- Seat Number (Seat Type first Character + random number between (1,5000))
- Booking id (first three characters of btype + Name first three characters + @uppal + 2024)
- Booking Type
- Discount Amount
- Booking Amount.

In [18]:

```
x="dedication"
s=list(x)
print(s)

for i in range(len(s)):
    if s[i]!="*":
        count=1
        for j in range(i+1,len(s)):
            if s[i]==s[j]:
                count+=1
                s[j]="*"
        print(s[i],"=",count)
```

executed in 11ms, finished 15:23:27 2024-08-21

```
['d', 'e', 'd', 'i', 'c', 'a', 't', 'i', 'o', 'n']
d = 2
e = 1
i = 2
c = 1
a = 1
t = 1
o = 1
n = 1
```

Dictionary

- Dictionary is also one of the predefined class and treated as 'dict' data type.
- Dictionary stores the value in the form of key and value pair.
- In keys and values , the value of key must be unique and value of values may or may not be unique.
- Keys must be immutable whereas values may or may not be immutable.

In [30]:

```
d={"name":"sham","age":25,"loc1":"hyderabad","loc":["hyderabad","chennai"],10:

print(d['name'])
d['role']="SDE"
```

executed in 9ms, finished 15:34:53 2024-08-21

```
sham
```

In [31]:

```
d
```

executed in 14ms, finished 15:34:55 2024-08-21

Out[31]:

```
{'name': 'sham',
 'age': 25,
 'loc1': 'hyderabad',
 'loc': ['hyderabad', 'chennai'],
 10: 'tuple',
 'role': 'SDE'}
```

Inbuilt Functions

1.clear()

In [69]:

```
d1={1:2,2:3,3:4,4:5,5:6}

d1.clear()
```

executed in 7ms, finished 15:53:46 2024-08-21

In [70]:

```
d1
```

executed in 10ms, finished 15:53:48 2024-08-21

Out[70]:

```
{}
```

2.popitem()

In [73]:

```
d1={1:2,2:3,3:4,4:5,5:6}

print(d1.popitem())
print(d1.popitem())
print(d1.popitem())
print(d1.popitem())
print(d1.popitem())
print(d1.popitem())
```

executed in 34ms, finished 15:54:28 2024-08-21

```
(5, 6)
(4, 5)
(3, 4)
(2, 3)
(1, 2)

---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
Cell In[73], line 8
      6 print(d1.popitem())
      7 print(d1.popitem())
----> 8 print(d1.popitem())

KeyError: 'popitem(): dictionary is empty'
```

3.pop()

In [74]:
```
d1={1:2,2:3,3:4,4:5,5:6}
d1.pop(4)
```

executed in 10ms, finished 15:54:48 2024-08-21

Out[74]:
```
5
```

In [75]:
```
print(d1)
```

executed in 6ms, finished 15:54:53 2024-08-21

```
{1: 2, 2: 3, 3: 4, 5: 6}
```

In [43]:
```
d1.pop(7)
```

executed in 30ms, finished 15:40:18 2024-08-21

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
Cell In[43], line 1
----> 1 d1.pop(7)

KeyError: 7
```

4.get()

In [76]:
```
d1={1:2,2:3,3:4,4:5,5:6}
print(d1.get(1))
print(d1.get(5))
print(d1.get(100))
print(d1[2])
```

executed in 9ms, finished 15:55:26 2024-08-21

```
2
6
None
3
```

5.copy()

In [55]:

```
d1={1:2,2:3,3:4,4:5,5:6}

d2=d1.copy()  # shallow copy

d2["name"]="social"
print(d1)
print(d2)
print(id(d1))
print(id(d2))
```

executed in 9ms, finished 15:44:59 2024-08-21

```
{1: 2, 2: 3, 3: 4, 4: 5, 5: 6}
{1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 'name': 'social'}
2324310253312
2324310253632
```

In [56]:

```
d3=d1

d3["name"]="social"  # Deep copy
print(d1)
print(d3)

print(id(d1))
print(id(d3))
```

executed in 8ms, finished 15:45:18 2024-08-21

```
{1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 'name': 'social'}
{1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 'name': 'social'}
2324310253312
2324310253312
```

6.keys()

In [57]:

```
d1={1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 'name': 'social'}

print(d1.keys())
```

executed in 8ms, finished 15:45:58 2024-08-21

```
dict_keys([1, 2, 3, 4, 5, 'name'])
```

7.values()

In [58]:

```
print(d1.values())
```

executed in 8ms, finished 15:46:14 2024-08-21

```
dict_values([2, 3, 4, 5, 6, 'social'])
```

8.items()

In [59]:

```python
print(d1.items())
```

executed in 6ms, finished 15:46:32 2024-08-21

```
dict_items([(1, 2), (2, 3), (3, 4), (4, 5), (5, 6), ('name', 'social')])
```

In [78]:

```python
d1={1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 'name': 'social'}

for key,val in d1.items():
    print(key,":",val)
```

executed in 8ms, finished 15:57:48 2024-08-21

```
1 : 2
2 : 3
3 : 4
4 : 5
5 : 6
name : social
```

9.update()

In [79]:

```python
d1={1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 'name': 'social'}
d2={"loc":"mumbai"}

d2.update(d1)
```

executed in 7ms, finished 15:58:06 2024-08-21

In [80]:

```python
print(d1)
```

executed in 5ms, finished 15:58:08 2024-08-21

```
{1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 'name': 'social'}
```

In [81]:

```python
print(d2)
```

executed in 5ms, finished 15:58:12 2024-08-21

```
{'loc': 'mumbai', 1: 2, 2: 3, 3: 4, 4: 5, 5: 6, 'name': 'social'}
```

In [ ]:

```python

```

WAP to find the missing values from the given list

```
l=[1,10,5,2,7,9,15,11]
```

In [86]:

```
l=[1,10,5,2,7,9,15,11]
l.sort()
print(l)
mini=min(l)
maxi=max(l)

missing_values=[]

for i in range(mini,maxi+1):
    if i not in l:
        missing_values.append(i)

print(missing_values)
```

executed in 9ms, finished 16:03:43 2024-08-21

```
[1, 2, 5, 7, 9, 10, 11, 15]
[3, 4, 6, 8, 12, 13, 14]
```

In [ ]:

In [89]:

```
menu={
    "Starters":["chicken 65","gobi manhurian","mushroom","chicken tikka","pane
        "Main":["chicken biryani","meals","Mutton biryani","prawns biryani","v
     "Desserts":["vannila icecream","double ka meetha","cold drinks","chocolat
}
```

executed in 5ms, finished 16:13:20 2024-08-21

In [90]:

```
print(menu)
```

executed in 6ms, finished 16:13:25 2024-08-21

```
{'Starters': ['chicken 65', 'gobi manhurian', 'mushroom', 'chicken tikka', 'p
aneer tikka'], 'Main': ['chicken biryani', 'meals', 'Mutton biryani', 'prawns
biryani', 'veg biryani'], 'Desserts': ['vannila icecream', 'double ka meeth
a', 'cold drinks', 'chocolate icecream', 'butterscoth icecream']}
```

In [92]:

```
import pandas as pd
```

executed in 5ms, finished 16:13:59 2024-08-21

In [93]:

```
s=pd.Series(menu)
```

executed in 11ms, finished 16:14:16 2024-08-21

In [94]:

```
s
```

executed in 19ms, finished 16:14:19 2024-08-21

Out[94]:

```
Starters    [chicken 65, gobi manhurian, mushroom, chicken...
Main        [chicken biryani, meals, Mutton biryani, prawn...
Desserts    [vannila icecream, double ka meetha, cold drin...
dtype: object
```

In [95]:

```
x=pd.DataFrame(menu)
```

executed in 6ms, finished 16:14:48 2024-08-21

In [96]:

```
x
```

executed in 21ms, finished 16:14:49 2024-08-21

Out[96]:

	Starters	Main	Desserts
0	chicken 65	chicken biryani	vannila icecream
1	gobi manhurian	meals	double ka meetha
2	mushroom	Mutton biryani	cold drinks
3	chicken tikka	prawns biryani	chocolate icecream
4	paneer tikka	veg biryani	butterscoth icecream

49. input --> a="coding"
output --> dict1={'C': 'ccc', 'O': 'ooo', 'D': 'ddd', 'I': 'iii', 'N': 'nnn', 'G': 'ggg'}

50. input --> (keys)pop = [30.55, 2.77, 39.21] (values)countries = ["afghanistan", "albania", "algeria"]

output --> {30.55: 'afghanistan', 2.77: 'albania', 39.21: 'algeria'}

In [98]:

```
a="coding"

d={}

for i in a:
    d[i.upper()]=i*3

print(d)
```

executed in 7ms, finished 16:24:14 2024-08-21

```
{'C': 'ccc', 'O': 'ooo', 'D': 'ddd', 'I': 'iii', 'N': 'nnn', 'G': 'ggg'}
```

In [100]:

```
pop = [30.55, 2.77, 39.21]
countries = ["afghanistan", "albania", "algeria"]

d={}

for i,j in zip(pop,countries):
    d[i]=j

print(d)
```

executed in 8ms, finished 16:26:07 2024-08-21

```
{30.55: 'afghanistan', 2.77: 'albania', 39.21: 'algeria'}
```

Q) Find the maximum value key:

dict={"hp":4,"toshiba":7,"macbook":10,"dell":8,"lenovo":9}

In [109]:

```
d1={"hp":4,"toshiba":7,"macbook":10,"dell":8,"lenovo":9}

key=list(d1.keys())
print(key)

value=list(d1.values())
print(value)

maxi=max(value)

idx=value.index(maxi)
print(idx)

print(key[idx])
```

executed in 8ms, finished 16:39:37 2024-08-21

```
['hp', 'toshiba', 'macbook', 'dell', 'lenovo']
[4, 7, 10, 8, 9]
2
macbook
```

In [113]:

```
d1={"hp":4,"toshiba":7,"macbook":10,"dell":8,"lenovo":9}

max_val=0
max_key=""

for key,val in d1.items():
     if val>max_val:
        max_val=val
        max_key=key

print(max_key)
```

executed in 7ms, finished 16:42:34 2024-08-21

```
macbook
```

In [ ]: