PIYUSH KUMAR MISHRA

230957212

WEEK 4:

Exercise 1: Customer Directory Management System
Objective
The goal of this case study is to develop a Python program that manages customer information for a telecommunications company. This involves creating a function to handle customer data, extracting relevant information, and printing the data in various formats.

Requirements
1. Function Definition:
A. Define a function named tel_directory that takes a list of dictionaries representing customer information. Each dictionary contains:
a. customer_id (Unique identifier for the customer)
b. customer_name (Name of the customer)
c. Subscription_type (Type of subscription: "prepaid" or "postpaid")

2. Data Input:
B. Input at least 10 customer records using the tel_directory function.

3. Data Extraction:
C. Extract data from the list of dictionaries and create a list of lists. Each list should be structured based on different key combinations.

4. Output:
D. Print the extracted data in three different combinations of key fields:
a. Combination 1: A list containing [customer_id, customer_name, Subscription_type] b. Combination 2: A list containing [customer_id, customer_name]
c. Combination 3: A list containing [customer_name, Subscription_type]

Exercise 2: Enhancing the Customer Directory Management System

Add a new function named Search_Customer to the existing Customer Directory Management System.
This function should:
1. Function Purpose:

A. Search for a customer by their name within the directory.

2. Function Details:

A. The function should take the customer_name as input.

B. If the customer is found in the directory, display their information.

C. If the customer is not found, print a message indicating that the customer is not in the directory.

Exercise 3: Enhancing the Customer Directory Management System

Add a new function named Search_subscription to the existing Customer Directory Management System. This function should:

1. Function Purpose:

A. Search for customers based on their subscription type.

2. Function Details:

A. The function should take the Subscription_type ("prepaid" or "postpaid") as input.

B. Display the information of all customers who have the specified subscription type.

C. If no customers have the given subscription type, print appropriate message.

```
def tel_directory(customers):

    combo1 = [[cust['customer_id'], cust['customer_name'], cust['subscription_type']] for cust in customers]

    combo2 = [[cust['customer_id'], cust['customer_name']] for cust in customers]

    combo3 = [[cust['customer_name'], cust['subscription_type']] for cust in customers]


    print("Combination 1: [customer_id, customer_name, subscription_type]")

    print(combo1)


    print("Combination 2: [customer_id, customer_name]")

    print(combo2)


    print("Combination 3: [customer_name, subscription_type]")

    print(combo3)
```

```
def Search_Customer(customers, customer_name):
    found = False
    for cust in customers:
        if cust['customer_name'].lower() == customer_name.lower():
            print(f"Customer Found: ID: {cust['customer_id']}, Name: {cust['customer_name']}, Subscription Type: {cust['subscription_type']}")
            found = True
            break
    if not found:
        print("Customer not found in the directory.")

def Search_subscription(customers, subscription_type):
    found = False
    for cust in customers:
        if cust['subscription_type'].lower() == subscription_type.lower():
            print(f"Customer Found: ID: {cust['customer_id']}, Name: {cust['customer_name']}, Subscription Type: {cust['subscription_type']}")
            found = True
    if not found:
        print("No customers found with the specified subscription type.")

def main():
    customers = []

    num_customers = int(input("Enter the number of customers to add: "))
    for _ in range(num_customers):
        customer_id = input("Enter customer ID: ")
        customer_name = input("Enter customer name: ")
```

```
        subscription_type = input("Enter subscription type: ")

        # Adding customer information to the list
        customers.append({
            'customer_id': customer_id,
            'customer_name': customer_name,
            'subscription_type': subscription_type
        })

    # Display the different combinations
    tel_directory(customers)

    # Search for a customer by name
    search_name = input("Enter customer name to search: ")
    Search_Customer(customers, search_name)

    # Search for customers by subscription type
    search_subscription = input("Enter subscription type to search: ")
    Search_subscription(customers, search_subscription)

if __name__ == "__main__":
    main()
```

OUTPUT:

```
Enter the number of customers to add:  2
Enter customer ID:  101
Enter customer name:  MAYANK
Enter subscription type:  prepaid
Enter customer ID:  102
```

Enter customer name: Nishant
Enter subscription type: postpaid
Combination 1: [customer_id, customer_name, subscription_type]
[['101', 'MAYANK', 'prepaid'], ['102', 'Nishant', 'postpaid']]
Combination 2: [customer_id, customer_name]
[['101', 'MAYANK'], ['102', 'Nishant']]
Combination 3: [customer_name, subscription_type]
[['MAYANK', 'prepaid'], ['Nishant', 'postpaid']]
Enter customer name to search: MAYANK
Customer Found: ID: 101, Name: MAYANK, Subscription Type: prepaid
Enter subscription type to search: prepaid
Customer Found: ID: 101, Name: MAYANK, Subscription Type: prepaid