Class 3

Introduction to Python: Strings, Integers, and Floats

Questions?

- Email me at metrocoders312@gmail.com and I will get back to you as soon as I can. ☺

Review from Last Week

- The code to have your program say a personal hello to you:
  - `name = input(‘Give your name : ‘)`
  - `print(‘Hello ‘ + name)`
- In this code:
  - name is variable
  - the program always reads user input as a string
  - What is a variable and string? Let’s find out!

What is a variable?

- Imagine you have a box labeled CANDY.
  Inside the box CANDY, there are 20 pieces of candy.
  Then, we can say CANDY = 20.
- OR
- Let’s say we have a cage BIRD_CAGE.
  Inside the cage BIRD_CAGE, we have a parrot.
  Then, BIRD_CAGE = parrot.
- In these examples, the variables are CANDY and BIRD_CAGE. These are like containers which hold the values 20 and parrot, respectively.
- In Python, we can write these variables as follows:
  - `CANDY = 20`
  - `BIRD_CAGE = ‘parrot’`
- NOTE: variables are case-sensitive:
  - `Age ≠ age ≠ AGE`

Going back to last week…

- Let’s go back to our code from last week:
  - `name = input(‘Give your name : ‘)`
  - `print(‘Hello ‘ + name)`
- **name** is a variable that stores the value that the user inputs, meaning that name is a container that holds the name of the user
- In the terminal, the session could go as follows:

```
Give your name : Miss Maria
Hello Miss Maria
```

What is a string?

- A string in Python is a type of value that is encompassed by quotes
- Correct examples:

```
string1 = ‘hello’
string2 = ‘12’   #a string can be a number, but we cannot do math
                     #unless we convert it to an integer
```

- Incorrect examples:

```
string3 = “hello’   #two different types of quotation marks

string4 = 12         #a string must include quotation marks
```

- Note: in Python we can make notes in our code using #
  (the program does not read anything in the line after # as code)

When we do “math” with strings…

- String concatenation happens!
- Ex:

Script	Shell/Terminal
➢S1 = ‘Miss Maria’ ➢S2 = ‘is’ ➢S3 = ‘your teacher.’ ➢S = S1+S2+S3 ➢print(S1) ➢print(S2) ➢print(S3) ➢print(S)	➢ Miss Maria ➢ is ➢ your teacher. ➢ Miss Maria is your teacher.

Moving on to Integers!

- Integers are whole numbers (not fractions or decimals!)
- to convert a string number to an integer type → use int() function

Script	Shell/Terminal
➢s1 = ‘34’ ➢s2 = ‘12’ ➢n1 = int(s1) # int() takes away quotes ➢n2 = int(s2) ➢print(s1 + s2) ➢print(n1 + n2)	➢ 3412 ➢ 46

Going from an integer to a string

- To convert an integer to a string→ use str() function

Script	Shell/Terminal
➢s = ‘Number = ’ ➢n = 123 ➢print(s + n + ‘.’)	➢ Error! (Does not work.)

- The above does not work because we cannot “add” (concatenate) a string and an integer since they are different types!

Script	Shell/Terminal
➢s = ‘Number = ’ ➢n = 123 ➢n = str(n) ➢print(s + n + ‘.’)	➢ Number = 123.

One more type: float

- In simple words, a float is a decimal number.
  - `f = 1.23243235345`
- To convert to a string from a float:
  - `s = str(f)`
- To convert to an integer from a float:
  - `n = int(f) # this cuts off the numbers after the decimal point`
- To convert to a float from a string:
  - `s = ‘12.2’`
  - `f = float(s)`

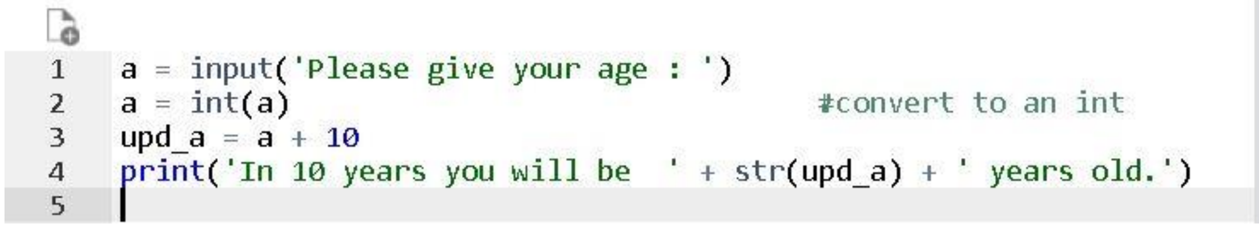
- Note : You can add a float and integer mathematically.

Problem:

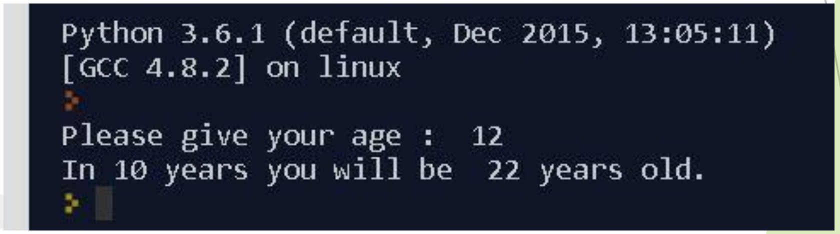
- Input: Ask user for their age.
- Output: Print out what their age will be in 10 years.

Solution:

Script	Shell/Terminal
➢a = input(‘Please give your age : ’) ➢a = int(a) #convert to an int ➢upd_a = a + 10 ➢print(‘In 10 years you will be ’ + upd_a + ‘ years old.’)	➢ Please give your age : 12 ➢ In 10 years you will be 22 years old.



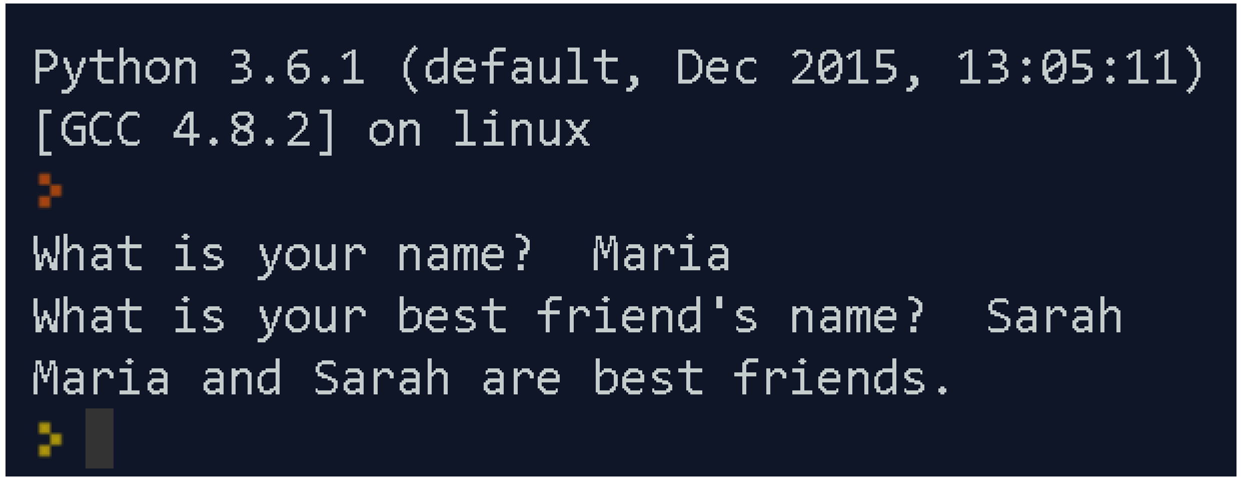
```
a = input('Please give your age : ')
a = int(a)                          #convert to an int
upd_a = a + 10
print('In 10 years you will be  ' + str(upd_a) + ' years old.')

```



```
Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux
>
Please give your age :  12
In 10 years you will be  22 years old.
>
```

Challenge Problem 1:

- Ask the user for her name.
- Ask the user for her best friend’s name.
- Print out a statement saying that user_name and friend_name are best friends.
- An example of how the solution should look like in terminal:

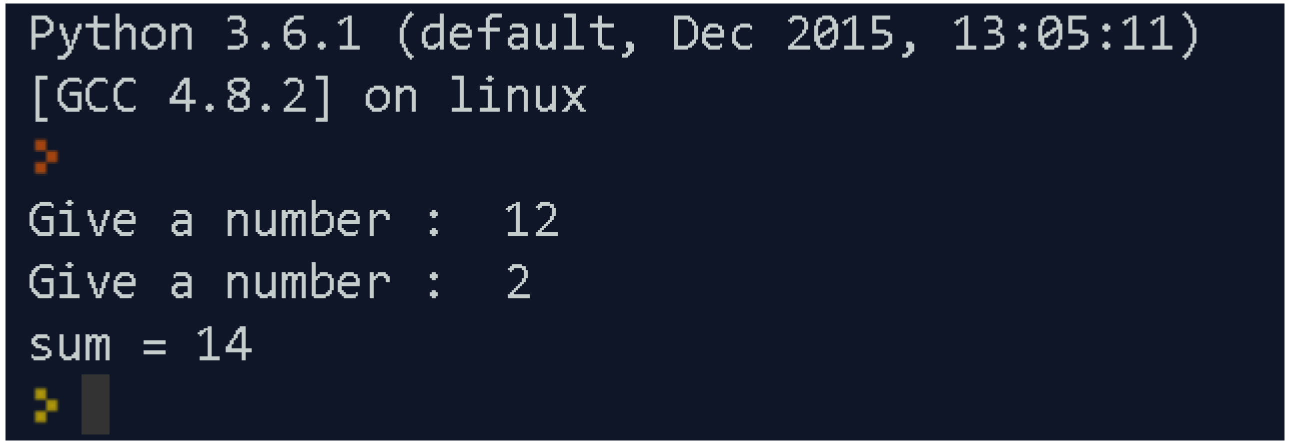


```
Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux
>
What is your name?  Maria
What is your best friend's name?  Sarah
Maria and Sarah are best friends.
>
```

- Now try and see if you can figure out the code!
- Hint: to get the apostrophe in “friend’s” to not mess with the full string you must code it as follows: `‘What is your best friend\’s name? ’`

Challenge Problem 2:

- Ask the user for a number A.
- Ask the user for a second number B.
- Print out a statement saying that sum = A + B .
- An example of how the solution should look like in terminal:

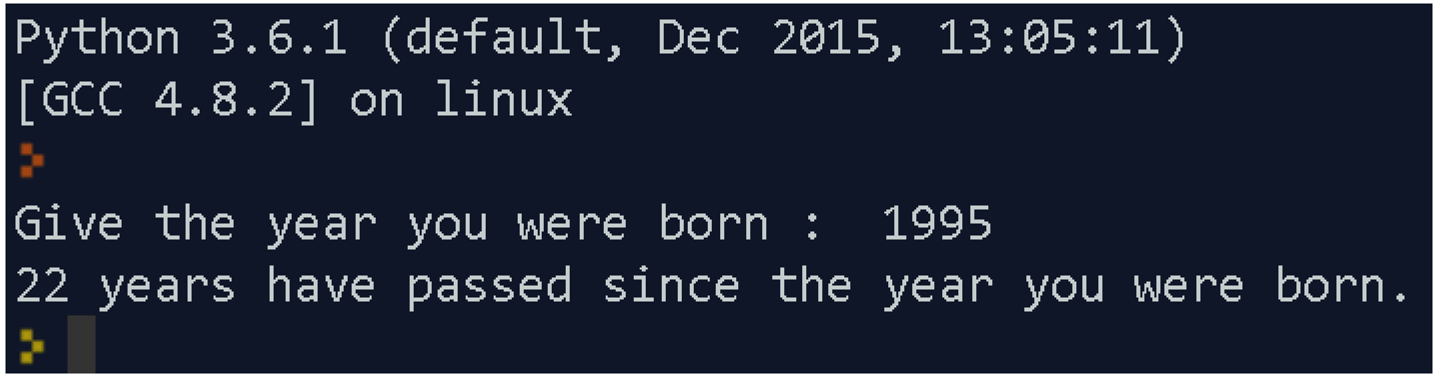


```
Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux
>
Give a number :  12
Give a number :  2
sum = 14
>
```

- Now try and see if you can figure out the code!
- Remember: user input it taken in as a string. To do math with it, you need to convert it to an integer type.

Challenge Problem 3:

- Ask the user for the year they were born.
- Subtract this from 2017, the current year we are in.
- The solution will print out the number of years passed since the birth of the user.
- An example of how the solution should look like in terminal:



```
Python 3.6.1 (default, Dec 2015, 13:05:11)
[GCC 4.8.2] on linux

Give the year you were born :  1995
22 years have passed since the year you were born.
```

- Now try and see if you can figure out the code!
- Remember: user input it taken in as a string. To do math with it, you need to convert it to an integer type.