Variable:

Reserved memory location to store values. Every value in Python has a datatype.

- **Rules**:
    1. Variable name starts with a letter or the underscore character
    2. Variable name cannot start with a number
    3. Variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and _)
    4. Case-sensitive (NAME & Name)
    5. Reserved words can not be used the name of the variable.

Datatype:

1. Number – Integer, Float
2. String – String Characters
3. Sequence – List, Tuple, Range
4. Dictionary
5. Set
6. None

- **Format**:

Datatypes	Format
Number - Integer	X=25
Number – Float	X=25.2
String	X=” Hello”
List	X= [“one”,” two”, “three”]
Tuple	X= (“one”,” two”, “three”)
Range	X=range (10)
Dictionary	X= {“name”: “raj”, “age”:10}
Set	X= {“one”,” two”, “three”}

- **Print**:

  print (“The type of Variable”, X, “Format:”, type(X))

- **Delete**:

  del <Variable> (Example: del X)

Operators:

Special symbols that perform operations on manipulate the values.

- **Types**:

Serial	Operators
1	Arithmetic Operators
2	Comparison or Relation Operator
3	Assignment Operator
4	Logical Operator
5	Bitwise Operator
6	Membership Operator
7	Identity Operator

- **Arithmetic Operator (Mathematical operations)**:

Types	Addition (+), Subtraction (-), Multiplication (*), Division (/), Modulus (%), Exponent (**), Floor (//)
Syntax	**Addition**: a + b, **Subtraction**: a – b, **Multiplication**: a * b, **Division**: a / b, **Modulus**: a % b (returns the remainder), **Exponent:** a ** b (returns first raised to power second), **Floor:** a // b (carries out integer type)

- **Comparison Operator (Compare the Values – True/False)**:

Types	Greater than (>), Less than (<), Equal (==), Not Equal (! =), Greater than or Equal (>=), Less than or Equal (<=)
Syntax	**Greater than**: a > b, **less than:** a < b, **Equal:** a == b, **Not Equal**: a! = b, **Greater than or Equal:** a >=b, **Less than or Equal:** a <=b

- **Assignment Operator (Assign the values to the variable)**:

Types	Equal (=), Addition (+=), Subtraction (-=), Multiplication (*=), Division (/=), Modulus (%=), Exponent (**=), Floor (//=)
Syntax	**Equal**: a= a + b, **Addition:** a+=b, **Subtraction:** a-=b, **Multiplication:** a*=b, **Division:** a/=b, **Modulus:** a%=b, **Exponent:** a**=b, **Floor:** a//=b **Note:** a+=b Meaning a= a + b (same as)

- **Logical Operator (Logical Operations)**:

Types	Logical AND (and), Logical OR (or), Logical NOT (not)
Syntax	**Logical AND**: a and b, **Logical OR**: a or b, **Logical NOT:** not a

- **Membership Operator (value: sequence or not)**:

Types	in, not in
Syntax	**in:** a in b, **not in:** a not in b

- **Identity Operator (compare two objects)**:

Types	is, is not
Syntax	**is:** a is b, **is not:** a not is b

Decision Making (Conditional Statement):

A block of statements has to execute or not based on a conditional.

- **Types**:

Serial	Statements
1	If Statement
2	Else Statement
3	Nested if Statement

- **If Statement:**

Syntax	if <conditional>: print
Example	if a>=10: print (“Greater than 10 “)

- **Else Statement:**

Syntax	if <conditional>:
	  print
	else:
	  print

Example	if a>=10: print (“Greater than 10 “)

	else: print (“No Value”)

- **Nested If Statement:**

Syntax	if <conditional>:
	  if <conditional>:
	  print (“ “)
	  else:
	  print (“ “)
	else:
	  print (“ “)

Example	if a>=10:
	  if a==0:
	  print (“No”)
	  else:
	  print (“Value”)
	else:
	  print (“Nil”)