PYTHON

From Simple to Complex

With

Examples

AYESHA NOREEN

Bachelor's in Software Engineering,

Master's in Computer Science

from COMSATS University, Islamabad

NOTE!!!

In these notes Screenshots of practice examples and coding are added. The code files are also available in code folder that contain .ipynb files that are created on Jupyter notebook.

Chapter1
Python Basic Concepts

- ## Comments in Python

comments are used for user understanding.

○ Single Line Comment

# is used for comment a single line in python.

e.g. #This is my 1st program.

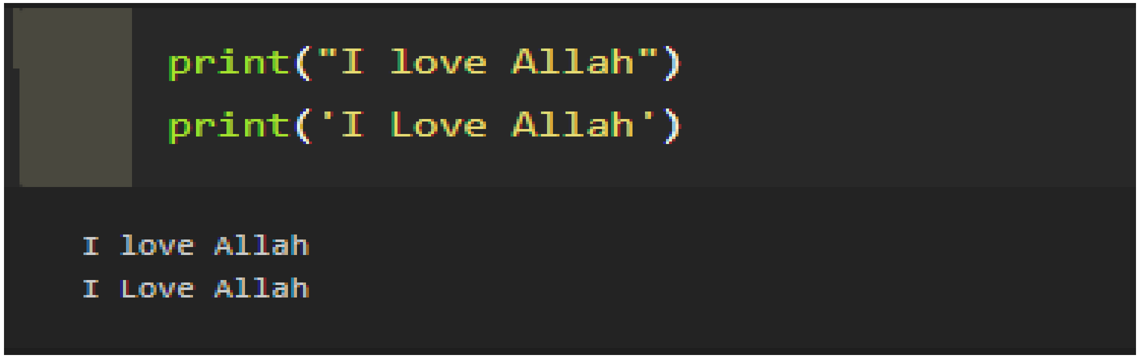
A shortcut that is used for comments is goto line where you want to comment press ctrl+\

○ Multiline Comment

For multiline comment add /* at start and */ at end or ““ Multilines are here””

• Print() Function

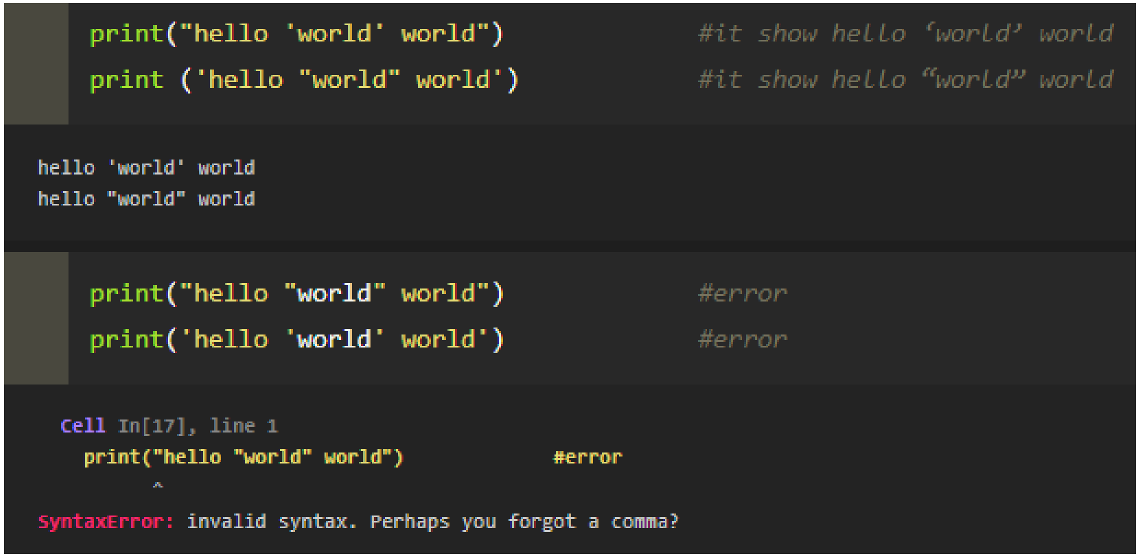
Print() function is used for output. We may use single or double quotes. As,



```
print("I love Allah")
print('I Love Allah')
```

```
I love Allah
I Love Allah
```

We can use single quotes Inside double quotes and double quotes inside single quotes but if we use double quotes inside double quotes or single quotes inside single quotes than give error. As,



```
print("hello 'world' world")            #it show hello ‘world’ world
print ('hello "world" world')           #it show hello “world” world
```

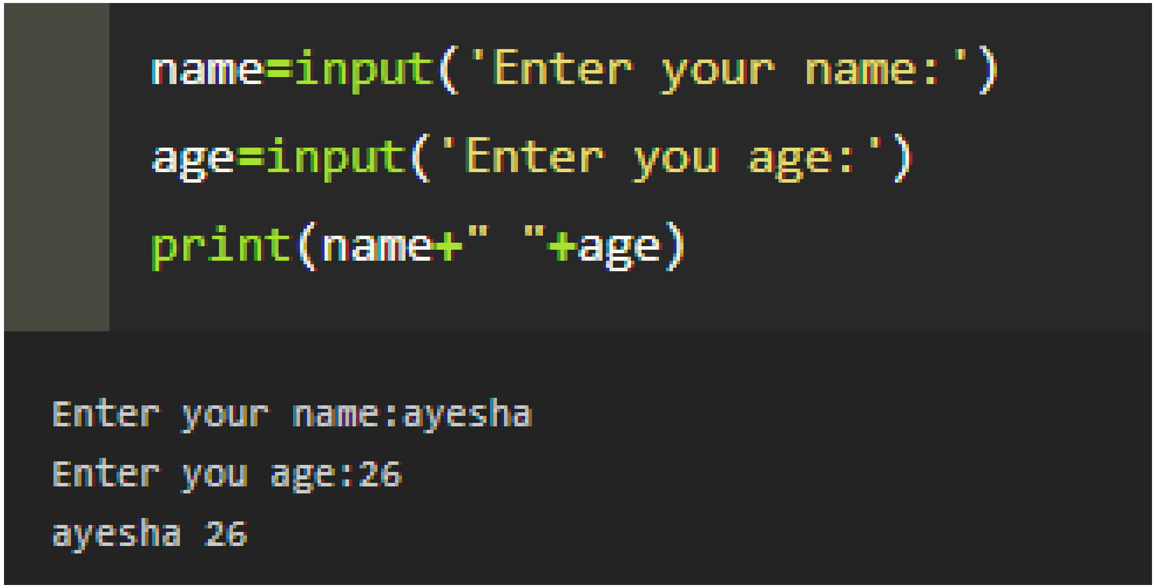
```
hello 'world' world
hello "world" world
```

```
print("hello "world" world")            #error
print('hello 'world' world')            #error
```

```
  Cell In[17], line 1
    print("hello "world" world")              #error
         ^
SyntaxError: invalid syntax. Perhaps you forgot a comma?
```

- ## Input() Function

It is used to take input from user. It always take input in string if input is a number than it also treat it as a string. As,

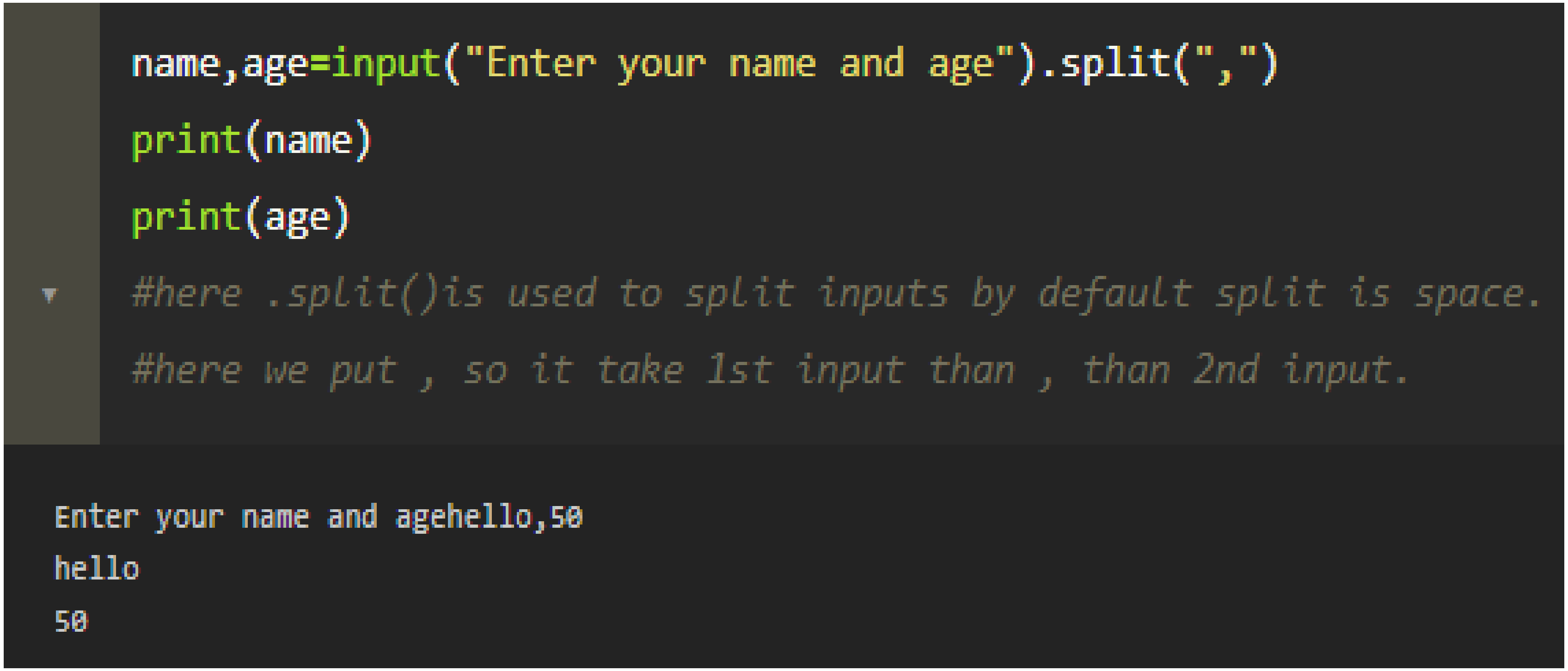


```
name=input('Enter your name:')
age=input('Enter you age:')
print(name+" "+age)
```

```
Enter your name:ayesha
Enter you age:26
ayesha 26
```

• Multiple Inputs in a single line

Multiple inputs can be taken in a single line and Split method is used to separate multiple inputs. As,



```
name,age=input("Enter your name and age").split(",")
print(name)
print(age)
#here .split()is used to split inputs by default split is space.
#here we put , so it take 1st input than , than 2nd input.
```

```
Enter your name and agehello,50
hello
50
```

- **Type() function**

It is used to check data type of a variable.

It is used to check we have which type of data. e.g.

```
string="ayesha"
List=["Ayesha"]
print(type(string))
#it prints <class 'str'> mean string type
print(type(List))
#it prints<class 'list'> mean list type
```

- **Type conversion**

It is used to convert one type into another type.

- ○ **int() Function**

It is used to convert a string into int.

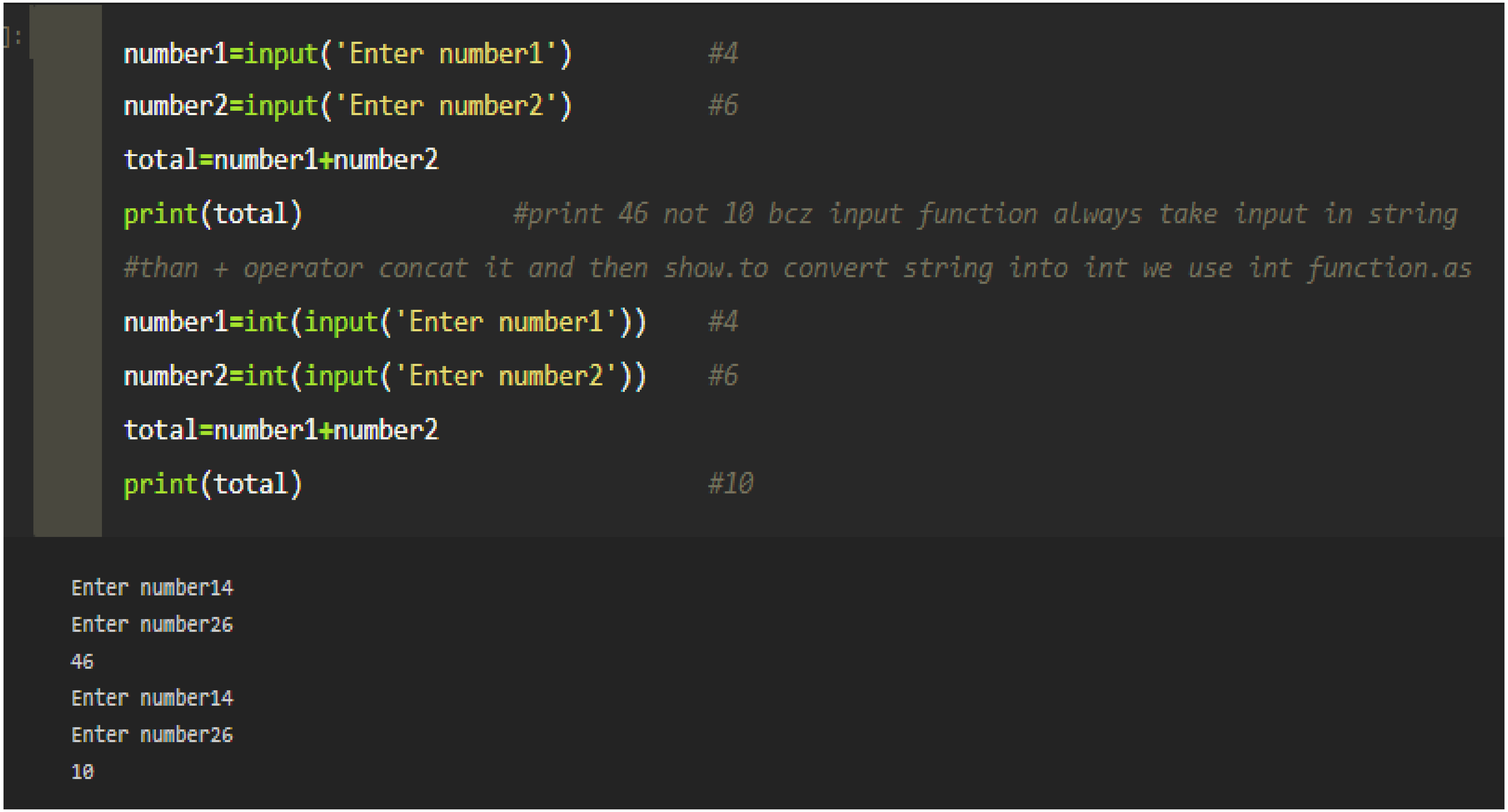
- ○ **float() Function**

It is used to convert a int or string into float

- ○ **str() Function**

It is used to convert a int or float into string

Example



```
number1=input('Enter number1')          #4
number2=input('Enter number2')          #6
total=number1+number2
print(total)              #print 46 not 10 bcz input function always take input in string
#than + operator concat it and then show.to convert string into int we use int function.as
number1=int(input('Enter number1'))     #4
number2=int(input('Enter number2'))     #6
total=number1+number2
print(total)                            #10
```

```
Enter number14
Enter number26
46
Enter number14
Enter number26
10
```

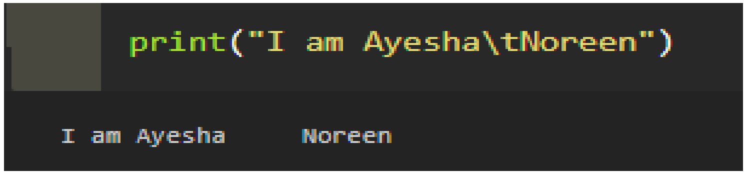
Data Types

Data types used in python are:

Text Type:	str
Numeric Types:	int, float, complex
Sequence Types:	list, tuple, range
Mapping Type:	dict
Set Types:	set, frozenset
Boolean Type:	bool
Binary Types:	bytes, bytearray, memoryview
None Type:	NoneType

• Escape Sequences

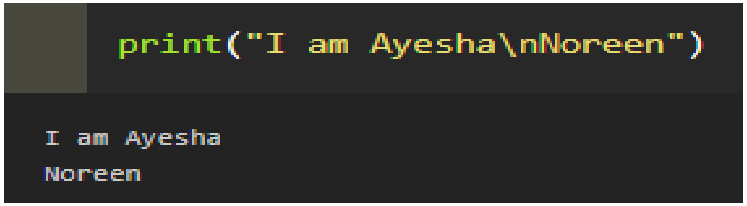
\t (tab)



```
print("I am Ayesha\tNoreen")
```

```
I am Ayesha	Noreen
```

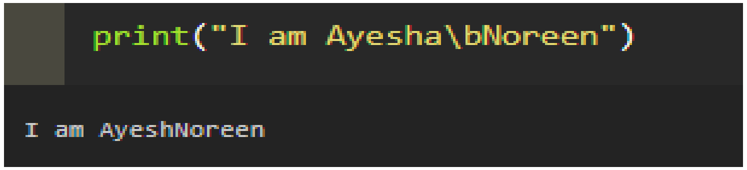
\n (new line)



```
print("I am Ayesha\nNoreen")
```

```
I am Ayesha
Noreen
```

\b (backspace)



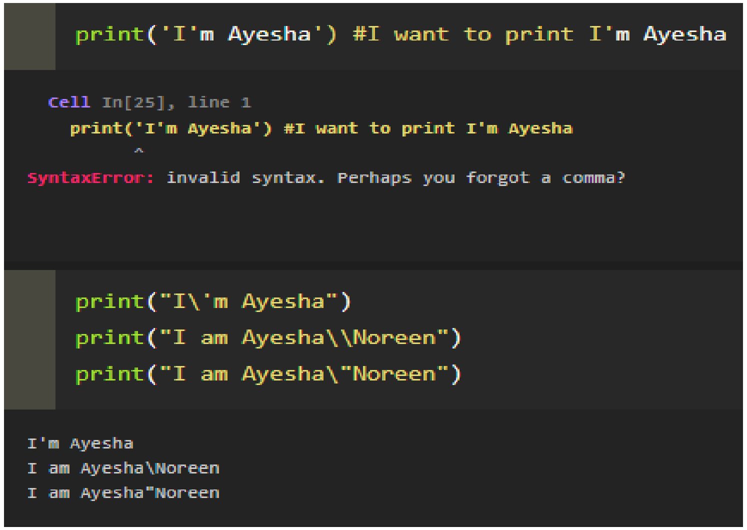
```
print("I am Ayesha\bNoreen")
```

```
I am AyeshNoreen
```

\\ (print single\)

\’ (print ’)

\”(print”)



```
print('I'm Ayesha') #I want to print I'm Ayesha
```

```
  Cell In[25], line 1
    print('I'm Ayesha') #I want to print I'm Ayesha
          ^
SyntaxError: invalid syntax. Perhaps you forgot a comma?
```

```
print("I\'m Ayesha")
print("I am Ayesha\\Noreen")
print("I am Ayesha\"Noreen")
```

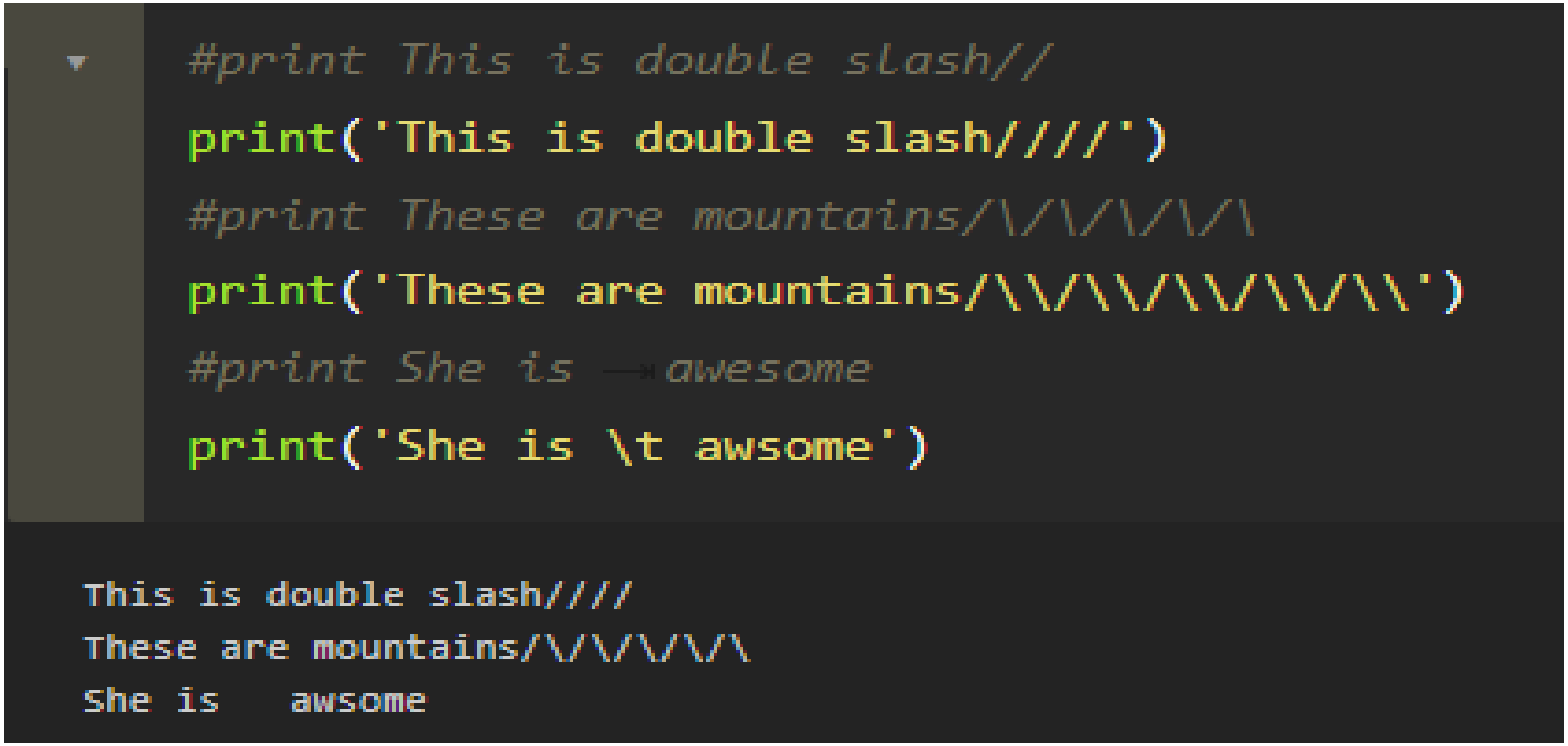
```
I'm Ayesha
I am Ayesha\Noreen
I am Ayesha"Noreen
```

- **TODO Task**

Print this is double slash//

These are mountains /\/\/\/\/\

She is awsome

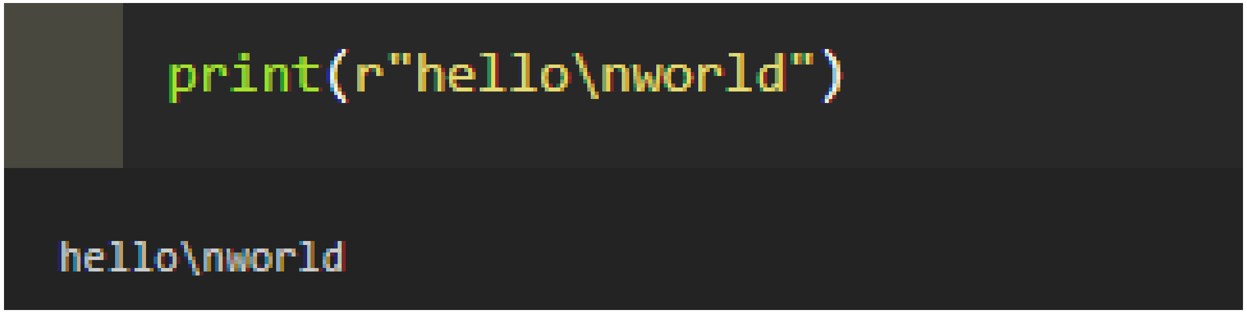


```
#print This is double slash//
print('This is double slash////')
#print These are mountains/\/\/\/\/\
print('These are mountains/\\/\\/\\/\\/\\')
#print She is →awesome
print('She is \t awsome')
```

```
This is double slash////
These are mountains/\/\/\/\/\
She is   awsome
```

- **Raw String**

If we want to treat any escape sequence as normal text than we put r before double quote. As,



```
print(r"hello\nworld")
```

```
hello\nworld
```

It prints hello\nworld
Not hello
World
Because of raw string r

- **Boolean properties**

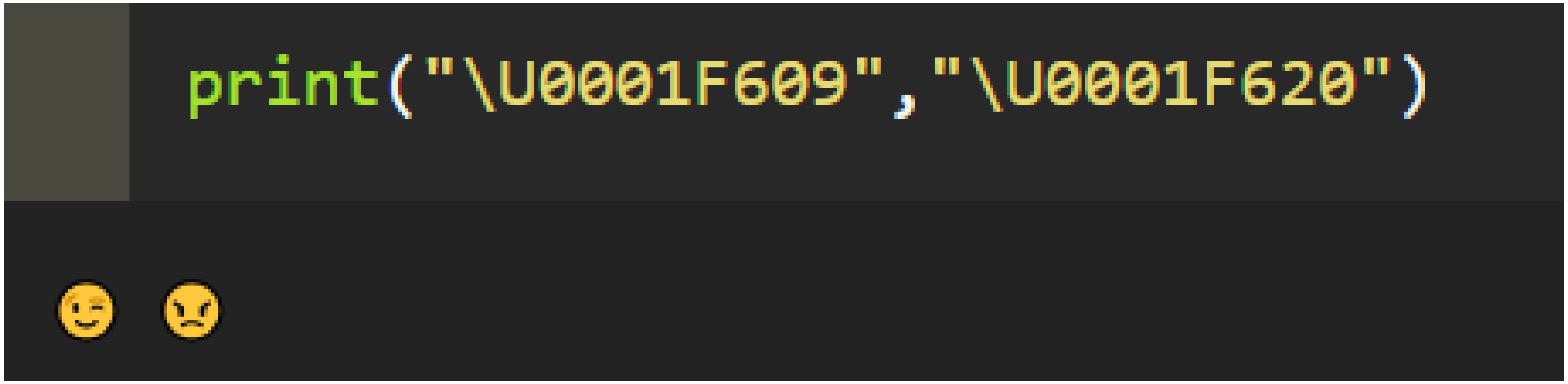
There are two type of Boolean properties.
A statement is either true or false.
If true return 1 if not return 0.

- **DRY Method**

DRY mean Don’t Repeate Yourself
It mean if a piece of code is repeated again and again than reduce it.

• Print Emoji

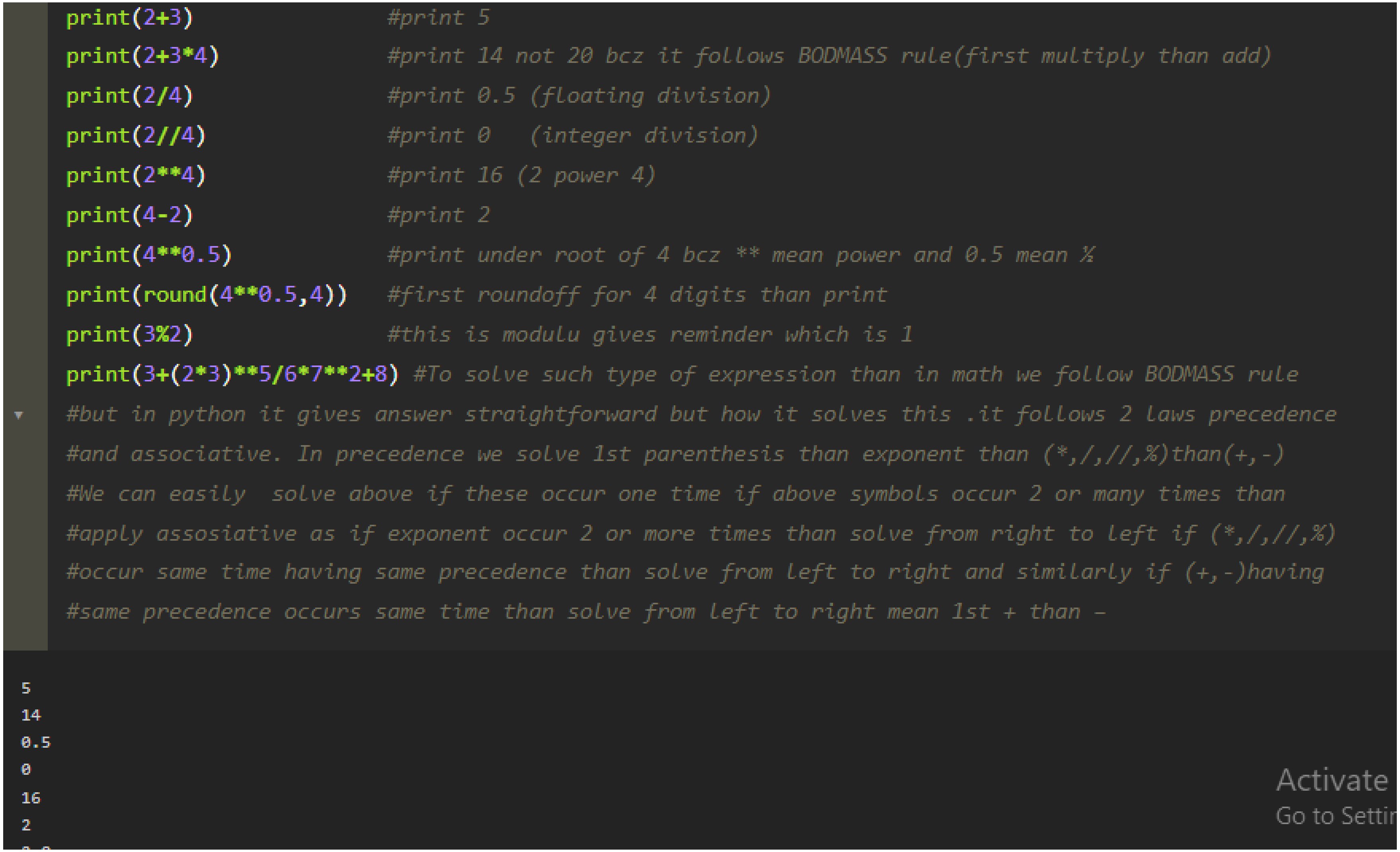
Search unicodes of emojis than copy and paste these uni codes.
For Example,
Unicode for an emoji is U+1F618 replace + with 000 and put backslash\ at start. As,



```
print("\U0001F609","\U0001F620")
```

```
😉 😠
```

• Calculator



```
print(2+3)                  #print 5
print(2+3*4)                #print 14 not 20 bcz it follows BODMASS rule(first multiply than add)
print(2/4)                  #print 0.5 (floating division)
print(2//4)                 #print 0   (integer division)
print(2**4)                 #print 16 (2 power 4)
print(4-2)                  #print 2
print(4**0.5)               #print under root of 4 bcz ** mean power and 0.5 mean ½
print(round(4**0.5,4))      #first roundoff for 4 digits than print
print(3%2)                  #this is modulu gives reminder which is 1
print(3+(2*3)**5/6*7**2+8) #To solve such type of expression than in math we follow BODMASS rule
#but in python it gives answer straightforward but how it solves this .it follows 2 laws precedence
#and associative. In precedence we solve 1st parenthesis than exponent than (*,/,//,%)than(+,-)
#We can easily  solve above if these occur one time if above symbols occur 2 or many times than
#apply assosiative as if exponent occur 2 or more times than solve from right to left if (*,/,//,%)
#occur same time having same precedence than solve from left to right and similarly if (+,-)having
#same precedence occurs same time than solve from left to right mean 1st + than -
```

```
5
14
0.5
0
16
2
```