1. **Session 1 & 2**

# Task 1: Accept Empid, EmpName, Monthly\_Salary, Tot\_Deductions, Tot\_Allowances

emp\_id = input("Enter Employee ID: ")

emp\_name = input("Enter Employee Name: ")

monthly\_salary = float(input("Enter Monthly Salary: "))

tot\_deductions = float(input("Enter Total Deductions: "))

tot\_allowances = float(input("Enter Total Allowances: "))

salary\_in\_hand = monthly\_salary - tot\_deductions + tot\_allowances

print(f"Employee Name: {emp\_name}, Salary in Hand: {salary\_in\_hand}")

# If Condition 1: Accept 3 integers from User and Display Maximum

num1 = int(input("Enter first integer: "))

num2 = int(input("Enter second integer: "))

num3 = int(input("Enter third integer: "))

if num1 >= num2 and num1 >= num3:

print(f"Maximum: {num1}")

elif num2 >= num1 and num2 >= num3:

print(f"Maximum: {num2}")

else:

print(f"Maximum: {num3}")

# If Condition 2: Accept 3 integers from User and Display Minimum

if num1 <= num2 and num1 <= num3:

print(f"Minimum: {num1}")

elif num2 <= num1 and num2 <= num3:

print(f"Minimum: {num2}")

else:

print(f"Minimum: {num3}")

# Loops Task 1: Accept Integers until User's Choice

#to calculate:

#sum, average, maximum and minimum

integers = []

while True:

user\_input = input("Enter an integer (or type 'stop' to finish): ")

if user\_input.lower() == 'stop':

break

else:

integers.append(int(user\_input))

sum\_of\_integers = sum(integers)

average\_of\_integers = sum\_of\_integers / len(integers) if integers else 0

max\_integer = max(integers) if integers else None

min\_integer = min(integers) if integers else None

print(f"Sum: {sum\_of\_integers}")

print(f"Average: {average\_of\_integers}")

print(f"Maximum: {max\_integer}")

print(f"Minimum: {min\_integer}")

# Loops Task 2: Accept a String and Perform Operations

user\_string = input("Enter a string: ")

# Length of String

length\_of\_string = len(user\_string)

print(f"Length: {length\_of\_string}")

# Reverse of String

reversed\_string = user\_string[::-1]

print(f"Reversed String: {reversed\_string}")

# Every alternate character in Uppercase

alt\_case\_string = ''.join([char.upper() if i % 2 == 0 else char for i, char in enumerate(user\_string)])

print(f"Alternate Case String: {alt\_case\_string}")

# Count Vowels

vowels = "aeiouAEIOU"

vowel\_count = sum(1 for char in user\_string if char in vowels)

print(f"Number of Vowels: {vowel\_count}")

# Task 4: Accept Username and Date of Birth, Create Password String

username = input("Enter your username: ")

dob = input("Enter your Date of Birth (dd-mon-yy): ")

password = username[:4] + dob[-2:] + "$"

print(f"Generated Password: {password}")

# Task 5: Simple Encryption (Caesar Cipher Shift by 3)

encrypted\_password = ''.join([chr(ord(char) + 3) for char in password])

print(f"Encrypted Password: {encrypted\_password}")

# Task 3.1: Calculate Area of Circle and Parallelogram

radius = float(input("Enter radius of circle: "))

circle\_area = 3.14159 \* radius \* radius

print(f"Area of Circle: {circle\_area}")

base = float(input("Enter base of parallelogram: "))

height = float(input("Enter height of parallelogram: "))

parallelogram\_area = base \* height

print(f"Area of Parallelogram: {parallelogram\_area}")

# Task 4: Accept an Integer and Find Square Root

integer\_value = int(input("Enter an integer: "))

sqrt\_value = integer\_value \*\* 0.5

print(f"Square root of {integer\_value}: {sqrt\_value}")

# Session 3 / 4

fruits\_list = []

while True:

# Display the menu

print("\nMenu:")

print("1. Add New Fruits")

print("2. Buy Fruits")

print("3. Show Total Fruits in the List")

print("4. Exit")

option = input("Choose an option (1/2/3/4): ")

if option == '1':

# Add new fruits

n = int(input("Enter the number of fruits to add: ")) # Number of fruits to add

for i in range(n):

fruit\_name = input(f"Enter the name of fruit {i+1}: ") # Get fruit name

price = float(input(f"Enter the price of {fruit\_name} per kg: ")) # Get price

# Append price at even index, fruit name at odd index

fruits\_list.append(price)

fruits\_list.append(fruit\_name)

elif option == '2':

if len(fruits\_list) == 0:

print("No fruits available. Please add fruits first.")

else:

# Show the available fruits menu

print("\nAvailable Fruits Menu:")

for i in range(1, len(fruits\_list), 2):

print(f"{i//2 + 1}. {fruits\_list[i]} - {fruits\_list[i-1]} per kg")

total\_price = 0 # Initialize total price

# Customer selects fruits to buy

while True:

choice = input("Enter the number of the fruit to buy or 'done' to finish: ")

if choice.lower() == 'done':

break

fruit\_index = (int(choice) - 1) \* 2 # Calculate the index of the chosen fruit

total\_price += fruits\_list[fruit\_index] # Add the price to the total

print(f"\nTotal Price of Fruits Bought: {total\_price}")

elif option == '3':

if len(fruits\_list) == 0:

print("No fruits in the list.")

else:

# Show the total list of fruits and prices

print("\nTotal Fruits in the List:")

for i in range(1, len(fruits\_list), 2):

print(f"{i//2 + 1}. {fruits\_list[i]} - {fruits\_list[i-1]} per kg")

elif option == '4':

print("Thank you for using. Goodbye!")

break

else:

print("Invalid option, please try again.")

# Tuple to store employee details (EmpId, Phone Numbers)

# Each element of the tuple is a tuple itself: (EmpId, [Phone Numbers])

employees = (

(101, ["9876543210", "8765432109"]),

(102, ["9123456780"]),

(103, ["9812345678", "9900112233"]),

(104, ["9654321987"]),

(105, ["9123487650", "9988776655"])

)

# Convert tuple to list for updates

employees\_list = list(employees)

# Main program flow

while True:

print("\nMenu:")

print("1. Display Employee Phone Numbers")

print("2. Update Employee Phone Number")

print("3. Exit")

option = input("Choose an option (1/2/3): ")

if option == '1':

# Display phone numbers of the employee

emp\_id = int(input("Enter the Employee ID to search: "))

# Search for employee by ID

emp\_found = False

for emp in employees:

if emp[0] == emp\_id:

print(f"Phone numbers for Employee ID {emp\_id}: {', '.join(emp[1])}")

emp\_found = True

break

if not emp\_found:

print(f"Employee with ID {emp\_id} not found in the database.")

elif option == '2':

# Update phone number of an employee

emp\_id = int(input("Enter the Employee ID to update: "))

# Search for employee by ID

emp\_found = False

for emp in employees\_list:

if emp[0] == emp\_id:

new\_phone\_number = input("Enter the new phone number to add: ")

# Update the phone number

emp[1].append(new\_phone\_number)

print(f"Phone number {new\_phone\_number} added for Employee ID {emp\_id}.")

emp\_found = True

break

if not emp\_found:

print(f"Employee with ID {emp\_id} not found in the database.")

elif option == '3':

print("Thank you for using. Goodbye!")

break

else:

print("Invalid option, please try again.")

# Create sets for the fruits each salesman has

Fruit\_Salesman1 = {"Apple", "Banana", "Mango", "Orange", "Grapes"}

Fruit\_Salesman2 = {"Pineapple", "Banana", "Mango", "Peach", "Grapes"}

# Menu for operations

while True:

print("\nMenu:")

print("1. Find out Common Fruits with both Salesmen")

print("2. List Extra Fruits with Both Salesmen")

print("3. List Total Fruits with both Salesmen")

print("4. Exit")

option = input("Choose an option (1/2/3/4): ")

if option == '1':

# Find common fruits between both salesmen

common\_fruits = Fruit\_Salesman1.intersection(Fruit\_Salesman2)

if common\_fruits:

print(f"Common fruits between both salesmen: {', '.join(common\_fruits)}")

else:

print("There are no common fruits between both salesmen.")

elif option == '2':

# List extra fruits with both salesmen (fruits that are not common)

extra\_fruits\_salesman1 = Fruit\_Salesman1.difference(Fruit\_Salesman2)

extra\_fruits\_salesman2 = Fruit\_Salesman2.difference(Fruit\_Salesman1)

print(f"Extra fruits with Salesman 1: {', '.join(extra\_fruits\_salesman1)}")

print(f"Extra fruits with Salesman 2: {', '.join(extra\_fruits\_salesman2)}")

elif option == '3':

# List total fruits with both salesmen (all unique fruits)

total\_fruits = Fruit\_Salesman1.union(Fruit\_Salesman2)

print(f"Total unique fruits with both salesmen: {', '.join(total\_fruits)}")

elif option == '4':

print("Thank you for using. Goodbye!")

break

else:

print("Invalid option, please try again.")