ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to the Principal, Ms. Shehanaz Cottar, for her encouragement and all the support that she provided for this project. I sincerely appreciate this magnanimity by taking me into her fold for which I shall remain indebted to her.

I also extend my hearty thanks to Computer Science teacher, Ms. Dipti Lagad, who guided me to the successful completion of this project. I take this opportunity to express my deep sense of gratitude for her valuable guidance, constant encouragement and immense motivation which has sustained my efforts at all stages of this project work.

INDEX

REQUIREMENTS

1. **Hardware Requirements:**
   i. x86 64-bit CPU (intel/AMD architecture)
   ii. 4GB RAM
   iii. 5GB free disk space
2. **Software Requirements:**
   i. Modern Operating System:
      a. Windows 7 or 10
      b. Mac OS X 10.11 or higher
      c. Linux: RHEL 6/7, 64-bit
   ii. Python 3 or later
   iii. Additional Libraries:
      a. Sys
      b. Datetime
      c. Time
      d. Mysql.connector

PROGRAM SPECIFICATIONS

NewsPaper Management Program

Our Python program, "NewspaperPal," functions as a comprehensive information management system catering to both customers and staff members. Featuring an aesthetically pleasing opening frame and menu structure, the program leverages MySQL database connectivity to handle customer subscriptions efficiently. User authentication is implemented to ensure secure interactions, allowing existing customers to manage subscriptions, view details, and effect changes, while new customers can seamlessly register with unique usernames and passwords. Staff members benefit from analytics functionalities, enabling them to analyze customer data, calculate total fees, and gain insights into subscription patterns. The program exhibits a user-friendly interface, incorporating time delays for visual appeal. Noteworthy features include robust error handling during registration and the utilization of looping structures for repetitive tasks. Overall, "NewspaperPal" serves as an effective tool for newspaper subscription management, combining database interaction, user authentication, and analytical capabilities for an enhanced user experience.

SOURCE CODE

```
import mysql.connector as sqltor
import time
from prettytable import PrettyTable


def opening_frame():
    print("*****************************************")
    time.sleep(0.6)
    print("*    Welcome to NewspaperPal!    *")
    time.sleep(0.6)
    print("*   Your Source for Daily News   *")
    time.sleep(0.6)
    print("*****************************************")
    time.sleep(0.6)


def print_menu_options():
    print("*****************************************")
    time.sleep(0.6)
    print("*           NewspaperPal Menu          *")
    time.sleep(0.6)
    print("*****************************************")
    time.sleep(0.6)
    print("1] Existing Customer")
    time.sleep(0.6)
    print("2] New Customer")
    time.sleep(0.6)
    print("3] Staff member")
    print("*****************************************")

def print_customer_menu():
    print("*****************************************")
    time.sleep(0.6)
    print("*            Customer Menu            *")
    time.sleep(0.6)
    print("*****************************************")
    print("1] View Subscriptions")
    time.sleep(0.6)
    print("2] View Subscription Duration")
    time.sleep(0.6)
    print("3] View Publication")
    time.sleep(0.6)
    print("4] View All Subscriptions")
    time.sleep(0.6)
    print("5] Add a Subscription")
    time.sleep(0.6)
    print("6] Change Username and Password")
    time.sleep(0.6)
    print("7] Exit")
    time.sleep(0.6)
    print("*****************************************")

def print_registration_menu():
    print("*****************************************")
    time.sleep(0.6)
    print("*        Registration Menu          *")
    time.sleep(0.6)
    print("*****************************************")
    time.sleep(0.6)
    print("1] Continue Registration")
    time.sleep(0.6)
    print("2] Back to Main Menu")
    time.sleep(0.6)
    print("*****************************************")
```

```
def staff_menu():
    print("*********************************************")
    time.sleep(0.6)
    print("*              Staff Member Menu             *")
    time.sleep(0.6)
    print("*********************************************")
    time.sleep(0.6)
    print("1] View All Customer Names")
    time.sleep(0.6)
    print("2] Calculate Total Fee of Each customer")
    time.sleep(0.6)
    print("3] Calculate Total No. of Newspaper Publications")
    time.sleep(0.6)
    print("4] Display All NewsPaper Publications")
    time.sleep(0.6)
    print("5] Display Data of a Customer ")
    time.sleep(0.6)
    print("6] Display a specific Newspaper publicaiton data")
    time.sleep(0.6)
    print("7] Display subscription by duration")
    time.sleep(0.6)
    print("8] Display expiring subscriptions")
```

```
   time.sleep(0.6)

   print("9] Display most subscribed newspaper")

   time.sleep(0.6)

   print("10] Exit")

   time.sleep(0.6)

   print("***********************************************")


# Display the opening frame

opening_frame()


time.sleep(2)


mycon = sqltor.connect(host='localhost', user='new_user',

   password='password', database='newspaperpal')


if mycon.is_connected():

   print("Connection is Successful")


else:

   print("Not able to connect to the database")


cursor = mycon.cursor()
```

```
def login(username, password):
    query = f"SELECT * FROM login WHERE username='{username}'
    AND password='{password}'"
    cursor.execute(query)
    result = cursor.fetchone()
    return result is not None


def view_all_customer_names():
    query = "SELECT cname FROM customerdata"
    cursor.execute(query)
    customers = cursor.fetchall()


    if customers:
        print("### All Customer Names ###")
        time.sleep(0.6)
        for customer in customers:
            print(customer[0])
    else:
        print("No customers found.")
```

```
def calculate_total_prices():
    query = "SELECT SUM(price) FROM customerdata"
    cursor.execute(query)
    total_prices = cursor.fetchone()[0]

    if total_prices:
        print(f"Total Prices: {total_prices}")
    else:
        print("No prices found.")

def calculate_total_publications():
    query = "SELECT COUNT(DISTINCT newspaperpublication) FROM
    customerdata"
    time.sleep(0.6)
    cursor.execute(query)
    total_publications = cursor.fetchone()[0]

    if total_publications:
        print(f"Total Publications: {total_publications}")
    else:
        print("No publications found.")
```

```
def display_customer_data(username):
    query = f"SELECT * FROM customerdata WHERE cname='{username}'"
    cursor.execute(query)
    time.sleep(0.6)
    customer_data = cursor.fetchall()

    if customer_data:
        print(f"### Customer Data for {username} ###")
        time.sleep(0.6)
        table = PrettyTable(["Subscription ID", "Publication", "Duration", "Price"])
        for data in customer_data:
            table.add_row([data[0], data[1], data[2], data[3]])
        print(table)
    else:
        print(f"No data found for customer: {username}")

def view_subscriptions_by_duration_range(min_duration, max_duration):
    query = f"SELECT cname, newspaperpublication, duration FROM customerdata WHERE duration BETWEEN {min_duration} AND {max_duration}"
    cursor.execute(query)
    subscriptions_in_range = cursor.fetchall()

    if subscriptions_in_range:
        print("\nSubscriptions in Duration Range:")
        print("==================================")
        for data in subscriptions_in_range:
            print(f"{data[0]} - {data[1]}: {data[2]} days")
    else:
        print("No subscriptions found in the specified duration range.")



def display_all_publications():
    query = "SELECT DISTINCT newspaperpublication FROM
    customerdata"
    cursor.execute(query)
    publications = cursor.fetchall()

    if publications:
        print("### All Publications ###")
        table = PrettyTable(["Publication"])
        for publication in publications:
```

```
        table.add_row([publication[0]])
    print(table)
  else:
    print("No publications found.")

def search_newspaper_data(newspaper_name):
  query = f"SELECT * FROM customerdata WHERE
  newspaperpublication = '{newspaper_name}'"
  cursor.execute(query)
  newspaper_data = cursor.fetchall()

  if newspaper_data:
    print(f"\nData for '{newspaper_name}':")
    print("================================")
    for data in newspaper_data:
      print(f"Customer ID: {data[0]}")
      print(f"Newspaper Publication: {data[1]}")
      print(f"Subscription Duration: {data[2]} days")
      print("----------------------------")
  else:
    print(f"No data found for '{newspaper_name}'.")
```

```
def display_expiring_subscriptions(days_threshold=7):
    query = f"SELECT cname, MAX(duration) AS max_duration FROM
    customerdata GROUP BY cname HAVING max_duration <=
    {days_threshold}"
    cursor.execute(query)
    expiring_subscriptions = cursor.fetchall()

    if expiring_subscriptions:
        print("\nCustomers with Expiring Subscriptions:")
        print("===========================================")
        for data in expiring_subscriptions:
            print(f"{data[0]}: {data[1]} days remaining")
    else:
        print("No customer data found.")


def display_most_subscriptions():
    query = "SELECT cname, COUNT(*) AS subscription_count FROM
    customerdata GROUP BY cname ORDER BY subscription_count
    DESC"
    cursor.execute(query)
    most_subscriptions = cursor.fetchall()
    if most_subscriptions:
        print("\nCustomers with Most Subscriptions:")
        print("=====================================")
        for data in most_subscriptions:
            print(f"{data[0]}: {data[1]} subscriptions")
    else:
        print("No customer data found.")




def check_credentials(username, password):
    query = f"SELECT * FROM login WHERE username='{username}'
    AND password='{password}'"
    cursor.execute(query)
    result = cursor.fetchone()
    return result is not None

def display_subscriptions(username):
    query = f"SELECT newspaperpublication,subscription FROM
    customerdata WHERE cname='{username}'"
    cursor.execute(query)
    subscriptions = cursor.fetchall()
    if subscriptions:
        for subscription in subscriptions:
            print(subscription)
    else:
        print("No subscriptions found for this user.")

def change_subscription_duration(username):
    query = f"SELECT subscription FROM customerdata WHERE
    cname='{username}'"
    cursor.execute(query)
    subscriptions = cursor.fetchall()
    if subscriptions:
        for subscription in subscriptions:
            print(subscription)
    else:
        print("No subscriptions duration found for this user.")
    pass

def change_publication(username):
    query = f"SELECT newspaperpublication FROM cusotmerdata
    WHERE cname='{username}'"
    cursor.execute(query)
    subscriptions = cursor.fetchall()
    if subscriptions:
        for subscription in subscriptions:
            print(subscription)
    else:
        print("No subscriptions duration found for this user.")
    pass

def all_subscription(username):
    query = f"SELECT * FROM customerdata WHERE
    cname='{username}'"
    cursor.execute(query)
    subscriptions = cursor.fetchall()

    if subscriptions:
        for subscription in subscriptions:
            print(subscription)

    else:
        print("No subscriptions duration found for this user.")
        pass
```

```
def add_newspaper_subscription(username):
    print("### Add Newspaper Subscription ###")
    newspaper_publication = input("Enter the newspaper publication: ")
    subscription_duration = int(input("Enter the subscription duration (in
    days): "))

    # Insert the subscription data into the customerdata table
    cursor.execute("INSERT INTO customerdata (uid,
    newspaperpublication, duration) VALUES (%s, %s, %s)",
                   (username, newspaper_publication, subscription_duration))
    mycon.commit()
    print("Newspaper subscription added successfully!\n")


def change_username_password(username, password=123):
    # Check if the provided username and password are valid
    if check_credentials(username, password):
        new_username = input("Enter a new username: ")
        time.sleep(0.6)
        new_password = input("Enter a new password: ")
        # Check if the new username is already taken
        cursor.execute(f"SELECT * FROM login WHERE username =
    '{new_username}'")
        time.sleep(0.6)
        existing_user = cursor.fetchone()

        if existing_user:
            print("Username already exists. Please choose a different
    username.")
            time.sleep(0.6)

        else:
            # Update the username and password in the database
            cursor.execute("UPDATE login SET username = %s, password =
    %s WHERE username = %s AND password = %s",
                          (new_username, new_password, username, password))
            mycon.commit()
            time.sleep(0.6)
            print("Username and password changed successfully!\n")
    else:
        print("Invalid credentials. Unable to change username and
    password.")
```

```
# Main
print("Welcome!")
time.sleep(0.75)
print_menu_options()
print("Press '1' for old customer and '2' for new customer")
time.sleep(0.75)
n = int(input("Enter the number:- "))
time.sleep(1.25)

if n == 1:
    username = input("Enter your username: ")
    time.sleep(0.6)
    password = input("Enter your Password: ")

    if check_credentials(username, password):
        while True:
            print_customer_menu()
            number = int(input("Enter the number: "))
            time.sleep(0.75)
```

```
if number == 1:
    display_subscriptions(username)

elif number == 2:
    change_subscription_duration(username)

elif number == 3:
    change_publication(username)

elif number == 4:
    all_subscription(username)

elif number == 5:
    add_newspaper_subscription(username)

elif number == 6:
    change_username_password(username)

else:
    print("Please enter a valid number!!")

repeat = input("Do you want to repeat? (yes/no)").lower()
```

```
            if repeat != "yes":
                break

    else:
        print("Invalid credentials")

elif n == 2:
    print("### New Customer Registration ###")
    time.sleep(0.6)
    new_username = input("Enter a new username: ")
    time.sleep(0.6)
    new_password = input("Enter a new password: ")
    time.sleep(0.6)

    # Check if the username is already taken
    cursor.execute(f"SELECT * FROM login WHERE username =
    '{new_username}'")
    time.sleep(0.6)
    existing_user = cursor.fetchone()

    if existing_user:
        print("Username already exists. Please choose a different username.")
```

```
else:
    # Insert new customer data into the database
    cursor.execute("INSERT INTO login (username, password) VALUES (%s, %s)", (new_username, new_password))
    mycon.commit()
    print("Registration successful!\n")

if check_credentials(username, password):
    while True:
        print("Select the number:- ")
        time.sleep(0.75)
        print("1]To see your subscriptions")
        time.sleep(0.75)
        print("2]To view the subscription duration")
        time.sleep(0.75)
        print("3]To view the publication")
        time.sleep(0.75)
        print("4]To view all subscription")
        time.sleep(0.75)
        print("5]To add a Subscription")
        time.sleep(0.75)
```

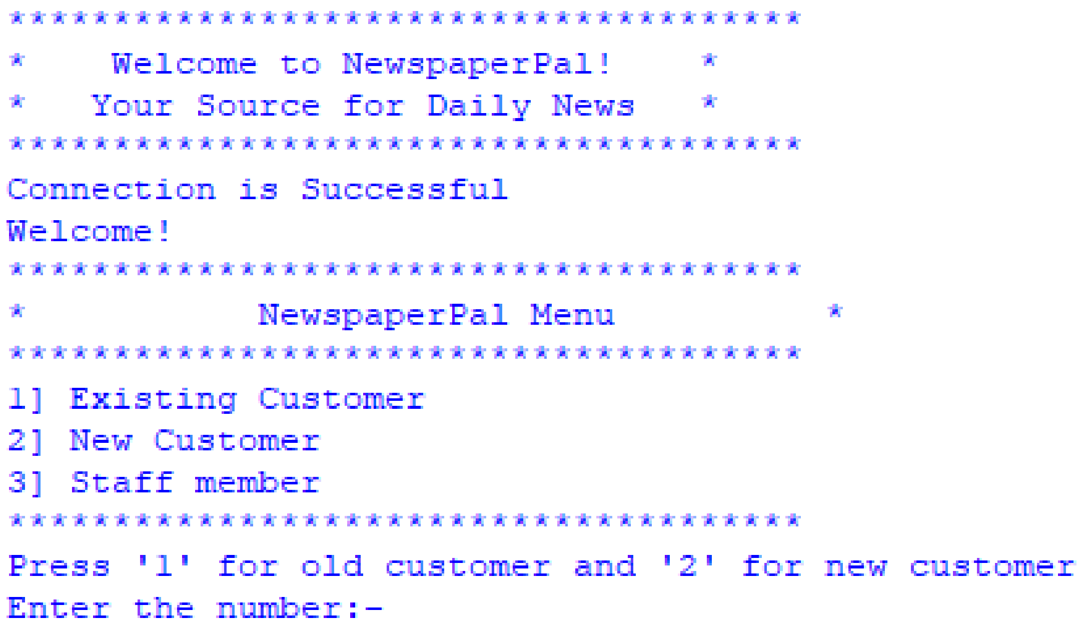
```
print("6]To Change your username and password")
number = int(input("Enter the number: "))
time.sleep(0.75)

if number == 1:
    display_subscriptions(username)

elif number == 2:
    change_subscription_duration(username)

elif number == 3:
    change_publication(username)

elif number == 4:
    all_subscription(username)

elif number == 5:
    add_newspaper_subscription(username)

elif number == 6:
    change_username_password(username)
```

```
            else:
                print("Please enter a valid number!!")

            time.sleep(0.6)
            repeat = input("Do you want to repeat? (yes/no)").lower()

            if repeat != "yes":
                break



elif n == 3:
    print("### Staff Member Login ###")
    staff_username = input("Enter your username: ")
    time.sleep(0.6)
    staff_password = input("Enter your password: ")
    time.sleep(0.6)
    if login(staff_username, staff_password):
        print("Login successful!\n")

    while True:
        staff_menu()
        time.sleep(0.6)
```

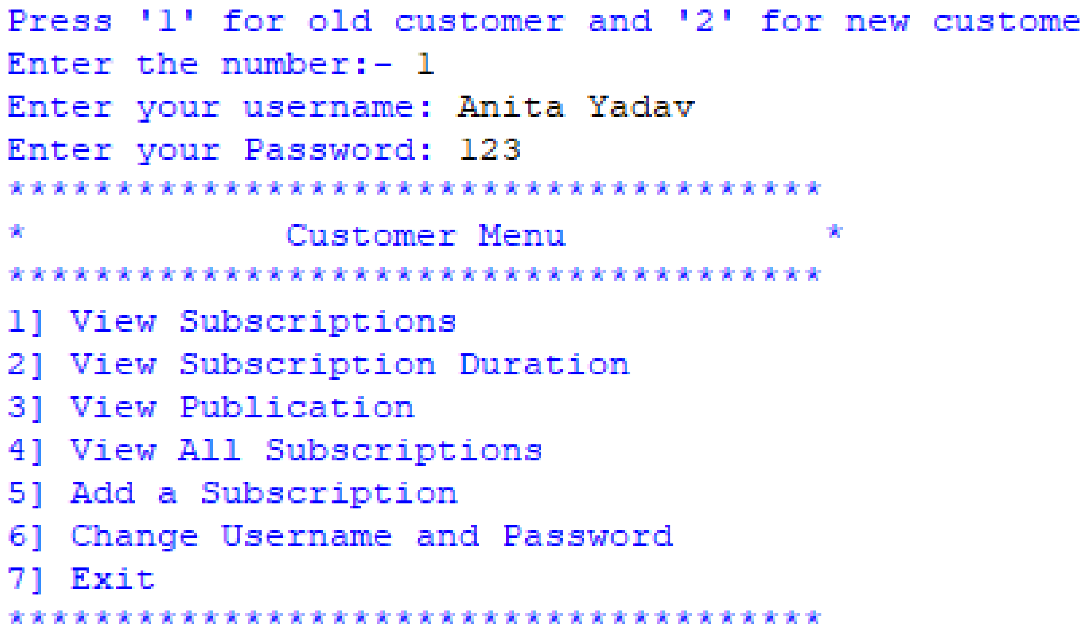
```
choice = input("Enter the number: ")

if choice == '1':
    view_all_customer_names()

elif choice == '2':
    calculate_total_prices()

elif choice == '3':
    calculate_total_publications()

elif choice == '4':
    display_all_publications()

elif choice == '5':
    name=input("Enter the customer name:- ")
    display_customer_data(name)

elif choice=='6':
    newspaper = input("Enter the newspaper you want to search for: ")
    search_newspaper_data(newspaper)
    elif choice=='7':
        a,b = int(input("input the range between which you want to view the subscription: "))
        view_subscriptions_by_duration_range(a,b)

    elif choice=='8':
        display_expiring_subscriptions()

    elif choice == '9':
        display_most_subscriptions()

    elif choice == '10':
        print("Exiting program. Goodbye!")
        break

    else:
        print("Please enter a valid number!")

else:
    print("Invalid credentials. Login failed.")
```

```
else:
   print("Please enter a valid number!!")


mycon.close()
```

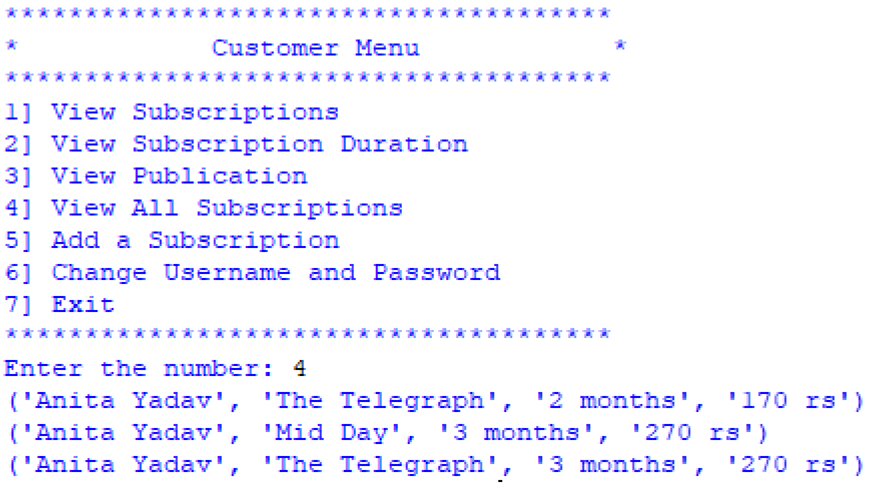
Console outputs



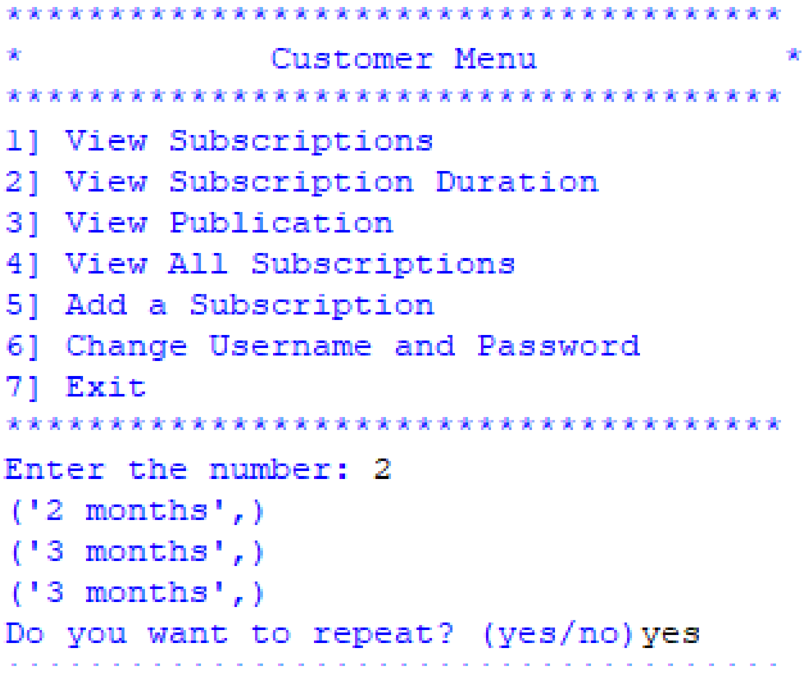
```
*************************************
*    Welcome to NewspaperPal!    *
*   Your Source for Daily News   *
*************************************
Connection is Successful
Welcome!
*************************************
*           NewspaperPal Menu           *
*************************************
1] Existing Customer
2] New Customer
3] Staff member
*************************************
Press '1' for old customer and '2' for new customer
Enter the number:-
```



```
Press '1' for old customer and '2' for new custome
Enter the number:- 1
Enter your username: Anita Yadav
Enter your Password: 123
*************************************
*             Customer Menu             *
*************************************
1] View Subscriptions
2] View Subscription Duration
3] View Publication
4] View All Subscriptions
5] Add a Subscription
6] Change Username and Password
7] Exit
*************************************
```

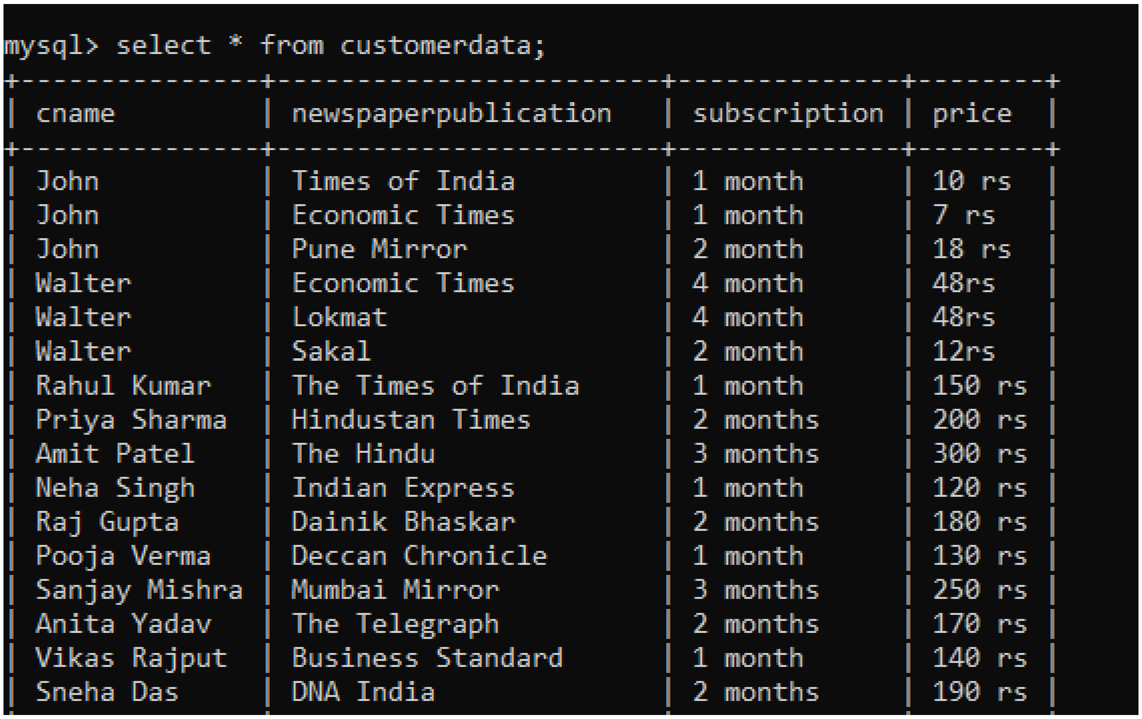


```
**************************************
*             Customer Menu           *
**************************************
1] View Subscriptions
2] View Subscription Duration
3] View Publication
4] View All Subscriptions
5] Add a Subscription
6] Change Username and Password
7] Exit
**************************************
Enter the number: 4
('Anita Yadav', 'The Telegraph', '2 months', '170 rs')
('Anita Yadav', 'Mid Day', '3 months', '270 rs')
('Anita Yadav', 'The Telegraph', '3 months', '270 rs')
```

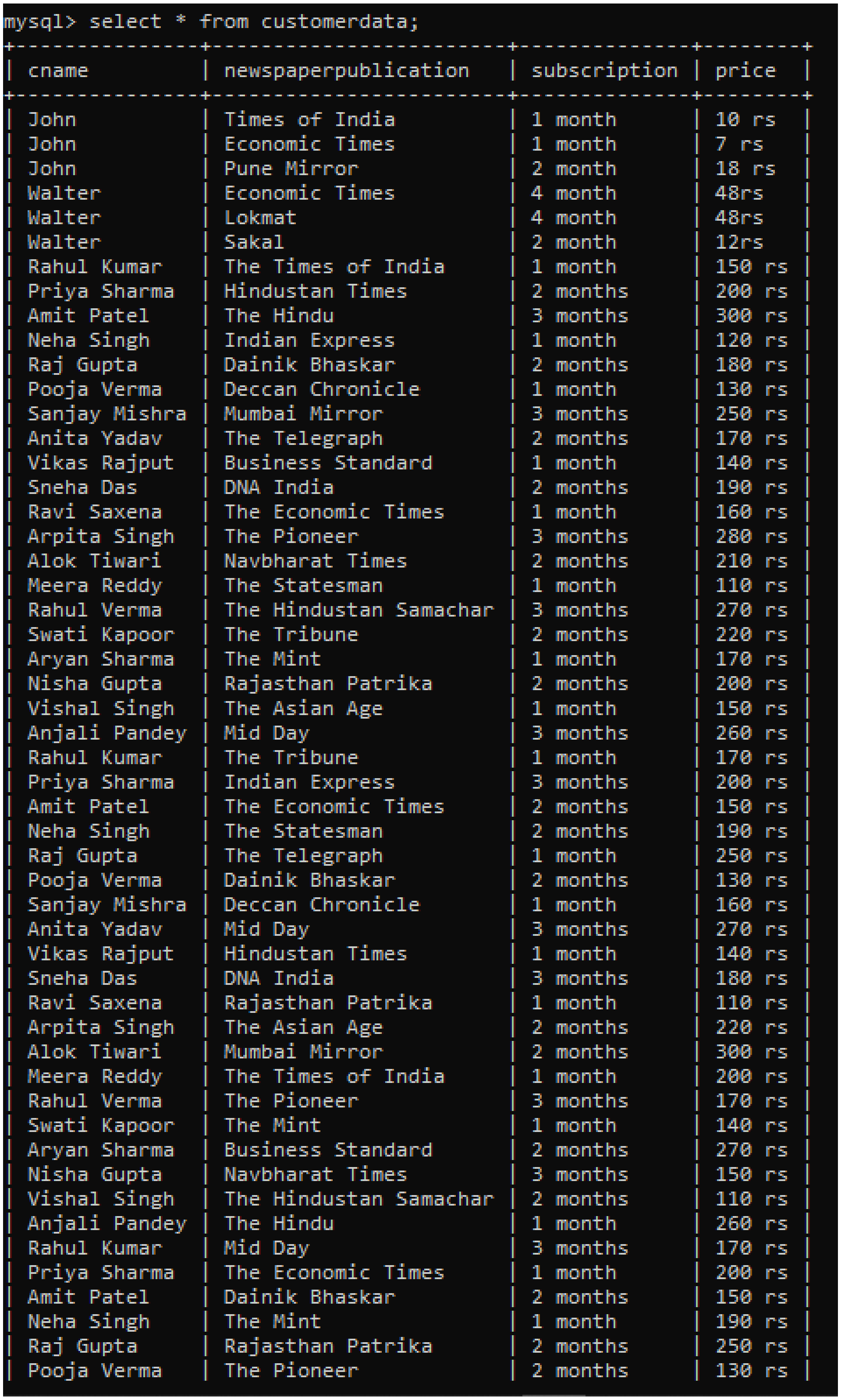


```
**************************************
*             Customer Menu           *
**************************************
1] View Subscriptions
2] View Subscription Duration
3] View Publication
4] View All Subscriptions
5] Add a Subscription
6] Change Username and Password
7] Exit
**************************************
Enter the number: 2
('2 months',)
('3 months',)
('3 months',)
Do you want to repeat? (yes/no)yes
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
```

MYSQL



```
mysql> select * from customerdata;
+---------------+----------------------+--------------+--------+
| cname         | newspaperpublication | subscription | price  |
+---------------+----------------------+--------------+--------+
| John          | Times of India       | 1 month      | 10 rs  |
| John          | Economic Times       | 1 month      | 7 rs   |
| John          | Pune Mirror          | 2 month      | 18 rs  |
| Walter        | Economic Times       | 4 month      | 48rs   |
| Walter        | Lokmat               | 4 month      | 48rs   |
| Walter        | Sakal                | 2 month      | 12rs   |
| Rahul Kumar   | The Times of India   | 1 month      | 150 rs |
| Priya Sharma  | Hindustan Times      | 2 months     | 200 rs |
| Amit Patel    | The Hindu            | 3 months     | 300 rs |
| Neha Singh    | Indian Express       | 1 month      | 120 rs |
| Raj Gupta     | Dainik Bhaskar       | 2 months     | 180 rs |
| Pooja Verma   | Deccan Chronicle     | 1 month      | 130 rs |
| Sanjay Mishra | Mumbai Mirror        | 3 months     | 250 rs |
| Anita Yadav   | The Telegraph        | 2 months     | 170 rs |
| Vikas Rajput  | Business Standard    | 1 month      | 140 rs |
| Sneha Das     | DNA India            | 2 months     | 190 rs |
```



```
mysql> select * from customerdata;
+---------------+------------------------+--------------+--------+
| cname         | newspaperpublication   | subscription | price  |
+---------------+------------------------+--------------+--------+
| John          | Times of India         | 1 month      | 10 rs  |
| John          | Economic Times         | 1 month      | 7 rs   |
| John          | Pune Mirror            | 2 month      | 18 rs  |
| Walter        | Economic Times         | 4 month      | 48rs   |
| Walter        | Lokmat                 | 4 month      | 48rs   |
| Walter        | Sakal                  | 2 month      | 12rs   |
| Rahul Kumar   | The Times of India     | 1 month      | 150 rs |
| Priya Sharma  | Hindustan Times        | 2 months     | 200 rs |
| Amit Patel    | The Hindu              | 3 months     | 300 rs |
| Neha Singh    | Indian Express         | 1 month      | 120 rs |
| Raj Gupta     | Dainik Bhaskar         | 2 months     | 180 rs |
| Pooja Verma   | Deccan Chronicle       | 1 month      | 130 rs |
| Sanjay Mishra | Mumbai Mirror          | 3 months     | 250 rs |
| Anita Yadav   | The Telegraph          | 2 months     | 170 rs |
| Vikas Rajput  | Business Standard      | 1 month      | 140 rs |
| Sneha Das     | DNA India              | 2 months     | 190 rs |
| Ravi Saxena   | The Economic Times     | 1 month      | 160 rs |
| Arpita Singh  | The Pioneer            | 3 months     | 280 rs |
| Alok Tiwari   | Navbharat Times        | 2 months     | 210 rs |
| Meera Reddy   | The Statesman          | 1 month      | 110 rs |
| Rahul Verma   | The Hindustan Samachar | 3 months     | 270 rs |
| Swati Kapoor  | The Tribune            | 2 months     | 220 rs |
| Aryan Sharma  | The Mint               | 1 month      | 170 rs |
| Nisha Gupta   | Rajasthan Patrika      | 2 months     | 200 rs |
| Vishal Singh  | The Asian Age          | 1 month      | 150 rs |
| Anjali Pandey | Mid Day                | 3 months     | 260 rs |
| Rahul Kumar   | The Tribune            | 1 month      | 170 rs |
| Priya Sharma  | Indian Express         | 3 months     | 200 rs |
| Amit Patel    | The Economic Times     | 2 months     | 150 rs |
| Neha Singh    | The Statesman          | 2 months     | 190 rs |
| Raj Gupta     | The Telegraph          | 1 month      | 250 rs |
| Pooja Verma   | Dainik Bhaskar         | 2 months     | 130 rs |
| Sanjay Mishra | Deccan Chronicle       | 1 month      | 160 rs |
| Anita Yadav   | Mid Day                | 3 months     | 270 rs |
| Vikas Rajput  | Hindustan Times        | 1 month      | 140 rs |
| Sneha Das     | DNA India              | 3 months     | 180 rs |
| Ravi Saxena   | Rajasthan Patrika      | 1 month      | 110 rs |
| Arpita Singh  | The Asian Age          | 2 months     | 220 rs |
| Alok Tiwari   | Mumbai Mirror          | 2 months     | 300 rs |
| Meera Reddy   | The Times of India     | 1 month      | 200 rs |
| Rahul Verma   | The Pioneer            | 3 months     | 170 rs |
| Swati Kapoor  | The Mint               | 1 month      | 140 rs |
| Aryan Sharma  | Business Standard      | 2 months     | 270 rs |
| Nisha Gupta   | Navbharat Times        | 3 months     | 150 rs |
| Vishal Singh  | The Hindustan Samachar | 2 months     | 110 rs |
| Anjali Pandey | The Hindu              | 1 month      | 260 rs |
| Rahul Kumar   | Mid Day                | 3 months     | 170 rs |
| Priya Sharma  | The Economic Times     | 1 month      | 200 rs |
| Amit Patel    | Dainik Bhaskar         | 2 months     | 150 rs |
| Neha Singh    | The Mint               | 1 month      | 190 rs |
| Raj Gupta     | Rajasthan Patrika      | 2 months     | 250 rs |
| Pooja Verma   | The Pioneer            | 2 months     | 130 rs |
```

BIBLIOGRAPHY

1.https://www.google.com/

2.https://www.w3schools.com/

3.https://www.geeksforgeeks.org/

4.https://www.youtube.com/

5.NCERT Computer Science Textbook