

# Set & Dictionary

# Set

- Sets are used to store multiple items in a single variable.
- A set is a collection which is unordered, unchangeable, and unindexed.
- Sets are written with curly brackets.
- Note: Set items are unchangeable, but you can remove items and add new items.

```
s1= {1,2,3,3,3,4,5}  
print(len(s1))  
print(type(s1))  
print(s1)
```

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

# Copy , Clear and Delete set

```
s2= {1,2,3,3,3,4,5}
s3=s2.copy()
print(s3)
s3.clear()
print(s3)
```

```
{1, 2, 3, 4, 5}
set()
```

```
s4= {1,2,3,3,3,4,5}
print(s4)
del s4
print(s4)
```

```
{1, 2, 3, 4, 5}
```

```
-----
NameError                                Tr
/tmp/ipykernel_33/3353354822.py in <module>
      2 print(s4)
      3 del s4
----> 4 print(s4)

NameError: name 's4' is not defined
```

# Add and Update

---

```
s5= {"a", "b", "c"}  
print(s5)  
s5.add("o")  
print(s5)
```

{'a', 'c', 'b'}

{'a', 'c', 'o', 'b'}

```
s5= {"a", "b", "c"}  
s6={"e", "f", "g"}  
s5.update(s6)  
print(s5)
```

{'g', 'a', 'b', 'e', 'f', 'c'}

## Remove and discard

```
s5= {"a", "b", "c"}  
s5.remove("B")  
print(s5)
```

```
-----  
KeyError                                Traceback (most recent call last)  
/tmp/ipykernel_33/2102676676.py in <module>  
      1 s5= {"a","b","c"}  
----> 2 s5.remove("B")  
      3 print(s5)  
  
KeyError: 'B'
```

```
s5= {"a", "b", "c"}  
s5.discard("B")  
print(s5)
```

```
{'a', 'c', 'b'}
```

```
# remove  
s5= {"a", "b", "c"}  
s5.remove("b")  
print(s5)
```

```
{'a', 'c'}
```

# Dictionary

```
d1={"name":"pallavi",  
    "roll": 25,"mobile":25  
}  
print(d1)  
print(type(d1))  
print(len(d1))
```

```
{'name': 'pallavi', 'roll': 25, 'mobile': 25}  
<class 'dict'>  
3
```

- Dictionaries are used to store data values in key : value pairs.
- A dictionary is a collection which is ordered, changeable and do not allow duplicates.
- Dictionaries are written with curly brackets, and have keys and values

# Accessing Items

```
x=d1["name"]  
print(x)
```

pallavi

```
x=d1.get("name")  
print(x)
```

pallavi

- You can access the items of a dictionary by referring to its key name, inside square brackets
- There is also a method called get() that will give you the same result

## Get Keys

```
# for keys  
x=d1.keys()  
print(x)
```

```
dict_keys(['name', 'roll', 'mobile'])
```

- The keys() method will return a list of all the keys in the dictionary.



## Get Values

```
# for values  
x=d1.values()  
print(x)
```

```
dict_values(['pallavi', 25, 25])
```

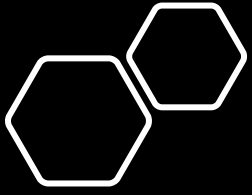
- The values() method will return a list of all the values in the dictionary.

## Get Items

```
# items  
x=d1.items()  
print(x)
```

```
dict_items([('name', 'pallavi'), ('roll', 25), ('mobile', 25)])
```

- The items() method will return each item in a dictionary, as tuples in a list



# Change Values

- You can change the value of a specific item by referring to its key name

```
# changing value  
d1["name"]="saumya"  
print(d1)
```

```
{'name': 'saumya', 'roll': 25, 'mobile': 25}
```

```
# changing/ updation  
d1["color"]="saumya"  
print(d1)
```

```
{'name': 'saumya', 'roll': 25, 'mobile': 25, 'color': 'saumya'}
```