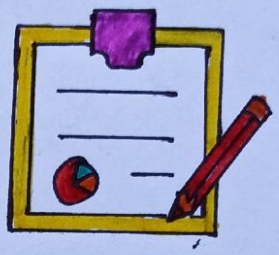
PYTHON NOTES
VARIABLES & DATATYPE



Python Variables :

Variable is a name that is used to refer to memory location. Python is also known as an identifier and used to hold value.

In python, we don't need to specify the type of variable python is a infer and smart enough to get variable type.

example : x = 10
name = "Atul Kumar"
value = 12.2

identifier:

Variable are the example of identifier. An identifier is used to identify the literals used in the program.

Rules to name an identifier —

- The first character of the variable must be alphabet or underscore (_).
- All character except first character may be an alphabe (A-z), (a-z) or digit (0-9).

Continue →

- Identifier name must not contain any white space or special character (!, @, #, %.)

- Identifier name not similar to any keyword defined in language.

Valid vs invalid identifier

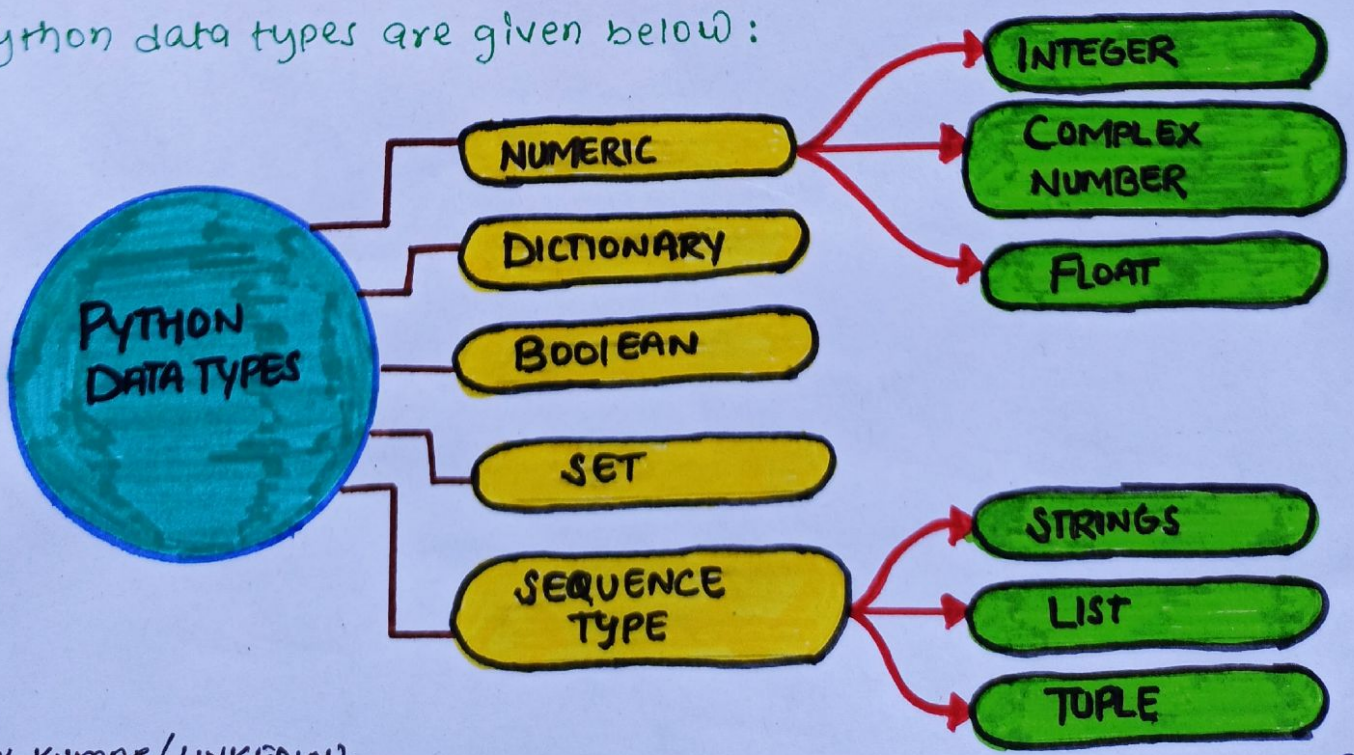
Valid	Invalid
• abc123	• 123abc
• abc_de	• abc@
• _abc	• 123
• ABC	• for
• abc	

Python Datatype

A variable can hold different type of value. for example name is must be stored in string where as id is stored in integer.

Variable can store data of different types, and different types can different thing.

Python data types are given below:



Operators in Python

The operator can be defined as a symbol which is responsible for particular operation between two operand.

Operator are used to perform operations on variable and value.

Python divides operator in following Groups:

- Arithmetic operator
- Comparison operator
- identity operator
- Membership operator.
- Assignment operator.
- logical operator
- Bitwise operator

Arithmetic Operator

Arithmetic operators are used to perform common mathematical operation.

Operator	name	example
+	addition	x + y.
-	Subtraction	x - y
*	Multiplication	x * y
/	Division	x / y
%	Modul	x % y
* *	exponentiation	x ** y
//	floor division	x // y

Assignment Operator

assignment operator are used to assign value to variable.

Operator	example	same as
=	x = 5	x = 5
+=	x += 1	x = x + 1
-=	x -= 1	x = x - 1
*=	x *= 1	x = x * 1
/=	x /= 1	x = x / 1
%=	x %= 1	x = x % 1
//=	x //= 1	x = x // 1
**=	x **= 1	x = x ** 1
&=	x &= 1	x = x & 1
|=	x |= 1	x = x | 1

Comparison Operator

Operator	Description	example
==	equal	x == y
!=	not equal	x != y
>	greater than	x > y
<	less than	x < y
>=	Greater/ equal	x >= y
<=	less/ equal	x <= y

logical Operator

Operator	Description	example
and	Returns true if both statement are true	x<5 and x<10
Or	Returns true if one of statement is true	x<5 or x<4
not	Reverse the result, Returns false if the result is true.	not(x>5 and x>10)

Python identity Operator

Identity Operator are used to compare the object, not if they are equal. but if they are actually same object with the same memory location.

Operator	Description	example
is	Return True if both variables are the same object.	x is y
is not	Returns true if both are not same object.	x is not y.

Python membership Operator.

Membership operator are used to test if a sequence is pretended is an object.

Operator	Description	example
in	Return True if a sequence with the specified value is present.	x in y
not in	Return True if a sequance with the specified value is not present.	x not in y.

Python Bitwise operator

Bitwise operator perform bit by bit operation on the values of two operand. Consider the following table.

Continue ⟶

→

Operator	name	Description
&	AND	Set each bit to 1 if both side are 1
|	OR	Set each bit to 1 if one of two bit is 1.
^	XOR	Set each bit to 1 if only one of two bits is 1.
~	NOT	Inverts all the bits
<<	Zero fill left shift	Shift left by pushing zeros in from the right and let the leftmost bits fall off.
>>	Signed right shift	Shift right by pushing copies of the leftmost bit in from the left, and let the rightmost bits fall off.

Input in Python

python input() function

Python input function is used to get input from user. it prompts for the user input and read a line. After reading data. it converts it into a string and returns that. it throws an error E of error is EOF is read.

Parameter

prompt : it is a string message with prompt for user input.

Return

it returns user input after converting into string.

let see an example of Input() function to understand its functionality.

Continue →

example:

Input

```
# Python Input() function.
val = Input("enter a value :")
print ("you enrered :", val)
```

Output

```
enter a value : 20
you entered : 20
```