In [20]:
```
#list
l1 = [10,1,30,0,5,10,100]
l2 = ['a','b','c']

l1.extend(l2)
print(l1)
```

```
[10, 1, 30, 0, 5, 10, 100, 'a', 'b', 'c']
```

In [4]:
```
#tuples
t1 = ("Ruparel","Python","Java",2023,True)
print(type(t1))
print(t1)
print(t1[1])
#del t1(1) #its can be generate error because tupel is immutable
print(t1)
```

```
<class 'tuple'>
('Ruparel', 'Python', 'Java', 2023, True)
Python
('Ruparel', 'Python', 'Java', 2023, True)
```

In [6]:
```
#set
a = {10,20,30,10,1,20,2,10,3}
b = {1,2,3,4,5}
print(type(a))
print(type(b))
print(a)
print(b)
print(a|b)#union
x = a.union(b)
print(x)
print(a&b)#intersection
x = a.intersection(b)
print(x)
print(a-b) #difference
print(b-a) #difference
```

```
<class 'set'>
<class 'set'>
{1, 2, 3, 20, 10, 30}
{1, 2, 3, 4, 5}
{1, 2, 3, 4, 5, 10, 20, 30}
{1, 2, 3, 4, 5, 10, 20, 30}
{1, 2, 3}
{1, 2, 3}
{10, 20, 30}
{4, 5}
```

In [11]:
```
s = {'R','u','p','a','r','e','l'}
print(s)
s.add('Z')
print(s)
s.discard('t')
print(s)
#s.remove('t') error generate
print(s)
s.pop()
print(s)
s.clear()
print(s)
s1 = s.copy()
print(s1)
```

```
{'R', 'l', 'e', 'u', 'a', 'p', 'r'}
{'R', 'l', 'e', 'u', 'a', 'p', 'r', 'Z'}
{'R', 'l', 'e', 'u', 'a', 'p', 'r', 'Z'}
{'R', 'l', 'e', 'u', 'a', 'p', 'r', 'Z'}
{'l', 'e', 'u', 'a', 'p', 'r', 'Z'}
set()
set()
```

In [ ]:
```

```