CHAPTER-05

LIST INTRODUCTION:

```
# list--> ordered collection of items.

###
numbers=[1,2,3,4]
print(numbers)
numbers=[1,2,3,4]
print(numbers[1])

###
words=["abrar","haider"]
print(words)
words=["abrar","haider"]
print(words[:2])

###
mixed=[1,2,3,4,"Abrar","Haider",None,]
print(mixed)
mixed=[1,2,3,4,"Abrar","Haider",None,]
print(mixed[-1])
mixed[1]="araf"
print(mixed)
mixed[4:]=["three","four"]
print(mixed)
```

ADD DATA TO LIST:

```
# how to add items to a list.
fruits=['grapes','apple']
fruits.append('mango')
print(fruits)
fruits=[]
fruits.append('mango')
fruits.append('apple')
fruits.append('grapes')
print(fruits)
```

DELETE DATA FROM LIST:

```
# Pop Method:
fruits=['orange','apple','pear','banana','kiwi']
fruits.pop()
print(fruits)

###
fruits=['orange','apple','pear','banana','kiwi']
fruits.pop(1) # position 1 item will be deleted
print(fruits)

# Delete operator:
fruits=['orange','apple','pear','banana','kiwi']
del fruits[1]

# Remove Method:
fruits=['orange','apple','pear','banana','kiwi']
fruits.remove('banana')
print(fruits)


# Adding Method: append, extend, insert.
# Deleting Method: pop, remove, del.
```

IN KEYWORD:

```
# Check item in the list

fruits=['orange','apple','pear','banana','kiwi']
if 'apple' in fruits:
    print('apple is present')
else:
    print('not present')
```

MORE METHODS:

```
# Count Method:
fruits=['orange','apple','pear','banana','apple','kiwi']
print(fruits.count('apple'))

# Sort Method:
fruits=['orange','apple','pear','banana','apple','kiwi']
fruits.sort() #will sort alphabetically
print(fruits)

num=[2,4,1,3]
num.sort()
print(num)

# Sorted function:
num=[2,4,1,3]
print(sorted(num))

# Clear Method:
num=[2,4,1,3]
num.clear()
print(num)

# Copy Method:
num=[2,4,1,3]
num_1=num.copy()
print(num_1)
```

IS VS EQUALS:

```
# List comparison:

fruits_1=['orange','apple','pear','banana','apple','kiwi']

fruits_2=['orange','pear','banana','apple']

print(fruits_1==fruits_2)
fruits_3=['orange','pear','banana','apple']
fruits_4=['orange','pear','banana','apple']
print(fruits_3==fruits_4) # values are same
print(fruits_3 is fruits_4) # is used to check whether the objects are in same memory location
```

JOIN AND SPLIT:

```
# Split Method:
user_info='harshit 24'.split() # will split from spaces
print(user_info)

###
user_info='harshit,24'.split(",") # will split from spaces
print(user_info)

###
name,age='harshit 24'.split() # will split from spaces
print(name)
print(age)

# Join Method:
user_info=["harshit","24"] #must be in string
print(",".join(user_info))
```

LIST VS ARRAYS:

```
# List vs array
# c , c++, java
# array int, string
# List - store any data / flexible
# python array module - fix data type
# numpy arrays - binding with c Libraries
# javascript array = python list / flexible
```

LIST VS STRINGS:

```
# In case of string: Immutable
s="abrar"
t=s.title()
print(t)
print(s) # string in s variable will not change

# In case of list: Mutable
l=['word1','word2','word3']
l.pop()
print(l) # will chane original list
```

LOOPING IN LIST:

```
# For loop:
fruits=['orange','apple','pear','banana','kiwi']
for fruit in fruits:
    print(fruit)

# While loop:
fruits=['orange','apple','pear','banana','kiwi']
i=0
while i<len(fruits):
    print(fruits[i])
    i+=1
```

LIST INSIDE LIST:

```
matrix=[[1,2,3],[4,5,6],[7,8,9]] # 2D list
for sublist in matrix:
    for i in sublist:
        print(i)

###
print(matrix[1][1]) # print any position value in the matrix

# Type funcion: To find the type of data.
print(type(matrix))
```

MORE ABOUT LIST:

```
# Generate lists with range functions:
num=list(range(1,11))
print(num)

# Something more about Pop Method:
num=list(range(1,11))
popped_item=num.pop()
print(num)
print(popped_item)

# Index Method:
num=list(range(1,11))
position=num.index(4)
print(position)
###
num=[1,2,3,4,5,6,7,8,9]
position=num.index(4)
print(position)
###
num=[1,2,3,4,5,6,7,8,9,1]
position=num.index(1)
print(position) # Just find the first 1
###
```

```
num=[1,2,3,4,5,6,7,8,9,1]
position=num.index(1,3) # Will start finding from position 3
print(position)
###
num=[1,2,3,4,5,6,7,8,9,1]
position=num.index(1,3,10) # Will start finding from position 3 to 9
print(position)

# Pass list to a function:
num=[1,2,3,4,5,6,7,8,9,1]
def negative_list(list):
    negative=[]
    for i in list:
        negative.append(-i)
    return negative
print(negative_list(num))
```

DATA ADDING METHODS:

```
# Insert Method:
fruits=['mango','orange']
fruits.insert(1,'grapes') #will insert in position 1
print(fruits)

# How to join(concaenate) two list:
fruits1=['mango','orange']
fruits2=['mango','orange']
fruits=fruits1+fruits2
print(fruits)

# Extend Method:
fruits1=['mango','orange']
fruits2=['mango','orange']
fruits1.extend(fruits2)
print(fruits1)
print(fruits2)
```

```
# Append Method:
fruits1=['mango','orange']
fruits2=['mango','orange']
fruits1.append(fruits2) #list inside list
print(fruits1)
print(fruits2)
```

MIN AND MAX:

```
###
num=[6,60,3]
print(min(num))
print(max(num))

###
def greatest_diff(l):
    return max(l)-min(l)
print(greatest_diff(num))
```

EXERCISE-01:

```
# Exercise_01:
# define a function which will take list containing numbers as input
# and return list containing square of every elements
# example
# numbers = [1,2,3,4]
# square_list (numbers) ----› return ----> [1,2,9, 16]

num=input("Enter list: ").split(",")
list_of_num=list(map(int,num))
# In this method, you use the map() function to apply the int()
# function to each string in the list of strings returned by the
split() method.
# Then you convert the resulting map object to a list using the
list() function.
print(list_of_num)
def square_list(l):
    square=[]
    for i in l:
        square.append(i**2)
    return square
print(square_list(list_of_num))
```

EXERCISE-02:

```
# Exercise_02:
# define a function which will take list as a argument and this
function
# will return a reversed list
# examples:
# [1,2,3,4] --- > [4,3, 2,1]
# ['wordi', 'word2'] ---> ['word2', 'wordi']
# Note you simply do this with reverse method or {t-1]
# but try to do this with the help of append and return method

# Way-1:
num=input("Enter list: ").split(",")
list_of_num=list(map(int,num))
def reverse_list(l):
    l.reverse()
    return l
print(reverse_list(list_of_num))

# Way-2:
num=input("Enter list: ").split(",")
list_of_num=list(map(int,num))
def reverse_list(l):
    return l[::-1]
print(reverse_list(list_of_num))

# Way-3:
num=input("Enter list: ").split(",")
list_of_num=list(map(int,num))
def reverse_list(l):
    reversed=[]
    for i in range(len(l)):
        popped=l.pop()
        reversed.append(popped)
    return reversed
print(reverse_list(list_of_num))
```

EXERCISE-03:

```
# Exercise_03:
# define a function that take list of words as argument and
# return list with reverse of every element in that list
# example
# ['abc', 'tuv', 'xyz'] ---> ['ba', 'vut', 'zyx']

# Way_01:
words_list=input("Enter your word list: ").split(",")
def func_1(w):
    new_word_list=[]
    for i in range(len(w)):
        popped=w.pop()
        new_word_list.insert(0,popped[::-1])
    return new_word_list
print(func_1(words_list))

# Way_02:
words_list=input("Enter your word list: ").split(",")
def func_2(w):
    new_word_list=[]
    for i in w:

        new_word_list.append(i[::-1])
    return new_word_list
print(func_2(words_list))
```

EXERCISE-04:

```python
# Exercise_04:
# filter odd even
# define a function
# input
# list ---> [1, 2,3,4,5, 6,7]
# ouput ----> [[1,3,5,7], [2,4,6]]

num=input("Enter list: ").split(",")
list_of_num=list(map(int,num))

def odd_even(l):
    new_list=[]
    odd=[]
    even=[]
    for i in l:
        if i%2==0:
            even.append(i)
        else:
            odd.append(i)
    new_list=[odd]+[even]
    return new_list

print(odd_even(list_of_num))
```

EXERCISE-05:

```python
# Exercise_05:
# common elements finder function
# define a functions which take 2 lists as input and return a list
# which contains common elements of both lists

# example:
# input ---> [1,2,5,8], [1,2,7,6]
# output ---> [1,2]

# Way_01:
num1=input("Enter 1st list: ").split(",")
```

```
list_of_num1=list(map(int,num1))
num2=input("Enter 2nd list: ").split(",")
list_of_num2=list(map(int,num2))

def common_num(l,w):
    common=[]
    for i in l:
        for j in w:
            if i==j:
                common.append(i)
    return common
print(common_num(list_of_num1,list_of_num2))

# Way_02:
def common_num(l,w):
    common=[]
    for i in l:
        if i in w:
            common.append(i)
    return common
print(common_num(list_of_num1,list_of_num2))
```

EXERCISE-06:

```
# Exercise_06:
# function
# [1,2,3, [1,2], [3,4]] , input
# 2
# type()

def num_of_list(l):
    x=0
    for i in l:
        if type(i)==list:
            x+=1
    return x
number_list=[1,2,3,[2,3],[3,4,5]]
print(num_of_list(number_list))
```