**1. What are the differences between operators and values in the following?**

**operators = \*, -, /, +**

**values = 'hello', -87.8, 6**

**Ans**. Operators: - refer to **special symbols (**\*, -, /, +) **that perform operations on values and variables**

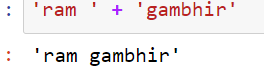
"value” it is the actual data it can be anything numbers, text, images ('hello', -87.8, 6) on which we can perform **some sort of computation or operations through operators.**

**Example: -**

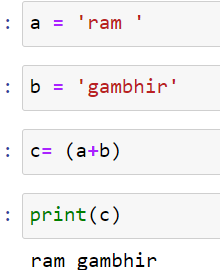
operators = \*, -, /, +

values = 'hello', -87.8, 6

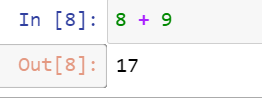
**operations on values**



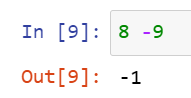
**operations on variables**



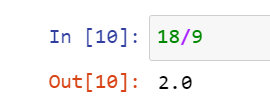
Example for addition we use +



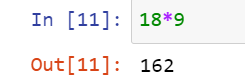
Example for subtraction we use -



Example for divide we use /



Example for multiplication we use \*

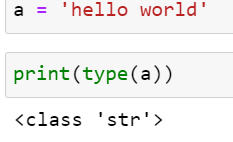


**2. What is the difference between string and variable?**

Ans: Variables: It acts like a container to store any data value.

String: String is a data type or type of value which we can store in a variable. Any number or text which are in single quote or double quote considered as String

Example :- in this a is a variable in which string value is stored ‘hello world”



**3. Describe three different data forms**.

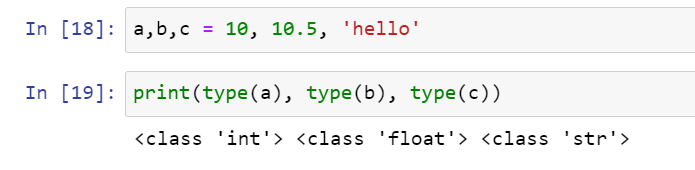
Ans : Three forms of data are :

int (interger):  integers are **zero, positive or negative whole numbers without a fractional part like 0,2,4,5,6,-1**

**float(float) : floats** are  **positive or negative numbers with a fractional part like 2.2, 3.4**

str(string) A string is a collection of one or more characters put in a single quote, double-quote or triple quote. Like ‘hello’

**example: with type function we can check data type.**

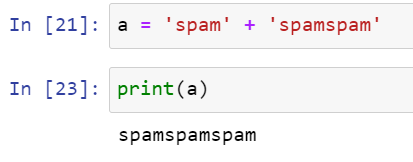


**4. What should the values of the following two terms be?**

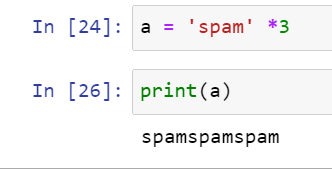
'spam' + 'spamspam'

'spam' \* 3

Ans . 'spam' + 'spamspam' it will concatenate both words like below



'spam' \* 3 : it will repeat the ‘spam’ word 3 times



**5. Why is it that data1 is a true variable name but 100 is not?**

**Ans. There are some rules for variable declaration**

* A variable name must start with a letter or the underscore character.
* A variable name cannot start with a number.
* A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and \_ )
* Variable names are case-sensitive (age, Age and AGE are three different variables)

**data1 is true variable because it started with a letter**

**100 is false variable or invalid variable because it started with a number and A variable name cannot start with a number**

**6. Which of the following three functions may be used to convert a value to an integer, a floating-point number, or a string?**

Ans. int() : it’s a function to covert a value to an integer

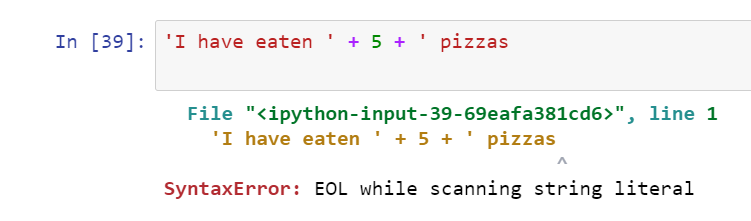
str() it’s a function to covert a value to a string

float() it’s a function to covert a value to a floating-point number

**7. What is the error caused by this expression? What would you do about it?**

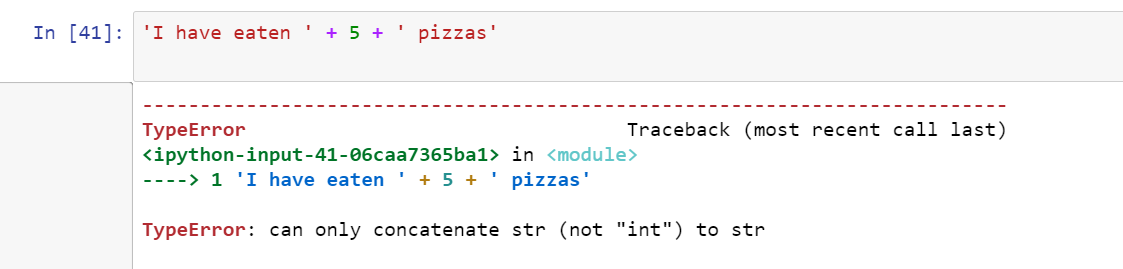
**'I have eaten ' + 5 + ' pizzas**

Ans. There are two error caused by this expression one is below:-



To resolve this we have to add ‘ at the end why because string should be in double quote here one quote is missing at the end

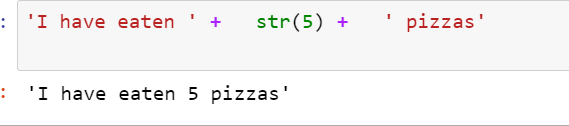
**When you resolve previous one after that you will get error again**



**why because we cannot concatenate string data dype with integer data type so here 5 should also be in string foramate. To do that we have multiple option we can convert 5 to string by str () funcation or by single or double quote’**



**or**



**8. What are the Boolean data type's two values? How do you go about writing them?**

**1. True**

**2. False**

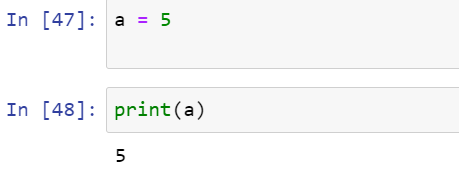
**T and F should be in capital and rest will be in lower case**

**9. Difference between the equal to and assignment operators?**

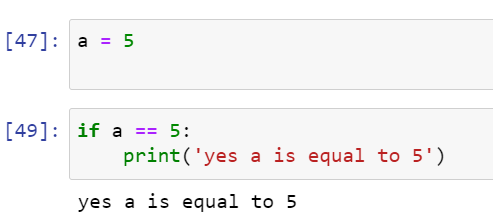
**= single equal to is use for assigning value (assignment operators)**

**== double equal to is use for comparison**

**(assignment operators)**



**comparison**



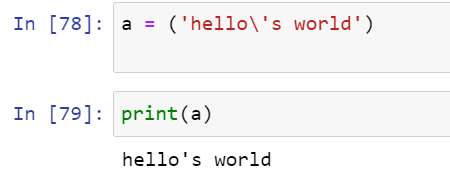
**10. Difference between python 2.x and 3.x**

* Python 3 syntax is simpler and easily understandable whereas Python 2 syntax is comparatively difficult to understand.
* Python 3 default storing of strings is Unicode whereas Python 2 stores need to define Unicode string value with “u.”

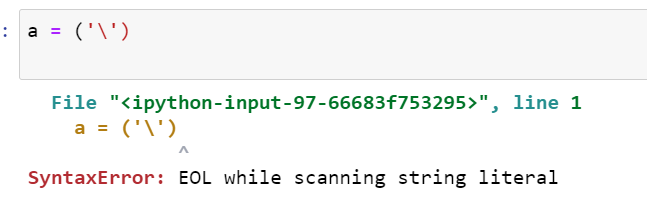
Python 3 rules of ordering comparisons are simplified whereas Python 2 rules of ordering comparison are complex.

11. How you will get “\” in python?

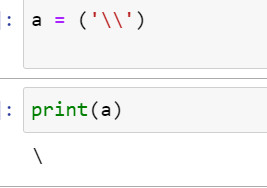
\ is use for to convert next point value as text



Here you can see we have used \ in statement to consider ‘ as a text. But if we want to print only \ can we print let see



We got an error because \ converted ‘ into text so to avoid that we will use “\\” here



12. What is a notation? How we can use it in Python?

the activity of representing something by a special system of marks or characters

Python's “**[:]**” notation, officially known as Slice Notation is used to extract the desired portion/slice from a given sequence

**Python's** slice **notation** is used to return a list or a portion of a list. The basic syntax is as follows: [start\_at:stop\_before:step].

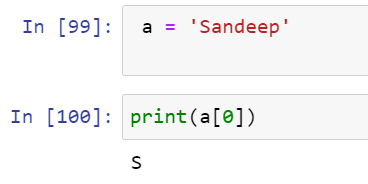
13. Explain indexing. Both positive and negative indexing.

Indexing sign is []

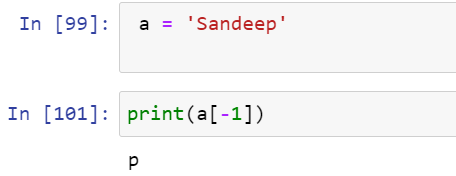
Indexing in Python is a way to refer the individual items within an iterable by its position.

Python can be accessed with positive or negative indexing

Positive Indexing starts from zero which denotes the first character and goes from left to right in increasing order.

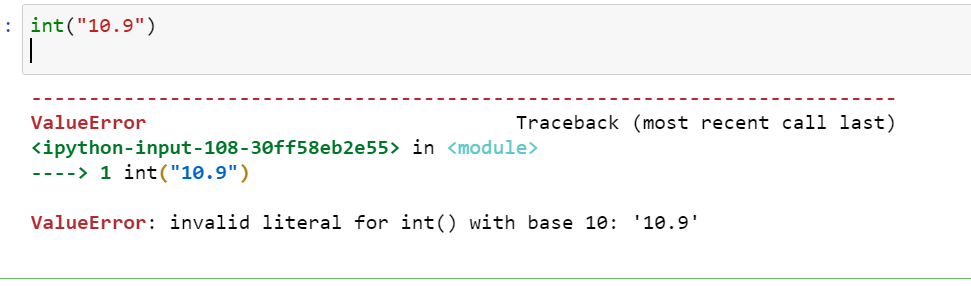


Negative Indexing starts with -1 which denotes the last character and goes from right to left in decreasing order.

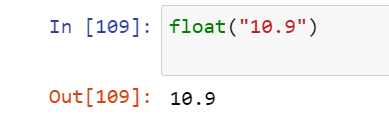


14. Will “int(“10.9”)” work in python?

No. it will give an error because we can convert only one datatype to another data type at a time. here we want two conversation at a time one is we want to covert string to int and at the same time we want float value to be converted in integer. So that’s not possible



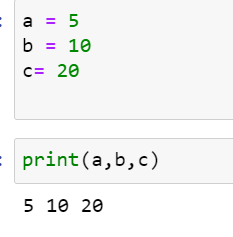
In below you can we conversion done because only single conversion is there string to float



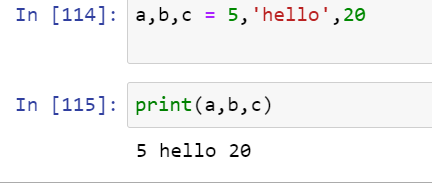
**15. Explain multivariable assignments.**

There are three ways to assign **multivariable**

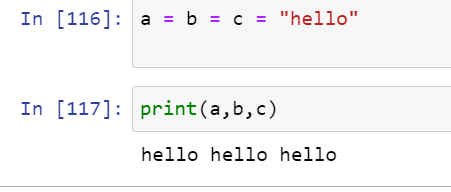
One way is to assign the values ​​separately in different lines.



Another way is to assign different values ​​to different variables on the same line.

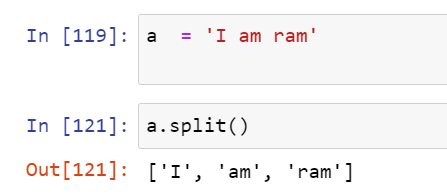


This is the third way when we want to store the same value in multiple variables.



**16. Explain Split function.**

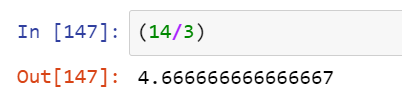
The split() method splits a string into a list. You can specify the separator, default separator is any whitespace.



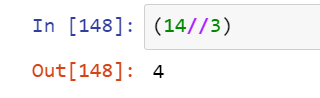
17. What is floor division. How it works different than integer function for negative and positive values?

floor division is an operation in Python that divides two numbers and rounds the result down to the nearest integer. The floor division happens via the double-backslash (//) operator.

**Normal Division**

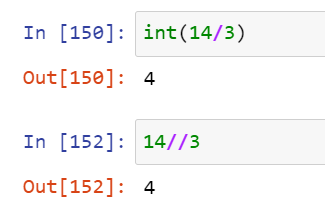


**Floor division**

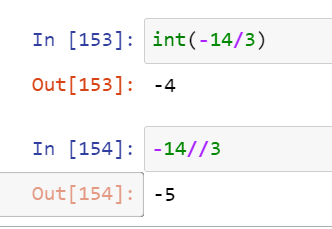


**different than integer function for negative and positive values?**

**For positive it will give same result**

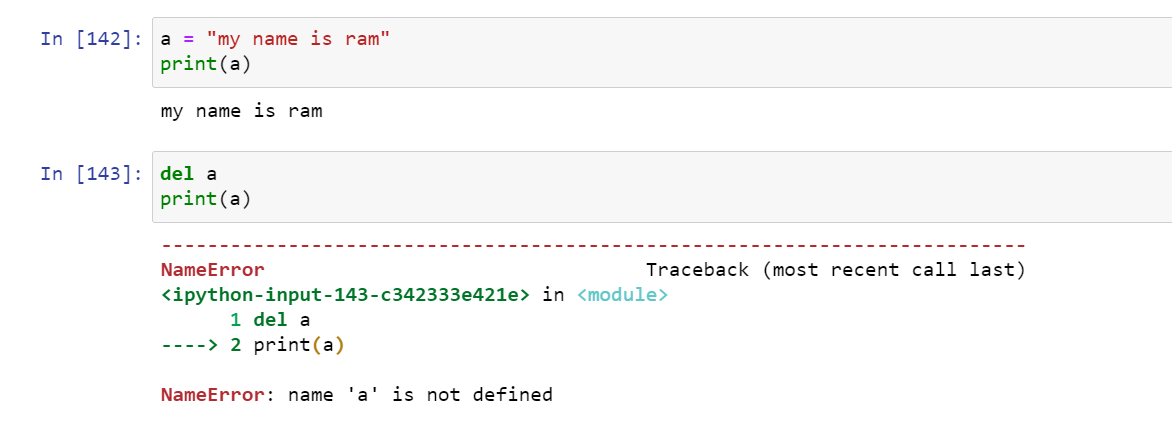


**For negative result will be different in floor division because floor division gives lower nearest value**



**18. How to delete a variable in Python?**

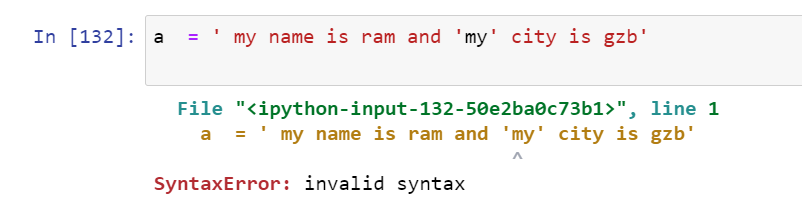
**With del() we can delete variable in python**



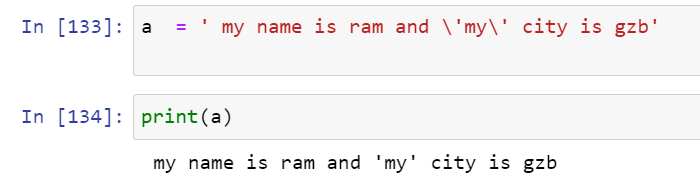
**19. Explain escape characters.**

In Python strings, **the backslash "\"** is a special character, also called the "escape" character. It is used in representing certain whitespace characters: "\t" is a tab, "\n" is a newline, and "\r" is a carriage return.

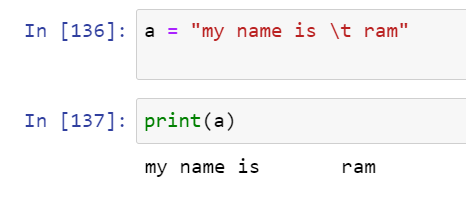
You will get an error if you use single quotes inside a string that is surrounded by single quotes:



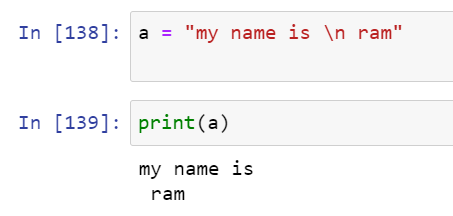
**To avoid this we will use \**



**\t for tab**

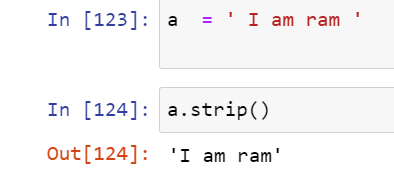


**\n for new line**



**20. How strip function works.**

**strip()** method removes characters or spaces from both left and right by default it will remove extra spaces from both sides.



**In a same way there two more sub part of strip function**

**lstrip()** method removes characters or spaces from left side only

**rstrip()** method removes characters or spaces from right side only