In [2]:
```
s1='can can you canner can not can you buy'
```

In [3]:
```
l=s1.split()
l
```

Out[3]:
```
['can', 'can', 'you', 'canner', 'can', 'not', 'can', 'you', 'buy']
```

In [7]:
```
for word in l:
    print(word,":",l.count(word))
```

```
can : 4
can : 4
you : 2
canner : 1
can : 4
not : 1
can : 4
you : 2
buy : 1
```

word-frequency

In [10]:
```
s1='can can you canner can not can you buy'
list_words=s1.split()
l1=[]
for word in list_words:
    if word not in l1:
        print(word,list_words.count(word))
        l1.append(word)
#step1
#can     can not in []       true  5 ['can']
#can     can not in ['can']   False xxxxxxx
#you   you not in ['can']     true 2 ['can','you']
```

```
can 4
you 2
canner 1
not 1
buy 1
```

Insert

In [11]:
```
l1=[1,2,3,4,100]
l1.insert(3,'a')
l1

# 1 2 3 'a' 4 100
```

Out[11]: [1, 2, 3, 'a', 4, 100]

Append

In [15]:
```
l1=['hyd','Mumbal','Chennai','Bengaluru']
#l2=['Mumbai','Chennai','Bengaluru']
l2=[]
for i in l1:
    if len(i)>5:
        print(i)
        l2.append(i)
l2
```

```
Mumbal
Chennai
Bengaluru
```

Out[15]: ['Mumbal', 'Chennai', 'Bengaluru']

In [20]:
```
#create a list of sqares of 1 to 10 numbers
sqr=[]
for i in range(1,11):
    print(i*i)
    sqr.append(i*i)
sqr
```

```
1
4
9
16
25
36
49
64
81
100
```

Out[20]: [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]

List - Comprahenshion

pattern - i

In [ ]:
```
 sqr=[]
for i in range(1,11):
    print(i*i)
    sqr.append(i*i)
sqr
```

In [ ]:
```
num=[<output><forloop with out:>]
```

In [21]:
```
num=[i*i for i in range(1,11)]
num
```

Out[21]:
```
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

In [30]:
```
# l1=['hyd','mumbai']
# l2=[]
# for i in l1:
#     l2.append(i.capitalize())

l1=['hyd','mumbai']
l2=[i.capitalize() for i in l1 ]
l2
```

Out[30]:
```
['Hyd', 'Mumbai']
```

In [ ]:
```

```

Pattern-2

for if

In [27]:
```
l1= ['hyd','Mumbai','Chennai','Bengaluru']
l2=[]
for i in l1:
    if len(i)>5:
        l2.append(i)
l2
```

Out[27]:
```
['Mumbai', 'Chennai', 'Bengaluru']
```

In [ ]:
```
l2=[<output><forloop><if codition>]
```

In [ ]:
```
l1= ['hyd','Mumbai','Chennai','Bengaluru']
l2=[]
```

In [ ]:
```
```

$Pattern - 3$:

$for - if - else$

In [ ]:
```
[<if output><if condition> else <else output> <for loop>]
```

In [31]:
```
#l1=[<output><forloop with out:>]
#l2=[<output><forloop><if codition>]

l=[]
for i in range(1,10):
    if i%2==0:
        l.append(f'even {i}')
    else:
        l.append(f'odd {i}')
l
[<if output> if <if condition> else <else output> <for loop>]
```

Out[31]:
```
['odd 1',
 'even 2',
 'odd 3',
 'even 4',
 'odd 5',
 'even 6',
 'odd 7',
 'even 8',
 'odd 9']
```

In [33]:
```
[f'even {i}'if i%2==0 else f'odd {i}' for i in range(1,10) ]
```

Out[33]:
```
['odd 1',
 'even 2',
 'odd 3',
 'even 4',
 'odd 5',
 'even 6',
 'odd 7',
 'even 8',
 'odd 9']
```

In [ ]:
```
Strings
List
sat sunday = tuple

- how to read the tuple
- different ways to read tuple
- type len max min sum reversed sorted
- concatenation
- in
- index
- differnce between in and index range using for loop
- mutaable immutable concept
- slice
- part 2
- methods
```

In [34]:
```
t1=(1,2,3)
type(t1)
```

Out[34]:
```
tuple
```

In [35]:
```
dir(t1)
```

Out[35]:
```
['__add__',
 '__class__',
 '__class_getitem__',
 '__contains__',
 '__delattr__',
 '__dir__',
 '__doc__',
 '__eq__',
 '__format__',
 '__ge__',
 '__getattribute__',
 '__getitem__',
 '__getnewargs__',
 '__gt__',
 '__hash__',
 '__init__',
 '__init_subclass__',
 '__iter__',
 '__le__',
 '__len__',
 '__lt__',
 '__mul__',
 '__ne__',
 '__new__',
 '__reduce__',
 '__reduce_ex__',
 '__repr__',
 '__rmul__',
 '__setattr__',
 '__sizeof__',
 '__str__',
 '__subclasshook__',
 'count',
 'index']
```

In [ ]:
```
- lambda functions
- dictionary
- file handeling session

Thursdday 15/02/2024 python will complete

1 week stats part 1
1 week EDA - exploraoty data analysis

write a normal code without syntax error === logic
```