```
# List,Tuple-Assignment
# 1. Variable Swapping:
#    Write a Python program that swaps the values of two variables without using a temporary variable.

# a=10
# b=20
# print("Before swap a,b",a,b)
# a=a+b
# b=a-b
# a=a-b
# print("After swap a,b",a,b)

# output:Before swap a,b 10 20
# After swap a,b 20 10

# 2. Variable Assignment with Multiple Values:
#    Assign multiple variables in a single line with different values (e.g., a=5, b=10, c=15).

# a,b,c=5,10,15
# print(a,b,c)

# output:-5,10,15


# 3. Integer Division and Modulus:
#    Take two integer inputs from the user and print their quotient and remainder.
# n1=int(input("enter number"))
# n2=int(input("enter no"))
# print('Qutient ',n1//n2)
# print('Remainder ',n1%n2)
# output:enter number10
# enter no3
# Qutient  3
# Remainder  1
```

```
# 4. Exponential and Power Operations:
#    Write a function that takes a base and an exponent from the user
# and returns the result of the base raised to the power of the exponent.

# base=int(input("enter base "))
# expo=int(input("enter exponetion"))
# print(base**expo)

# output:enter base 2
# enter exponetion3
# 8

# 5. Finding Absolute Difference:
#    Write a program that finds the absolute difference between two numbers.
# import math
# a=7
# b=-3
# print(int(math.fabs(a-b)))
# output:-10

# 6. Shorthand Arithmetic:
#    Perform the following operations on a variable x:
#    - Increment x by 10
#    - Decrement x by 5
#    - Multiply x by 3
#    - Divide x by 2 using shorthand notation.
# x=10
# x+=10
# print(x)
# x-=5
# print(x)
# x*=3
# print(x)
# x/=2
# print(x)
# output:20
# 15
```

```
# 45
# 22.5


# 7. Floor Division with Lists:
#    Given two lists of numbers, calculate the floor division of corresponding elements.
# import math
# l1=[10,20,30,40,50]
# l2=[3,5,3,2,5]
# i=0
# print(math.floor(l1[0]/l2[0]))
# print(math.floor(l1[1]/l2[1]))
# print(math.floor(l1[2]/l2[2]))
# print(math.floor(l1[3]/l2[3]))
# print(math.floor(l1[4]/l2[4]))

# output:
# 3
# 4
# 10
# 20
# 10



# 8. Bitwise Operators:
#    Use bitwise operators to determine if a number is odd or even.


# 9. String Concatenation with Variables:
#    Given two string variables, first_name and last_name,
# concatenate them into a single full name using f-strings.
# f_name="lavanya"
# l_name="mir"
# print(f"Welcom {f_name} {l_name}")
# output:
# Welcom lavanya mir
```

```
# 10. Complex Expressions:
#     Given a=4 and b=2, write an expression that combines multiplication,
#  addition, and division to evaluate a complex expression.
# a=4
# b=2
# print(a+b*a+b/b**2)
# 4+2*4+2/2**2
# 4+2*4+2/4
# 4+8+2/4
# 4+8+0.5
# output:-12.5

# 11. Input Validation for Integer:
#     Write a program that prompts the user for an integer
# and prints a message depending on whether the integer is positive, negative, or
zero.
# no=int(input("enter integer no "))
# if(no>=0):
#     if(no==0):
#         print('zero')
#     else:
#         print("positive")
# else:
#     print('Negative')

# output:
# enter integer no 10
# positive
# PS C:\Lavanya_Code\Pyton_Lectures> python List_Tuple_Assignment.py
# enter integer no -2
# Negative
# PS C:\Lavanya_Code\Pyton_Lectures> python List_Tuple_Assignment.py
# enter integer no 0
# zero

# 12. Concatenate Strings from User Input:
```

```
#     Take three string inputs from the user and
#  concatenate them into a single sentence with appropriate spaces.
# f=input("Enter first_name")
# l=input("Enter last_name")
# m=input("Enter Middle_name")
# print(f," ",m," ",l)

# output:
# Enter first_namelava
# Enter last_namesamarth
# Enter Middle_namemadure
# lava   madure   samarth

# 13. User Input and Arithmetic Operations:
#     Take two floating-point numbers as input from the user
# and print the sum, difference, product, and quotient.
# a=float(input("ennter no1 "))
# b=float(input("enter no2 "))
# print('a+b= ',a+b)
# print('a-b= ',a-b)
# print('a*b= ',a*b)
# print('a/b= ',a/b)

# output:
# enter no1 10
# enter no2 3
# a+b=  13.0
# a-b=  7.0
# a*b=  30.0
# a/b=  3.3333333333333335

# 14. Finding Largest of Three Numbers:
#     Accept three numbers from the user and find the largest of them without
using conditional statements.
#

# 15. User Input to Create a List:
```

```
#     Write a program that asks the user for a series of space-separated integers and stores them in a list.

# s=input("enter space separated int")
# s=input("enter space separated elements ").split(" ")
# s=list(map(int,s))
# print(s)
# output:-[10, 20, 30, 4]


# 16. Reverse User Input:
#     Take a string input from the user and reverse it.
# s=input("enter s ")
# print(s[::-1])

# output: htramas

# 17. Number of Vowels in User Input:
#     Take a string input from the user and return the number of vowels in it.



# 18. Multiple Inputs and Tuple Packing:
#     Accept multiple values from the user and pack them into a tuple.

# 19. Check If Input Is a Valid Number:
#     Write a program that asks the user for a string and
# checks if it can be converted to a valid number (integer or float).
# s=input("enter a ")
# if(s.isdigit()):
#     print("Valid Number")
# else:
#     print("not Valid Number")



# output:
# PS C:\Lavanya_Code\Pyton_Lectures> python List_Tuple_Assignment.py
```

```
# enter a 10202
# Valid Number
# PS C:\Lavanya_Code\Pyton_Lectures> python List_Tuple_Assignment.py
# enter a sama23
# not Valid Number


# 20. Input Validation with Range:
#     Ask the user to enter a number between 1 and 100 and
# handle cases where the input is not within this range.
# n=int(input('Enter no '))
# if(n not in range(1,100)):
#     print("number is not in range")
# else:
#     print("Number is in range")

# output:
# Enter no 1233
# number is not in range
# PS C:\Lavanya_Code\Pyton_Lectures> python List_Tuple_Assignment.py
# Enter no 2
# Number is in range

# 21. Using f-strings to Format Decimal:
#     Write a Python program that uses f-strings to format a floating-point number
to 2 decimal places.
# n=9.97888
# print(f'{n:.2f}')
# output:9.98

# 22. Dynamic String Construction Using f-strings:
#     Given two variables, name and age,
# construct a string using f-strings to print the sentence, “My name is [name] and I
am [age] years old.”
# name=input('Enter name ')
# age=int(input("enter age "))
# print(f'My name is {name} and I am {age} years old.')
# output:
```

```
# Enter name samarth
# enter age 67
# My name is samarth and I am 67 years old.

# 23. Calculate Circle Area with f-string:
#     Use f-strings to print the area of a circle when the radius is given.
# import math
# r=int(input("Enter radius "))
# print(math.pi*r**2)
# output: 50.2654

# 24. Formatted Date Output:
#     Write a program that takes a date in YYYY-MM-DD format and
# uses an f-string to print the date in Month Day, Year format.



# 25. Alignment with f-string:
#     Create a table with three columns: Item Name, Price, and Quantity.
# Use f-strings to format the columns such that each column is left-aligned.


# 26. Temperature Conversion with f-string:
#     Write a Python program that converts Celsius to Fahrenheit and prints the
result using an f-string.
# t=12
# print((t*(9/5))+32)




# 27. Factorial Calculation:
#     Write a program that calculates the factorial of a number using only
# multiplication (without recursion or loops).
n=4
n=(n*n-1)+(n*n-2)+(n*n-3)
print(n)
```

```
output:24


# 28. Sum of Squares of Numbers:
#     Take a list of integers and calculate the sum of their squares.
# l=[1,2,3,4,5]
# sum=0
# for i in l:
#     sum=sum+i**2
# print(sum)
# output:-55


# 29. Prime Number Check:
#     Write a function that checks if a number is prime.
# n=int(input("enter no "))
# f=0
# for i in range(2,n):
#     if n%i==0:
#         f=1
#         break

# if(f==0):
#     print('Prime')
# else:
#     print('Not prime')

# output:
# enter no 11
# Prime
# PS C:\Lavanya_Code\Pyton_Lectures> python List_Tuple_Assignment.py
# enter no 20
# Not prime

# 30. Even or Odd Sum:
#     Write a program that sums all even numbers and odd numbers from a given list separately.
```

```
# even=0
# odd=0
# l=[10,20,12,23,34,23]
# for i in l:
#     if(i%2==0):
#         even+=i
#     else:
#         odd+=i
# print("even: ",even,"Odd: ",odd)
# output:-
# even:  76 Odd:  46


# 31. Count Divisors:
#     Given a number, count how many numbers from 1 to that number are divisors of the given number.
# n=10
# c=0
# for i in range(1,n+1):
#     if(n%i==0):
#         c=c+1

# print(c)
# ouput:- 4









# 32. Remove Duplicates from List:
#     Write a Python program that removes all duplicates from a given list without using sets.
# uniq_l=[]
# l=[10,20,10,20,30,3,5,10]
# for i in l:
#     if i not in uniq_l:
```

```
#         uniq_l.append(i)
# print(uniq_l)
# output:-
# [10, 20, 30, 3, 5]

# 33. Merge Two Lists:
#     Write a program that takes two lists and merges them alternately
# (i.e., one element from the first list, then one from the second).
# marge=[]
# l1=[10,20,30,40,50]
# l2=[1,2,3,4,5]

# for i in range(len(l1)):
#     marge.append(l1[i])
#     marge.append(l2[i])
# print(marge)

# output:[10, 1, 20, 2, 30, 3, 40, 4, 50, 5]

# 34. List Slicing:

#     Given a list of integers, extract the last 3 elements of the list using slicing.
# l=[10, 1, 20, 2, 30, 3, 40, 4, 50, 5]
# print(l[-3:])

# output:[4,50,5]


# 35. Find Product of List:
#     Write a program that calculates the product of all elements in a list.
# l=[10,20,30]
# prod=1
# for i in l:
#     prod*=i
# print(prod)
# output:600
```

```
# 36. Find Maximum and Minimum in List:
#     Write a Python program that finds the maximum and minimum values from a list
# without using built-in functions.
# min=0
# max=0

# l=[10,20,3,4,109,34,-1,788]

# for i in l:
#     if i>max:
#         max=i
#     elif i<min:
#         min=i
# print(min,max)
#  output: -1 788


# 37. Tuple Concatenation:
#     Given two tuples, concatenate them into a single tuple and print the result.
# t1=(1,2,3,4)
# t2=(10,20,30.40)
# t3=t1+t2
# print(t3)

# output:
# (1, 2, 3, 4, 10, 20, 30.4)


# 38. Tuple Unpacking:
#     Write a Python program that demonstrates tuple unpacking for extracting values into separate variables.
# t=10,20,30,405,'lava',90.88
# print(t)

# output:(10, 20, 30, 405, 'lava', 90.88)

# 39. Tuple Slicing:
```

```
#     Given a tuple of numbers, extract a subset of the tuple from index 2 to index 5.
# t=(10,20,30,405,'lava',90.880)
# t=list(t)
# print(tuple(t[2:6]))
# output:
# (30, 405, 'lava', 90.88)

# 40. Count Occurrences in Tuple:
#     Write a program that counts how many times a specific element appears in a tuple.
# t=(10,20,30,10,3,4,10)
# n=10
# c=0
# for i in t:
#     if(i==n):
#         c+=1
# print(c)
# output: 3
```

Case Studies:

Case Study 1: Shopping Cart System

```
# Problem:
# Create a shopping cart system where a user can input the items they wish to buy.
# For each item, they should input:
# - Item name
# - Price
# - Quantity

# d={}


# n='y'
# i=1
# while(n=='y'):
```

```
#     name=input("enter item name ")
#     Price=int(input('Enter price '))
#     Quntity=int(input("Enter Quntity "))
#     d.update({i:{'Item':name,'Price':Price,'Quntity':Quntity}})
#     i+=1
#     print("Enter y want to continue:")
#     n=input()
# print("Total_Items ",d)
# d2=d
# You should calculate:
# - Total cost of all items
# - Apply a 10% discount if the total cost exceeds $100


# total_cost=0
# for i in d2.keys():
#     total_cost+=d2[i]['Price']*d2[i]['Quntity']


# if(total_cost>100):
#     dis=(total_cost/10)
#     print("Congratulations you got discount of 10%")
#     print("Total_Bill: ",total_cost,"Discounted_Bill ",total_cost-dis)
# else:
#     print("Total_Bill",total_cost)
# Output:
# - List of all items purchased with total price
# - Final total (with or without discount)
# - Itemized list with quantities
```

Output:-

```
PS C:\Lavanya_Code\Pyton_Lectures> python List_Tuple_Assignment.py
enter item name mixer
Enter price 200
Enter Quntity 2
Enter y want to continue:
y
enter item name saree
Enter price 1000
Enter Quntity 1
Enter y want to continue:
n
Total_Items  {1: {'Item': 'mixer', 'Price': 200, 'Quntity': 2}, 2: {'Item': 'saree', 'Price': 1000, 'Quntity': 1}}
Congratulations you got discount of 10%
Total_Bill:  1400 Discounted_Bill  1260.0
PS C:\Lavanya_Code\Pyton_Lectures>
```

```
#  Case Study 2: Student Grades System

# Problem:
# Write a program that stores student names and their grades in a tuple. The program should:
# - Take multiple students' names and grades as input
# - Allow the user to retrieve the grade for a specific student
# - Calculate the average grade of the class
# - Identify the highest and lowest grades in the class

# d={}
# i=0
# n=int(input("Enter no. of students in class"))
# while(n):
#     d.update({i:(input("Enter Name "),int(input("Enter grade ")))})
#     n-=1
#     i+=1
# print(d)
# sum=0
# max=0
# l=list(d.values())
# min=l[0][1]
```

```
# for k,v in d.values():
#     sum+=v
#     if(v>max):
#         max=v

#     if(v<min):
#         min=v
# print("Average_Grade_Class",sum/len(d))
# print("Highest Grade ",max)
# print('Lowest Grade',min)
```

output:-

```
PS C:\Lavanya_Code\Pyton_Lectures> python List_Tuple_Assignment.py
Enter no. of students in class2
Enter Name lavanya
Enter grade 10
Enter Name samarth
Enter grade 20
{0: ('lavanya', 10), 1: ('samarth', 20)}
Average_Grade_Class 15.0
Highest Grade  20
Lowest Grade 10
PS C:\Lavanya_Code\Pyton_Lectures>
```

```
#  Case Study 3: Inventory Management System

# Problem:
# Create an inventory management system where the program takes:
# - Item name
# - Quantity available
# - Price per item
inventory={

    'mixer':{
    'quantity':10,
    'price':100,
  },
```

```
    'cups':{
     'quantity':20,
     'price':10,
   }
}
print(inventory)
# The program should:
# - Keep track of the items in the inventory
# - Allow users to update the quantity of an item
# - Calculate the total value of the inventory based on the prices and quantities of all items
k=input("Enter name of item you want change ")
q=int(input("enter quantity "))
inventory[k]['quantity']=q
print(inventory)

total_value=0
for v in inventory.values():
  total_value+= v['quantity']*v['price']
print('total_value: ',total_value)
```

output:-

```
PS C:\Lavanya_Code\Pyton_Lectures> python List_Tuple_Assignment.py
{'mixer': {'quantity': 10, 'price': 100}, 'cups': {'quantity': 20, 'price': 10}}
Enter name of item you want change mixer
enter quantity 5
{'mixer': {'quantity': 5, 'price': 100}, 'cups': {'quantity': 20, 'price': 10}}
total_value:  700
PS C:\Lavanya_Code\Pyton_Lectures>
```