GOPI BIRLA MEMORIAL SCHOOL

COMPUTER SCIENCE JOURNAL

2024-25

Ashutosh Pawar

Grade XII, Sci

15609856

GOPI BIRLA MEMORIAL SCHOOL

Department of Computer Science

Certificate

This is to certify that the Computer Science Journal has been submitted by the candidate ***Ashutosh Pawar*** with Seat number ***15609856*** for Std XII practical examination of the Central Board of Secondary Education in the year 2024-2025.

It is further certified that this project is the individual work of the candidate.

Date:

Faculty's Signature Examiner's Signature Principal's Signature

INDEX

22/8/24	17	To write records in a CSV file
19/9/24	18	to create a database in Python
19/9/24	19	to create a database in MySQL
19/9/24	29	To create a table in MySQL
19/9/24	29	to show the structure of a table in MySQL
19/9/24	29	to insert values in the table
19/9/24	29	To update values in the table
03/10/24	29	to connect Python with MySQL
03/10/24	29	To insert data in the table of MySQL created in python
03/10/24	30	To retrieve data of the values inserted in a table of MySQL created in python
03/10/24	30	To update data in the table of MySQL created in python
03/10/24	30	To delete data in the table of MySQL created in python
03/10/24	30	To display all the data mysql connectivity
10/10/24	31	To make a menu-driven program for the table of MySQL created, in python

Date: 11/04/24

Aim: A Python code to display me in hr, min and sec

Source Code:

```
hr=r=m=sec=0
t=int(input("Enter the desired time in seconds: "))
hr=t//3600
r=t%3600
m=r//60
sec=r%60
print(hr,"Hour")
print(m,"Minute")
print(“and”,sec,"Seconds")
```

Output:

```
Enter the desired time in seconds: 26369
7 Hour
19 Minute
and 29 Seconds
```

Conclusion: The program was successfully executed..

Date: 11/04/24

Aim: A Python code to calculate and display mean, median and mode

Source Code:

```
import statistics

mean=statistcs.mean([89, 91, 96, 94, 96, 88, 91, 92, 95, 99, 91, 97, 91])

median=statistcs.median([89, 91, 96, 94, 96, 88, 91, 92, 95, 99, 91, 97, 91])

mode=statistics.mode([89, 91, 96, 94, 96, 88, 91, 92, 95, 99, 91, 97, 91])

print("Mean:", mean)

print("Median:", median)

print("Mode:", mode)
```

Output:

```
Mean: 3.07692307692308

Median: 92

Mode: 91
```

Conclusion: The program was successfully executed..

Date: 11/4/24

Aim: To display the first ten terms of the series

Source Code:

```
print("The first ten terms of the series are:")
for i in range(1, 11):
k=i*i+1
print(k, end=' ')
```

Output:

The first ten terms of the series:

2 5 10 17 26 37 50 65 82 101

Conclusion: The program was successfully executed..

Date: 18/4/24

Aim: to check if the entered number is Armstrong or not.

Source Code:

```
n=int(input("Enter a number to check if it is an Armstrong number: "))
sum=0
while(n>0):
    ans=n%10
    sum+=ans*ans*ans
    n=int(n/10)
if sum==n:
    print("It is an Armstrong Number")
else:
    print("Not an Armstrong Number")
```

Output:

```
Enter a number to check if it is an Armstrong number: 420
It is an Armstrong Number
```

Conclusion: The program was successfully executed..

Date: 18/04/24

Aim: to calculate the perimeter of the right-angled triangle

Source Code:

```
import math
pr=int(input("Enter the perpendicular of a triangle:"))
b=int(input("Enter the base of a triangle:"))
hypo=math.sqrt(pr*pr+b*b)
perimtr=pr+b+hypo
print("Hypotenuse =",hypo)
print("Perimeter of triangle =",perimtr)
```

Output:

```
Enter the perpendicular of a triangle:4
Enter the base of a triangle:3
Hypotenuse = 5.0
Perimeter of triangle = 12.0
```

Conclusion: The program was successfully executed..

Date: 25/4/24

Aim: to enter the string and check if it's palindrome or not

Source Code:

```
a=input("Enter any string : ")
nl=a[::-1]
if a==nl:
    print ("Given string is a palindrome")
else:
    print ("Given string is not a palindrome")
```

Output:

```
Enter any string: abcba
Given string is a palindrome
```

Conclusion: The program was successfully executed.

Date: 25/4/24

Aim: A Python code to display the value of Sine 45° and Cosine 60°

Source Code:

```
import math
a=45; b=60
x=22/(7*180)*a
y= 22/(7*180)*b
sin= math.sin(x)
cos= math.cos(y)
print("The value of Sine 45 degree =",sin)
print("The value of Cosine 60 degree =",cos)
```

Output:

```
The value of Sine 45 degree =0.7073302780849811
The value of Cosine 60 degree =0.4996349289865546
```

Conclusion: The program was successfully executed.

Date: 11/7/24

Aim: display the smallest digit of a number

Source Code:

```
n=int(input("Enter a number:"))
nst=str(n)
n=0
for i in nst:
    n=int(i)
    if int(i)<n:
        n=i
    else:
        continue
print("The smallest digit in the number is: ",n)
```

Output:

```
Enter a number:1212540
The smallest digit in the number is:  0
```

Conclusion: The program was successfully executed.

Date: 11/7/24

Aim: to calculate the difference between compound interest and simple interest

Source Code:

```
r1=6
r2=8
r3=10
p=int(input("Enter principal value:"))
amt=p*(1+r1/100)*(1+r2/100)* (1+r3/100)
ci=amt-p
print("Amount =",amt)
print("Compund interest =",ci)
```

Output:

```
Enter principal value:151436
Amount = 190700.32608000003
Compund interest = 39264.32608000003
```

Conclusion: The program was successfully executed.

Date: 18/7/24

Aim: Display the next prime number

Source Code:

```
n=int(input("Enter a number: "))
c=c1=m=0
for i in range(1, n+1):
    if(n%i==0):
        c=c+1
if(c==2):
    print(n, "is a prime number")
else:
    a=n
    m=n
    while(c1!=2):
        a=a+1
        c1=0
        for p in range(1, a+1):
            if(a%p==0):
                c1=c1+1
    print("The next prime number to ", m, " is ",a)
```

Output:

enter a number: 27

The next prime number to 27 is 29

Conclusion: The program was successfully executed.

Date: 18/7/24

Aim: to check a perfect number or automorphic number

Source Code:

```
p=c=k=s=0
print(‘’’Choose and type one of the following choices:
1.to check a perfect number
2.to check an automorphic number’’’)
ch=int(input("enter your choice: "))
if(ch==1):
    n=int(input("enter a number: "))
    for i in range(1, n):
        if(n%i==0):
            s=s+i
    if(s==n):
        print(n, "is a perfect number")
    else:
        print(n,"is not a Perfect Number")
elif(ch==2):
    m=int(input("enter a number"))
    ml-m
    p=m*m
    while(m!=0):
        m=m//10
        c=c+1
```

```
        k=int(math.pow(10,c))
        a=p%k
        if(a==m1):
            print("m1, is an automorphic number")
        else:
            print(ml,"is an not automorphic number")
else:
  print("Wrong choice!")
```

Output:

```
Choose and type one of the following choices:
1.to check a perfect number
2.to check an automorphic number
enter your choice: 1
enter a number: 7
7 is not a Perfect Number
```

Conclusion: The program was successfully executed.

Date: 18/7/24

Aim: To check a happy number

Source Code:

```
s=d=0
n=int(input("Enter a number:"))
s=n
while(s>9):
    n=s
    s=0
    while(n>0):
        d=n%10
        s=s+d*d
        n=n//10
if(s==1):
    print("it is a happy number")
else:
    print("it is not a happy number")
```

Output:

```
Enter a number:31
it is a happy number
```

Conclusion: The program was successfully executed.

Date: 25/7/24

Aim: to illustrate break statement

Source Code:

```
n=int(input("enter a number:"))
print("the digits are:")
while(n>0):
    d=n%10
    print(d)
    if d==0:
        print("zero encounters and the code terminates")
        break
    n=n//10
```

Output:

```
enter a number:40583
the digits are:
3
8
5
0
zero encounters and the code terminates
```

Conclusion: The program was successfully executed.

Date: 25/7/24

Aim: To display a pattern

Source Code:

```
p=7
print("the pattern:")
for i in range(1,6):
for j in range(9,p,-2):
print(j,end=' ')
print()
p=p-2
```

Output:

```
The pattern:
9
9 7
9 7 5
9 7 5 3
9 7 5 3 1
```

Conclusion: The program was successfully executed.

Date: 25/7/24

Aim: To display another type of pattern

Source Code:

```
p=1
print("the pattern:")
for i in range(1,6):
for j in range(1,i+1):
print(p,end=' ')
p=p+1
print()
```

Output:

```
the pattern:
1
2 3
4 5 6
7 8 9 10
11 12 13 14 15
```

Conclusion: The program was successfully executed.

Date: 08/8/24

Aim: menu-driven program to push, pop, peek and display elements

```
a=[]
def push(a):
    value=int(input("Enter element to push:"))
    a.append(value)
    print("element pushed successfully")
def popelement(a):
    x=a.pop()
    print("popped element=",x)
def peek(a):
    print("element at the top=",a[-1])
def display(a):
    for i in range(len(a)-1,-1,-1):
        print(a[i])
while True:
    ch=int(input("""Choose an option from the following:
1)push
2)pop
3)peek
4)display:"""))
    if ch==1:
        push(a)
    elif ch==2:
        if len(a)==0:
```

```
            print("stack overflow")
        else:
            popelement(a)
    elif ch==3:
        if len(a)==0:
            print("stack overflow")
        else:
            peek(a)
    elif ch==4:
        if len(a)==0:
            print("stack overflow")
        else:
            display(a)
    elif ch==5:
        break
    else:
        print("invalid input")
```

Output:

```
Choose an option from the following:
1)push
2)pop
3)peek
4)display:1
```

Enter element to push:2

element pushed successfully

Choose an option from the following:

1)push

2)pop

3)peek

4)display:1

Enter element to push3

element pushed successfully

Choose an option from the following:

1)push

2)pop

3)peek

4)display:1

Enter element to push4

element pushed successfully

Choose an option from the following:

1)push

2)pop

3)peek

4)display:1

Enter element to push5

element pushed successfully

Choose an option from the following:

1)push

2)pop

3)peek

4)display:4

5

4

3

2

Choose an option from the following:

1)push

2)pop

3)peek

4)display:2

popped element= 5

Choose an option from the following:

1)push

2)pop

3)peek

4)display:3

element at the top= 4

Conclusion: The program was successfully executed.

Date: 22/8/24

Aim: To write records in a CSV file

Source Code:

```
import csv
with open("Holiday.csv","w",newline='')as fobj:
    f=csv.writer(fobj)
    f.writerow(['ROLL','NAME','MARKS'])
    while True:
        roll=int(input("enter roll no:"))
        name=input("enter name:")
        marks=int(input("enter marks:"))
        data=[roll,name,marks]
        f.writerow(data)
        ch=int(input("1 enter more\n2 exit\nenter your choice"))
        if ch==2:
            break
    print("file created successfully")
```

Output:

```
enter roll no:1
enter name:mark
enter marks:97
1 enter more
2 exit
```

enter your choice1

enter roll no:2

enter name:george

enter marks:98

1 enter more

2 exit

enter your choice1

enter roll no:3

enter name:faraday

enter marks:99

1 enter more

2 exit

enter your choice2

file created successfully

Conclusion: the file was successfully created

Date: 19/9/24

Aim: to create a database in python

Source Code:

```
import mysql.connector as c
con=c.connect(host='localhost',
user='root',
password='1234')
cur=con.cursor()
cur.execute('create database if not exists comp24')
con.commit()
print("Database created successfully")
```

Output:

Database created successfully

Conclusion: database created successfully.

```
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 12
Server version: 8.0.39 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> create database school24;
Query OK, 1 row affected (0.07 sec)

mysql> use school24;
Database changed
mysql> create table employee(
    -> code integer primary key,
    -> name varchar(30) NOT NULL,
    -> desig varchar(30) NOT NULL,
    -> salary decimal check(salary>10000),
    -> doj date,
    -> state varchar(30),
    -> mob char(10) unique key,
    -> gender char default 'M'
    -> );
Query OK, 0 rows affected (0.16 sec)

mysql> show tables;
+--------------------+
| Tables_in_school24 |
+--------------------+
| employee           |
+--------------------+
1 row in set (0.03 sec)

mysql> desc employee;
+--------+---------------+------+-----+---------+-------+
| Field  | Type          | Null | Key | Default | Extra |
+--------+---------------+------+-----+---------+-------+
| code   | int           | NO   | PRI | NULL    |       |
| name   | varchar(30)   | NO   |     | NULL    |       |
| desig  | varchar(30)   | NO   |     | NULL    |       |
| salary | decimal(10,0) | YES  |     | NULL    |       |
| doj    | date          | YES  |     | NULL    |       |
| state  | varchar(30)   | YES  |     | NULL    |       |
| mob    | char(10)      | YES  | UNI | NULL    |       |
| gender | char(1)       | YES  |     | M       |       |
+--------+---------------+------+-----+---------+-------+
8 rows in set (0.03 sec)

mysql> select * from employee;
Empty set (0.01 sec)

mysql> insert into employee values(101,'Sourish','Manager',50000,'2022-10-11
','Karnataka','12345','M');
Query OK, 1 row affected (0.02 sec)

mysql> insert into employee values(102,'Kosha','Asst Manager',45000,'2020-06-09','Chhattisgarh','12355','F');
Query OK, 1 row affected (0.01 sec)

mysql> insert into employee values(103,'Sameer','Clerk',45000,'2021-06-09','
Punjab','12455','M');
Query OK, 1 row affected (0.01 sec)

mysql> select * from employee;
+------+---------+--------------+--------+------------+--------------+-------+--------+
| code | name    | desig        | salary | doj        | state        | mob   | gender |
+------+---------+--------------+--------+------------+--------------+-------+--------+
|  101 | Sourish | Manager      |  50000 | 2022-10-11 | Karnataka    | 12345 | M      |
|  102 | Kosha   | Asst Manager |  45000 | 2020-06-09 | Chhattisgarh | 12355 | F      |
|  103 | Sameer  | Clerk        |  45000 | 2021-06-09 | Punjab       | 12455 | M      |
+------+---------+--------------+--------+------------+--------------+-------+--------+
3 rows in set (0.00 sec)

mysql> insert into employee values(104,'Rahul','Clerk',44000,'2023-06-09','T


mysql> select * from employee;
Empty set (0.01 sec)

mysql> insert into employee values(101,'Sourish','Manager',50000,'2022-10-11
','Karnataka','12345','M');
Query OK, 1 row affected (0.02 sec)

mysql> insert into employee values(102,'Kosha','Asst Manager',45000,'2020-06-09','Chhattisgarh','12355','F');
Query OK, 1 row affected (0.01 sec)

mysql> insert into employee values(103,'Sameer','Clerk',45000,'2021-06-09','
Punjab','12455','M');
Query OK, 1 row affected (0.01 sec)

mysql> select * from employee;
+------+---------+--------------+--------+------------+--------------+-------+--------+
| code | name    | desig        | salary | doj        | state        | mob   | gender |
+------+---------+--------------+--------+------------+--------------+-------+--------+
|  101 | Sourish | Manager      |  50000 | 2022-10-11 | Karnataka    | 12345 | M      |
|  102 | Kosha   | Asst Manager |  45000 | 2020-06-09 | Chhattisgarh | 12355 | F      |
|  103 | Sameer  | Clerk        |  45000 | 2021-06-09 | Punjab       | 12455 | M      |
+------+---------+--------------+--------+------------+--------------+-------+--------+
3 rows in set (0.00 sec)

mysql> insert into employee values(104,'Rahul','Clerk',44000,'2023-06-09','T
amil Nadu','17455','M');
Query OK, 1 row affected (0.01 sec)

mysql> select code,name,salary from employee;
+------+---------+--------+
| code | name    | salary |
+------+---------+--------+
|  101 | Sourish |  50000 |
|  102 | Kosha   |  45000 |
|  103 | Sameer  |  45000 |
|  104 | Rahul   |  44000 |
+------+---------+--------+
4 rows in set (0.00 sec)
```

```
mysql> select * from employee where code=103;
+------+--------+-------+--------+------------+--------+-------+--------+
| code | name   | desig | salary | doj        | state  | mob   | gender |
+------+--------+-------+--------+------------+--------+-------+--------+
|  103 | Sameer | Clerk |  45000 | 2021-06-09 | Punjab | 12455 | M      |
+------+--------+-------+--------+------------+--------+-------+--------+
1 row in set (0.01 sec)

mysql> select * from employee where desig='clerk';
+------+--------+-------+--------+------------+------------+-------+--------+
| code | name   | desig | salary | doj        | state      | mob   | gender |
+------+--------+-------+--------+------------+------------+-------+--------+
|  103 | Sameer | Clerk |  45000 | 2021-06-09 | Punjab     | 12455 | M      |
|  104 | Rahul  | Clerk |  44000 | 2023-06-09 | Tamil Nadu | 17455 | M      |
+------+--------+-------+--------+------------+------------+-------+--------+
2 rows in set (0.00 sec)

mysql> insert into employee values(105,'Sachin','MTS',43000,'2023-07-09','Madhya Pradesh','18455','M');
Query OK, 1 row affected (0.02 sec)

mysql> insert into employee values(106,'Faraday','MTS',42000,'2021-09-09','Maharashtra','11455','M');
Query OK, 1 row affected (0.01 sec)

mysql> select * from employee;
+------+---------+--------------+--------+------------+----------------+-------+--------+
| code | name    | desig        | salary | doj        | state          | mob   | gender |
+------+---------+--------------+--------+------------+----------------+-------+--------+
|  101 | Sourish | Manager      |  50000 | 2022-10-11 | Karnataka      | 12345 | M      |
|  102 | Kosha   | Asst Manager |  45000 | 2020-06-09 | Chhattisgarh   | 12355 | F      |
|  103 | Sameer  | Clerk        |  45000 | 2021-06-09 | Punjab         | 12455 | M      |
|  104 | Rahul   | Clerk        |  44000 | 2023-06-09 | Tamil Nadu     | 17455 | M      |
|  105 | Sachin  | MTS          |  43000 | 2023-07-09 | Madhya Pradesh | 18455 | M      |
|  106 | Faraday | MTS          |  42000 | 2021-09-09 | Maharashtra    | 11455 | M      |
+------+---------+--------------+--------+------------+----------------+-------+--------+
6 rows in set (0.00 sec)

mysql> select * from employee where salary=45000 and desig='clerk';
+------+--------+-------+--------+------------+--------+-------+--------+
| code | name   | desig | salary | doj        | state  | mob   | gender |
+------+--------+-------+--------+------------+--------+-------+--------+


mysql> select * from employee where salary=45000 and desig='clerk';
+------+--------+-------+--------+------------+--------+-------+--------+
| code | name   | desig | salary | doj        | state  | mob   | gender |
+------+--------+-------+--------+------------+--------+-------+--------+
|  103 | Sameer | Clerk |  45000 | 2021-06-09 | Punjab | 12455 | M      |
+------+--------+-------+--------+------------+--------+-------+--------+
1 row in set (0.01 sec)

mysql> select * from employee where name like=Sa%;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '=
Sa%' at line 1
mysql> select * from employee where name like='Sa%';
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '=
'Sa%'' at line 1
mysql> select * from employee where name like='sa%';
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near '=
'sa%'' at line 1
mysql> select * from employee where name like 'sa%';
+------+--------+-------+--------+------------+----------------+-------+--------+
| code | name   | desig | salary | doj        | state          | mob   | gender |
+------+--------+-------+--------+------------+----------------+-------+--------+
|  103 | Sameer | Clerk |  45000 | 2021-06-09 | Punjab         | 12455 | M      |
|  105 | Sachin | MTS   |  43000 | 2023-07-09 | Madhya Pradesh | 18455 | M      |
+------+--------+-------+--------+------------+----------------+-------+--------+
2 rows in set (0.04 sec)

mysql> update employee set state='Telengana' where code=103;
Query OK, 1 row affected (0.02 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from employee;
+------+---------+--------------+--------+------------+----------------+-------+--------+
| code | name    | desig        | salary | doj        | state          | mob   | gender |
+------+---------+--------------+--------+------------+----------------+-------+--------+
|  101 | Sourish | Manager      |  50000 | 2022-10-11 | Karnataka      | 12345 | M      |
|  102 | Kosha   | Asst Manager |  45000 | 2020-06-09 | Chhattisgarh   | 12355 | F      |
|  103 | Sameer  | Clerk        |  45000 | 2021-06-09 | Telengana      | 12455 | M      |
|  104 | Rahul   | Clerk        |  44000 | 2023-06-09 | Tamil Nadu     | 17455 | M      |
|  105 | Sachin  | MTS          |  43000 | 2023-07-09 | Madhya Pradesh | 18455 | M      |
|  106 | Faraday | MTS          |  42000 | 2021-09-09 | Maharashtra    | 11455 | M      |
+------+---------+--------------+--------+------------+----------------+-------+--------+
6 rows in set (0.00 sec)

mysql>
```

Date: 03/10/24

Aim: to connect python with mysql

Source Code:

```
import mysql.connector as c
con=c.connect(host='localhost',
              user='root',
              password='ashu@105',
              database='pracs')
if con.is_connected():
   print("Successfully connected")
```

Output:

Successfully connected

Conclusion: MySQL was successfully connected with python.

Date: 03/10/24

Aim: To make a menu-driven program for the table of MySQL created, in python.

Source Code:

SQL:

```
CREATE DATABASE library;
USE library;
CREATE TABLE books (
    id INT AUTO_INCREMENT PRIMARY KEY,
    title VARCHAR(255),
    author VARCHAR(255),
    year INT
);
```

Python:

```
import mysql.connector
def connect_to_db():
    return mysql.connector.connect(
        host="localhost",
        user="root",
        password="ashu@105",
        database="librarypracs")

def view_records(cursor):
    cursor.execute("SELECT * FROM books")
```

```
    rows = cursor.fetchall()
    print("\n--- Books Table ---")
    for row in rows:
        print(f"ID: {row[0]}, Title: {row[1]}, Author: {row[2]}, Year: {row[3]}")
    print()

def insert_record(cursor, connection):
    title = input("Enter the book title: ")
    author = input("Enter the book author: ")
    year = int(input("Enter the publication year: "))
    query = "INSERT INTO books (title, author, year) VALUES (%s, %s, %s)"
    cursor.execute(query, (title, author, year))
    connection.commit()
    print("Record inserted successfully!\n")

def update_record(cursor, connection):
    book_id = int(input("Enter the ID of the book to update: "))
    new_title = input("Enter the new title: ")
    new_author = input("Enter the new author: ")
    new_year = int(input("Enter the new publication year: "))
    query = "UPDATE books SET title = %s, author = %s, year = %s WHERE id = %s"
    cursor.execute(query, (new_title, new_author, new_year, book_id))
    connection.commit()
```

```
    print("Record updated successfully!\n")

def delete_record(cursor, connection):
    book_id = int(input("Enter the ID of the book to delete: "))
    query = "DELETE FROM books WHERE id = %s"
    cursor.execute(query, (book_id,))
    connection.commit()
    print("Record deleted successfully!\n")

def main_menu():
    connection = connect_to_db()
    cursor = connection.cursor()

    while True:
        print("Menu:")
        print("1. View Records")
        print("2. Insert Record")
        print("3. Update Record")
        print("4. Delete Record")
        print("5. Exit")

        choice = input("Enter your choice (1-5): ")

        if choice == "1":
            view_records(cursor)
```

```
        elif choice == "2":
            insert_record(cursor, connection)
        elif choice == "3":
            update_record(cursor, connection)
        elif choice == "4":
            delete_record(cursor, connection)
        elif choice == "5":
            print("Exiting program. Goodbye!")
            break
        else:
            print("Invalid choice. Please try again.\n")

    cursor.close()
    connection.close()
```

Output:

```
Menu:
1. View Records
2. Insert Record
3. Update Record
4. Delete Record
5. Exit
Enter your choice (1-5): 1

--- Books Table ---
```

(No records found)

Menu:

1. View Records

2. Insert Record

3. Update Record

4. Delete Record

5. Exit

Enter your choice (1-5): 2

Enter the book title: The Great Gatsby

Enter the book author: F. Scott Fitzgerald

Enter the publication year: 1925

Record inserted successfully!

Menu:

1. View Records

2. Insert Record

3. Update Record

4. Delete Record

5. Exit

Enter your choice (1-5): 1

--- Books Table ---

ID: 1, Title: The Great Gatsby, Author: F. Scott Fitzgerald, Year: 1925

Menu:

1. View Records

2. Insert Record

3. Update Record

4. Delete Record

5. Exit

Enter your choice (1-5): 3

Enter the ID of the book to update: 1

Enter the new title: The Great Gatsby (Updated)

Enter the new author: F. Scott Fitzgerald

Enter the new publication year: 1926

Record updated successfully!

Menu:

1. View Records

2. Insert Record

3. Update Record

4. Delete Record

5. Exit

Enter your choice (1-5): 1

--- Books Table ---

ID: 1, Title: The Great Gatsby (Updated), Author: F. Scott Fitzgerald, Year: 1926

Menu:

1. View Records

2. Insert Record

3. Update Record

4. Delete Record

5. Exit

Enter your choice (1-5): 4

Enter the ID of the book to delete: 1

Record deleted successfully!

Menu:

1. View Records

2. Insert Record

3. Update Record

4. Delete Record

5. Exit

Enter your choice (1-5): 1

--- Books Table ---

(No records found)

Menu:

1. View Records

2. Insert Record

3. Update Record

4. Delete Record

5. Exit

Enter your choice (1-5): 5

Exiting program. Goodbye!

Conclusion: The program was successfully executed.