

Variable Declaration

Name = 'vipul'

Age =27

Salary = 1000.0

Print(name)

Print(age)

Print(salary)

Variable declaration in single line

Ename, Eage, Esalary = "vipul",27,1000.0

Assigning Multiple variable same value

Vipul=Rahul=govind="BLUE"

Five Data Types in Python

1. Numbers
2. String
3. List
4. Tuple
5. Dictionary

Check Data Types

type(age)

type(salary)

type(name)

Operations on Numbers

V1=12

V2=10

Result=v1+v2

Result=v1-v2

Resultrem=v1%v2 //modulus for remainder

Converting Data types

Salary=1000.0

int(Salary)

'or' intsal=int(Salary)

float(age)

Lists

it works like arrays

We can store multiple values on a single variable and each stored value will be at different index.

```
student=["vipul","amar","kunal","prabhu","ramesh"]
```

```
student[0]
```

```
student[1]
```

```
student.append("amit")          #to append
```

```
student[3]=gautam              #to replace
```

```
student.append("vipul")
```

```
student.count("vipul")          #returns 2 as list is having 2 vipul
```

```
student.remove("vipul")         #remove first vipul from left to right
```

```
student.sort()                  #will sort alphabetically
```

```
student.reverse()
```

```
student2=["pranay","ashish"]
```

```
totalstudent=student+student2   #concat the list
```

```
marks=[98,95,99,45]
```

```
max(marks)
```

```
min(marks)
```

Dictionary

Here **index** is not used but **key** is used to access a **value**

```
Student1={"Name":"Vipul","Age":27}
```

```
To check: Student1["Name"]
```

```
To update: Student1["Age"]=28
```

```
To add new info: Student1["College"]="IIT Delhi"
```

```
To delete a key: del student1["College"]
```

```
To find length: len(Student1)
```

```
To check type: type(Student1)
```

```
To Check keys: Student1.keys()
```

```
To Check values: Student1.values()
```

To update:

```
Student2={"College":"CMS","Email":"Hello@abc.com"}
```

```
Student1.update(Student2)
```

To clear: `student1.clear()`

To delete: `del Student1`

Tuple

Tuples are just like lists but they are immutable objects means you can't change its values. It will remain same.

```
tuple1=("Praveen","Rahul","Pranav",1,2,3)
```

```
tuple2()
```

To access: `tuple1[2]`

To slice: `tuple[0:2]`

To start with 1: `tuple[1:]`

*****You can update any value*****

To delete: `del tuple2`

To find length: `len(tuple1)`

To Concat:

```
tuple2=(4,3,2,6,7)
```

```
tuple3=tuple1+tuple2
```

To repeat: `tuple1*2`

To find: 2 in `tuple2`

Maximum value: `max(tuple2)`

Minimum value: `min(tuple2)`

Convere list into tuple: `tuple(list1)`

