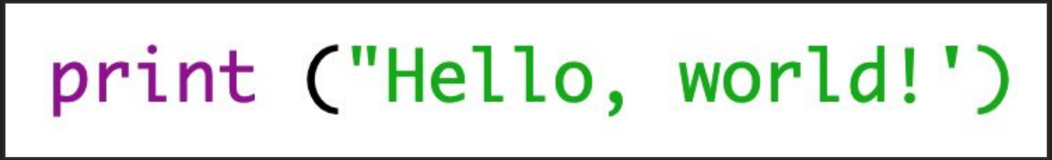
What kind of error is this?



```
print ("Hello, world!')
```

- Syntax
- Runtime
- Logical

What kind of error is this?

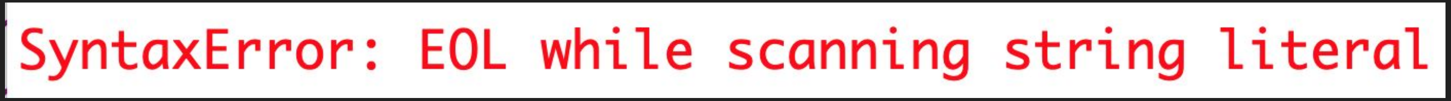


```
print ("Hello, world!')
```

→ Syntax

— Runtime

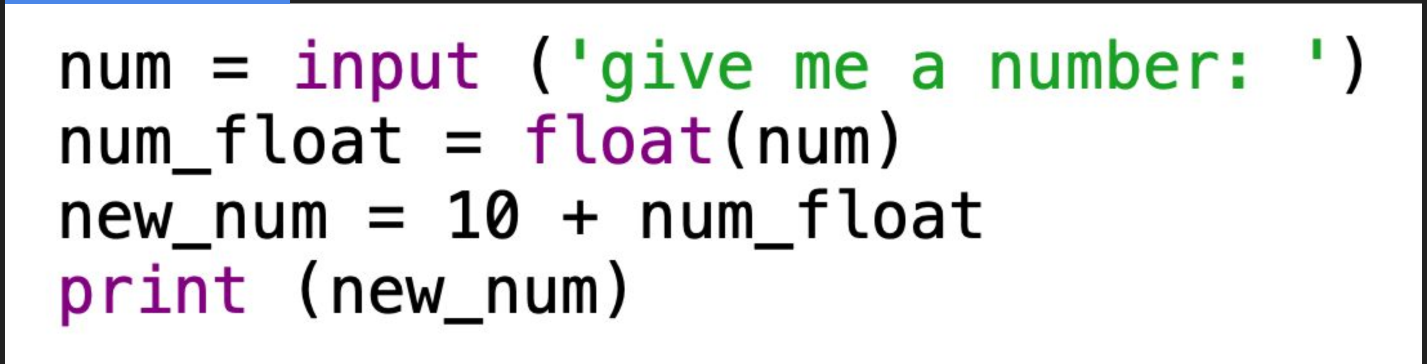
— Logical



```
SyntaxError: EOL while scanning string literal
```

What kind of error is this?

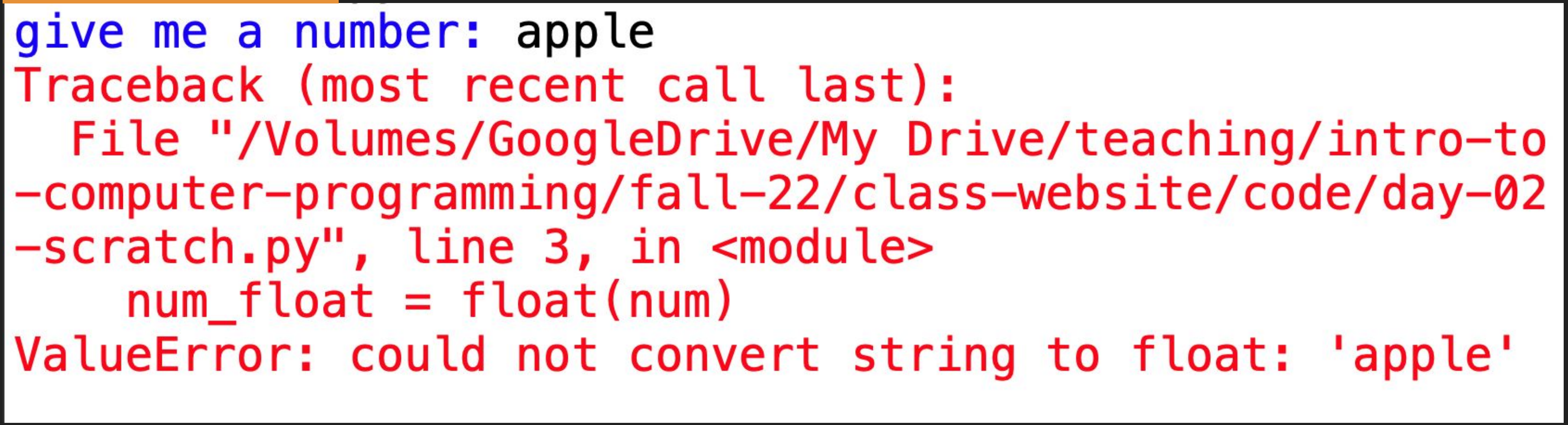
Source:



```python
num = input ('give me a number: ')
num_float = float(num)
new_num = 10 + num_float
print (new_num)
```

- Syntax
- Runtime
- Logical

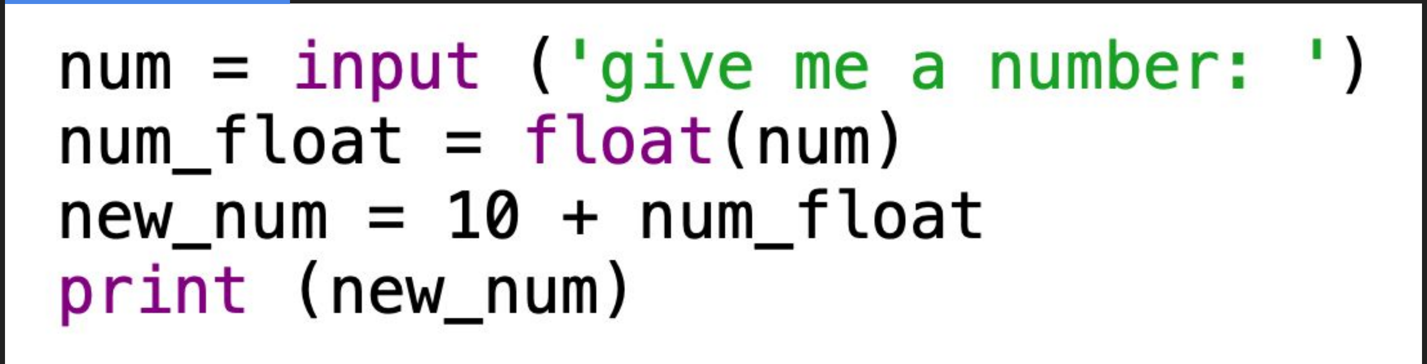
Execution:



```
give me a number: apple
Traceback (most recent call last):
  File "/Volumes/GoogleDrive/My Drive/teaching/intro-to
-computer-programming/fall-22/class-website/code/day-02
-scratch.py", line 3, in <module>
    num_float = float(num)
ValueError: could not convert string to float: 'apple'
```

What kind of error is this?

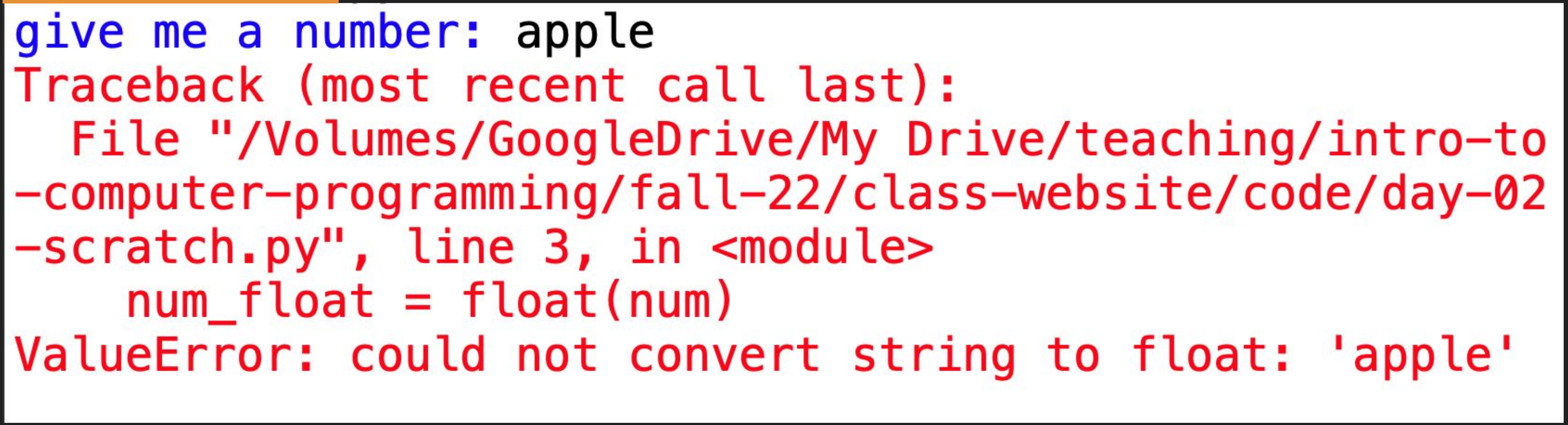
Source:



```
num = input ('give me a number: ')
num_float = float(num)
new_num = 10 + num_float
print (new_num)
```

- Syntax
- → Runtime
- Logical

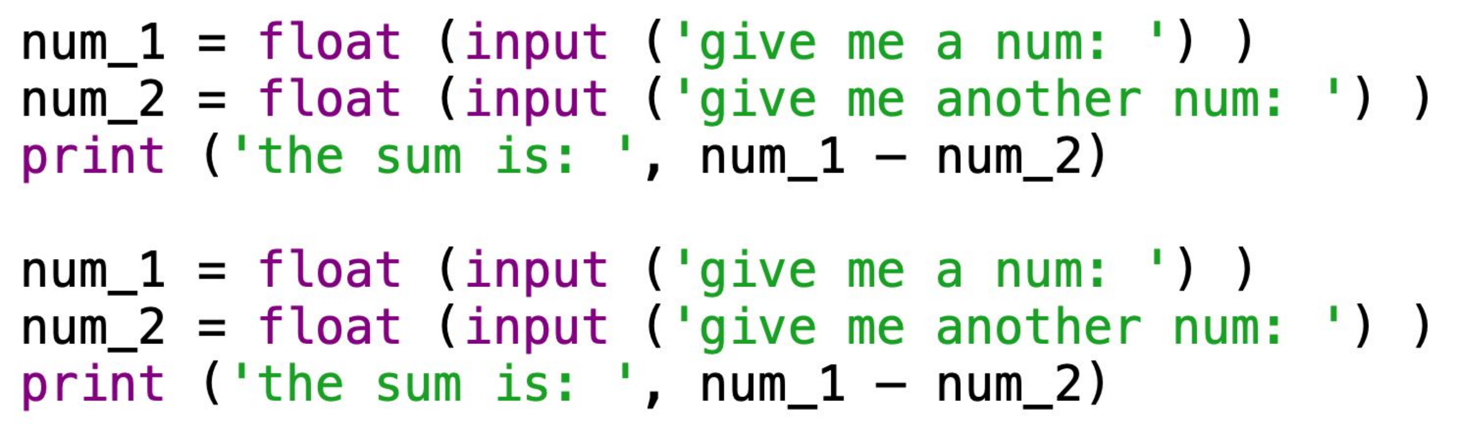
Execution:



```
give me a number: apple
Traceback (most recent call last):
  File "/Volumes/GoogleDrive/My Drive/teaching/intro-to
-computer-programming/fall-22/class-website/code/day-02
-scratch.py", line 3, in <module>
    num_float = float(num)
ValueError: could not convert string to float: 'apple'
```

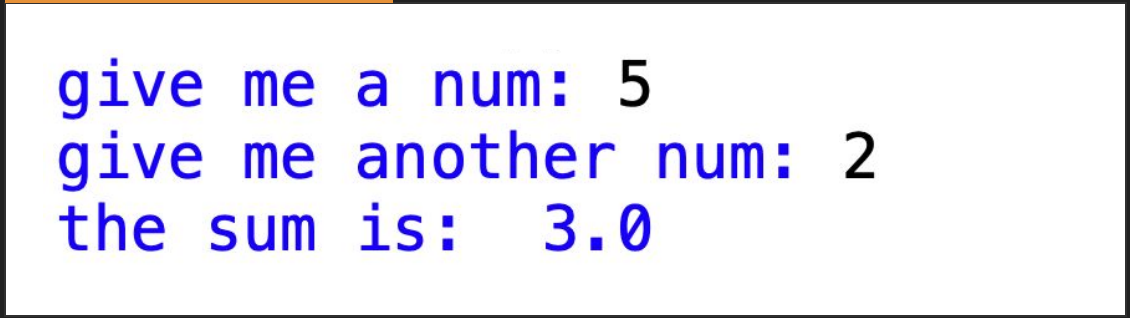
What kind of error is this?

Source:



```
num_1 = float (input ('give me a num: ') )
num_2 = float (input ('give me another num: ') )
print ('the sum is: ', num_1 - num_2)

num_1 = float (input ('give me a num: ') )
num_2 = float (input ('give me another num: ') )
print ('the sum is: ', num_1 - num_2)
```

Execution:

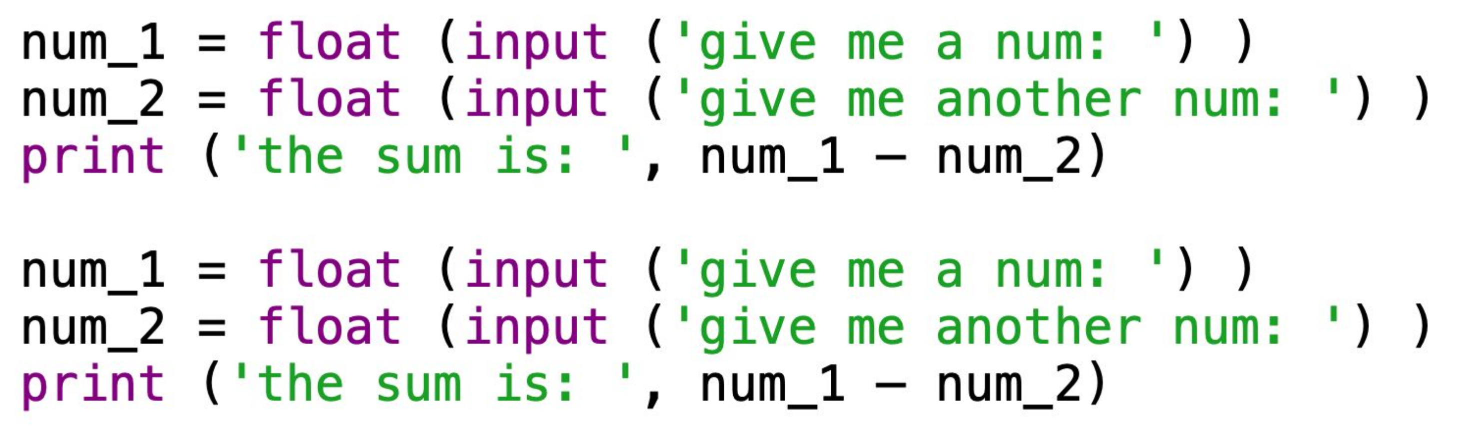


```
give me a num: 5
give me another num: 2
the sum is:  3.0
```

- Syntax
- Runtime
- Logical

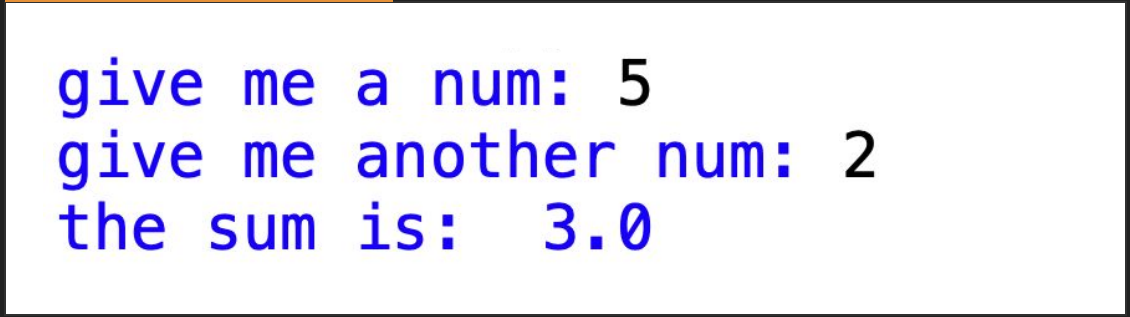
What kind of error is this?

Source:



```
num_1 = float (input ('give me a num: ') )
num_2 = float (input ('give me another num: ') )
print ('the sum is: ', num_1 - num_2)

num_1 = float (input ('give me a num: ') )
num_2 = float (input ('give me another num: ') )
print ('the sum is: ', num_1 - num_2)
```

Execution:



```
give me a num: 5
give me another num: 2
the sum is:  3.0
```

– Syntax

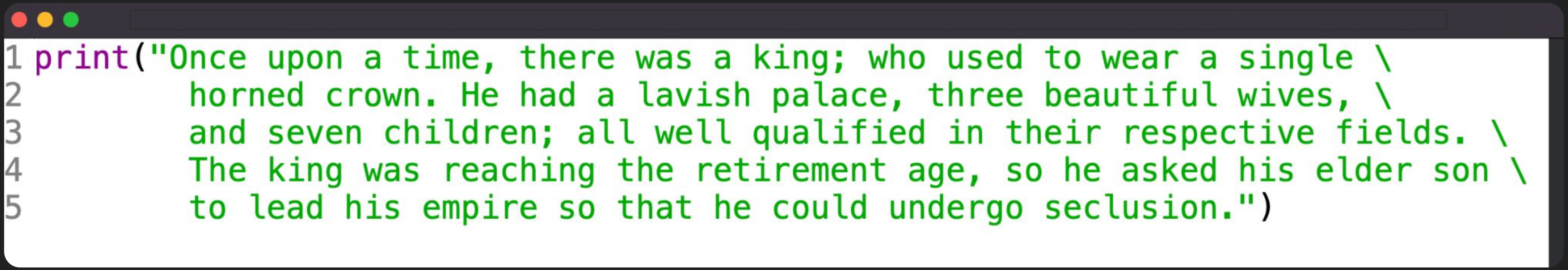
– Runtime

→ Logical

Formatting Strings + Numbers

Line Continuation

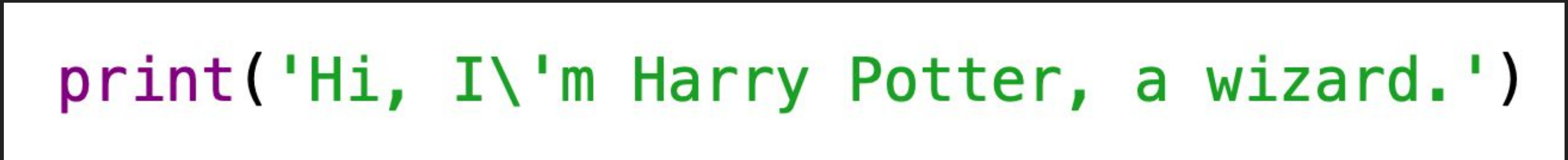
- Sometimes the code you write can get very long
- You can use the `\` symbol to indicate to Python that you would like to continue your code onto another line



```
print("Once upon a time, there was a king; who used to wear a single \
        horned crown. He had a lavish palace, three beautiful wives, \
        and seven children; all well qualified in their respective fields. \
        The king was reaching the retirement age, so he asked his elder son \
        to lead his empire so that he could undergo seclusion.")
```

Escape Characters

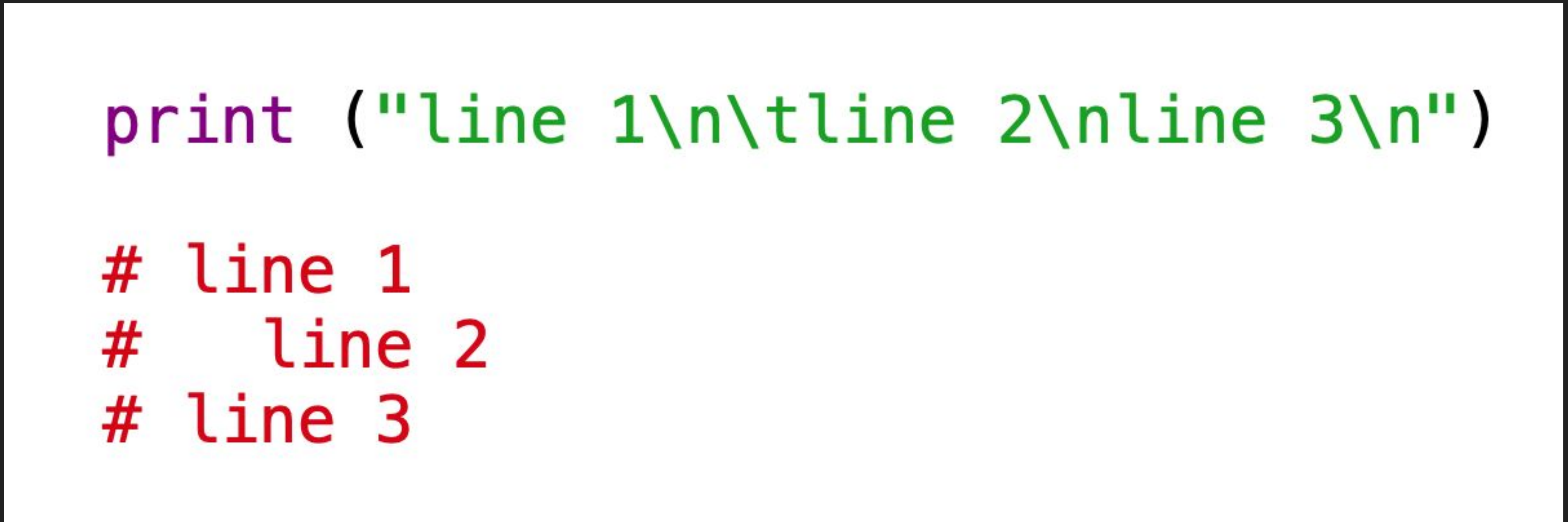
- An “escape character” allows you to perform special actions inside the confines of a delimiter
- In Python, the escape character is `\`
- It causes Python to treat the next character as “special”



```
print('Hi, I\'m Harry Potter, a wizard.')
```

Escape Characters

- There are a number of special characters you can use in conjunction with the escape character to perform special string operations
- `\n` forces a line break
- `\t` creates a tab



```
print ("line 1\n\tline 2\nline 3\n")

# line 1
#   line 2
# line 3
```

The `format` function

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

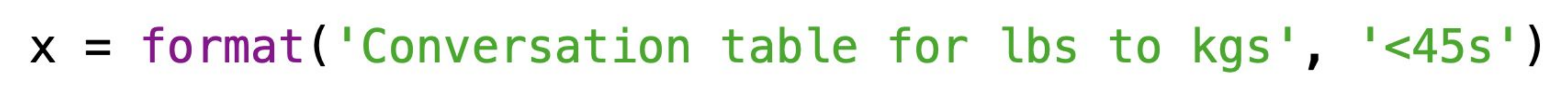
The `format` function

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"`		

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

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
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
```

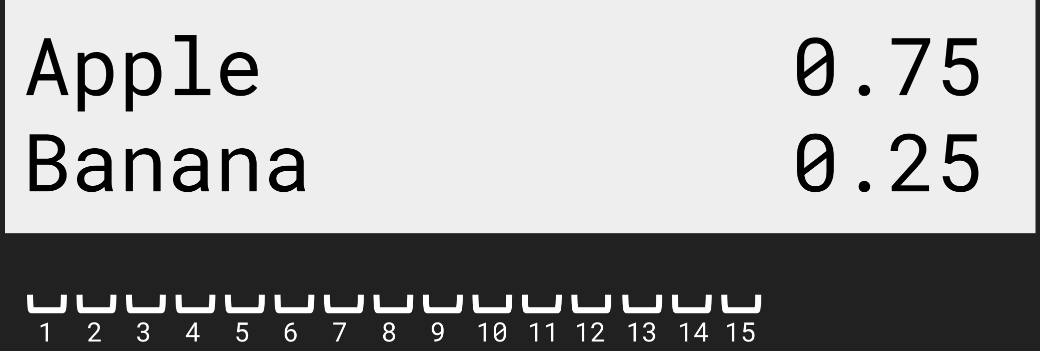
```
print(format(Harriet, '<15s'))
```

```
Harriet
␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
```

“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)
```

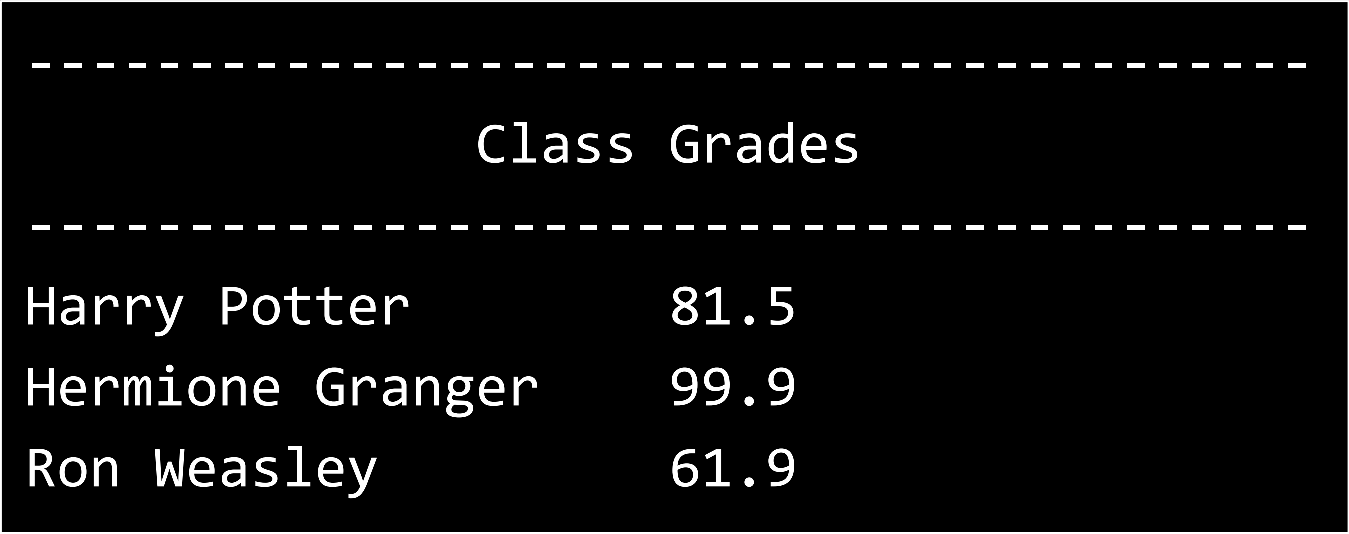


```
Apple           0.75
Banana          0.25
```

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

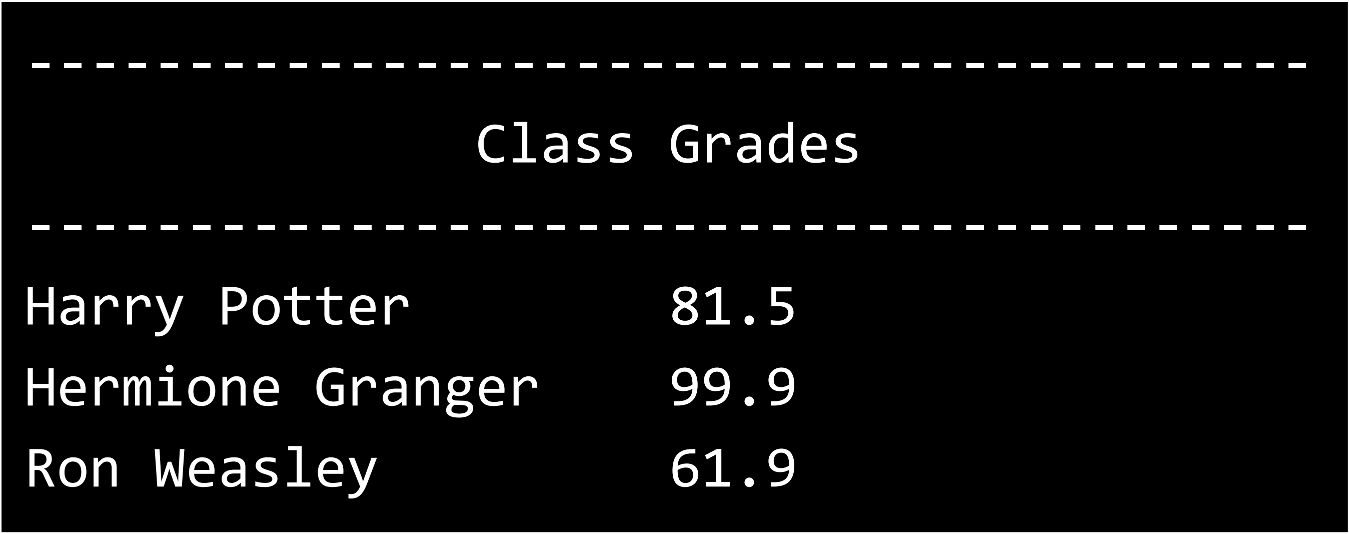
Programming Challenge: Formatting a Table

- Reproduce the following table using format (40 spaces wide)



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

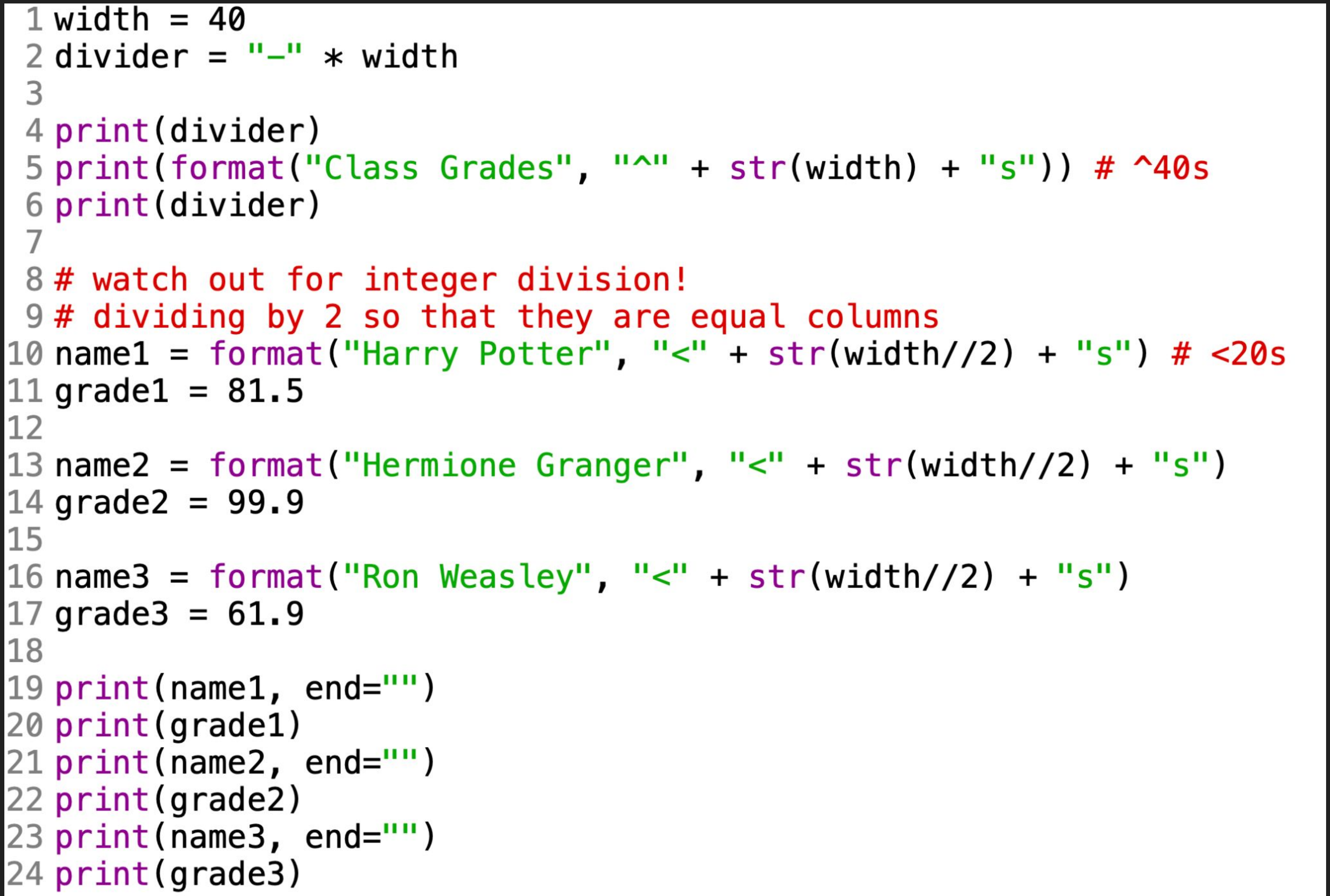
Programming Challenge: Formatting a Table (Extension)



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

- **Take in user input to specify how wide they want the table to be.**
  - How do I replace my formatting spec with user input?
  - How can we incorporate variables/formulas?
  - **Hint**: concatenation!

Programming Challenge: Formatting a Table (Extension)



```python
width = 40
divider = "-" * width

print(divider)
print(format("Class Grades", "^" + str(width) + "s")) # ^40s
print(divider)

# watch out for integer division!
# dividing by 2 so that they are equal columns
name1 = format("Harry Potter", "<" + str(width//2) + "s") # <20s
grade1 = 81.5

name2 = format("Hermione Granger", "<" + str(width//2) + "s")
grade2 = 99.9

name3 = format("Ron Weasley", "<" + str(width//2) + "s")
grade3 = 61.9

print(name1, end="")
print(grade1)
print(name2, end="")
print(grade2)
print(name3, end="")
print(grade3)
```

Homework

- Assignment #1 (due next class)
- Self-Paced Learning Module #3 (due in one week)
- Quiz #3 (due in one week)