

## CODE:-

```
// Header file for database connection
import java.sql.*;
import java.util.*;

// Importing files to get date
import java.time.LocalDate;

class DB_connect {
    Connection con = null;
    Statement s;

    // Constructor to connect to database
    DB_connect() {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            con = DriverManager.getConnection("jdbc:mysql://localhost:3307/java_termwork", "root",
"Appu_2003");
        } catch (ClassNotFoundException e) {
            System.out.println("MySQL JDBC driver not found.");
            e.printStackTrace();
        } catch (SQLException e) {
            System.out.println("Error connecting to the database.");
            e.printStackTrace();
        }
    }
}

// Class to manage pooja
class Manage_pooja {

    // Function to insert new pooja type
    void insert_new_pooja() {
        Scanner sc = new Scanner(System.in);
        String pooja_name;
        int rate;
        DB_connect db = new DB_connect();
        try {
            System.out.println();
            System.out.println("=====");
            System.out.println("New Pooja : ");
            System.out.print("Enter Name For New Pooja : ");
            pooja_name = sc.next();
            pooja_name = pooja_name.toLowerCase();
            Statement stmt = db.con.createStatement();
            ResultSet rs = stmt.executeQuery("Select * from pooja_list where pooja_name = '" +
pooja_name + "'");
            if (rs.next()) {
                System.out.println();
                System.out.println("=====");
            }
        }
    }
}
```

```

        System.out.println();
        System.out.println("Pooja Name Already Exists !!!!");
        System.out.println();
        System.out.println("=====");
        return;
    } else {
        System.out.print("Enter Rate Of New Pooja : ");
        rate = sc.nextInt();
        int ins_res = stmt.executeUpdate("Insert into pooja_list values('" + pooja_name + "'," + rate
+ ");");
        if (ins_res > 0) {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println("New Pooja Name Inserted !!!!");
            System.out.println();
            System.out.println("=====");
            return;
        } else {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println("Unable To Insert New Pooja !!!!");
            System.out.println();
            System.out.println("=====");
            return;
        }
    }
} catch (Exception e) {
    System.out.println(e);
}
}

```

```

// Function to update new pooja list
void update_pooja() {
    String old_pooja_name;
    int old_rate;
    DB_connect db = new DB_connect();
    Scanner sc = new Scanner(System.in);
    System.out.println();
    System.out.println("=====");
    System.out.println("Update Pooja : ");
    System.out.print("Enter Old Pooja Name : ");
    old_pooja_name = sc.next();
    old_pooja_name = old_pooja_name.toLowerCase();
    try {
        Statement stmt = db.con.createStatement();
        ResultSet rs = stmt.executeQuery("Select * from pooja_list where pooja_name = '" +
old_pooja_name + "';");
    }
}

```

```

System.out.println("Available Poojas");
// System.out.println("Pooja Name\t Rate");
while (rs.next()) {
    System.out.print(rs.getString(1) + "\t");
    System.out.println(rs.getInt(2));
}
int upd_Choice;
System.out.println();
System.out.println("Enter Column Number You Want To Update ( 1 or 2 ) : ");
upd_Choice = sc.nextInt();
if (upd_Choice == 1) {
    String new_name;
    System.out.println();
    System.out.print("Enter Updated Value For Pooja Name : ");
    new_name = sc.next();
    new_name = new_name.toLowerCase();
    int upd_res = stmt.executeUpdate("Update pooja_list set pooja_name = '" + new_name
        + "' where pooja_name = '" + old_pooja_name + "';");

    if (upd_res > 0) {
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Pooja Name Updated !!!!");
        System.out.println();
        System.out.println("=====");
        return;
    } else {
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Pooja Name Not Updated !!!!");
        System.out.println();
        System.out.println("=====");
        return;
    }
} else if (upd_Choice == 2) {
    int new_rate;
    System.out.println();
    System.out.print("Enter New Price : ");
    new_rate = sc.nextInt();
    int upd_res = stmt.executeUpdate("Update pooja_list set rate = '" + new_rate
        + "' where pooja_name = '" + old_pooja_name + "';");

    if (upd_res > 0) {
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Pooja Name Updated !!!!");
        System.out.println();
        System.out.println("=====");
    }
}

```

```

        return;
    } else {
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Pooja Name Not Updated !!!!");
        System.out.println();
        System.out.println("=====");
        return;
    }
} else {
    System.out.println();
    System.out.println("=====");
    System.out.println();
    System.out.println("Invalid Column Number !!!!");
    System.out.println();
    System.out.println("=====");
    return;
}
} catch (Exception e) {
    System.out.println(e);
}
}
}

```

```

// Function to delete record from pooja_list
void delete_p_list() {
    String old_pooja_name;
    int old_rate;
    DB_connect db = new DB_connect();
    Scanner sc = new Scanner(System.in);
    System.out.println();
    System.out.println("=====");
    System.out.println("Delete Pooja : ");
    System.out.print("Enter Pooja Name : ");
    old_pooja_name = sc.next();
    old_pooja_name = old_pooja_name.toLowerCase();
    try {
        Statement stmt = db.con.createStatement();
        ResultSet rs = stmt.executeQuery("Select * from pooja_list where pooja_name = '" +
old_pooja_name + "';");

        int del_res = stmt.executeUpdate("Delete from pooja_list where pooja_name = '" +
old_pooja_name + "';");
        if (del_res > 0) {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println(old_pooja_name + " Pooja Deleted !!!!");
            System.out.println();
            System.out.println("=====");
            return;
        }
    }
}

```

```

        } else {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println(old_pooja_name + " Pooja Not Deleted !!!");
            System.out.println();
            System.out.println("=====");
            return;
        }
    } catch (Exception e) {
        System.out.println(e);
    }
}

void show_all_pooja() {
    DB_connect db = new DB_connect();
    try {
        Statement stmt = db.con.createStatement();
        ResultSet rs = stmt.executeQuery("Select * from pooja_list;");

        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Available Poojas");
        System.out.println();
        while (rs.next()) {
            System.out.print(rs.getString(1) + "\t");
            System.out.println(rs.getInt(2));
        }
        System.out.println();
        System.out.println("=====");

    } catch (Exception e) {
        System.out.println(e);
    }
}

// Function to display all records
void show_all() {
    DB_connect db = new DB_connect();
    try {
        Statement stmt = db.con.createStatement();
        ResultSet result = stmt.executeQuery("Select * From all_pooja_records");

        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("All Pooja Records : ");
        System.out.println();
        // System.out.println("Sr.no.\tPooja\tRate\tDevotee\tStar\tDate\tCreated At");
        while (result.next()) {

```

```

        System.out.print(result.getInt(1) + "\t");
        System.out.print(result.getString(2) + "\t");
        System.out.print(result.getInt(3) + "\t");
        System.out.print(result.getString(4) + "\t");
        System.out.print(result.getString(5) + "\t");
        System.out.print(result.getString(6) + "\t");
        System.out.println(result.getString(7));
    }
    System.out.println();
    System.out.println("=====");

} catch (Exception e) {
    System.out.println(e);
}
}

// Function to display total bill of the devotee
void display_bill(int total) {
    DB_connect db = new DB_connect();
    // Displaying total bill of the devotee
    try {
        Statement stmt = db.con.createStatement();
        ResultSet result1 = stmt.executeQuery("Select * from temp_records");
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Total Bill : ");
        System.out.println();
        // System.out.println("Sr.no.\tPooja\tRate\tDevotee\tStar\tDate");
        while (result1.next()) {
            System.out.print(result1.getInt(1) + "\t");
            System.out.print(result1.getString(2) + "\t");
            System.out.print(result1.getInt(3) + "\t");
            System.out.print(result1.getString(4) + "\t");
            System.out.print(result1.getString(5) + "\t");
            System.out.println(result1.getString(6));
        }
        System.out.println();
        System.out.println("Total Amount : " + total);
        System.out.println();
        System.out.println("=====");
        stmt.executeUpdate("Drop table temp_records");
    } catch (Exception e) {
        System.out.println(e);
    }
}

int devotee_details(String devotee_name, String created_at) {
    DB_connect db = new DB_connect();
    try {
        Statement stmt = db.con.createStatement();

```

```

        ResultSet result = stmt.executeQuery("Select * from all_pooja_records where devotee_name
= " + devotee_name
        + " and created_at = " + created_at + " ");

```

```

        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Devotee Details : ");
        System.out.println();
        // System.out.println("Sr.no.\tPooja\tRate\tDevotee\tStar\tDate");
        while (result.next()) {
            System.out.print(result.getInt(1) + "\t");
            System.out.print(result.getString(2) + "\t");
            System.out.print(result.getInt(3) + "\t");
            System.out.print(result.getString(4) + "\t");
            System.out.print(result.getString(5) + "\t");
            System.out.println(result.getString(6));
        }

```

```

        System.out.println();
        System.out.println("=====");
        return 1;

```

```

    } catch (Exception e) {
        System.out.println(e);
        return 0;
    }
}

```

// Funtion where multiple pooja is to be inputed for one devotee

```

void one_dev_mul_pooja() {
    String pooja_name, devotee_name, birth_star;
    char Choice;
    Scanner sc = new Scanner(System.in);

```

```

    System.out.println();
    System.out.println("=====");
    System.out.println();
    System.out.print("Enter Devotee Name : ");
    devotee_name = sc.next();
    devotee_name = devotee_name.toLowerCase();
    System.out.print("Enter Devotee Birth Star : ");
    birth_star = sc.next();
    birth_star = birth_star.toLowerCase();
    DB_connect db = new DB_connect();

```

```

// Exception handling of the database
try {

```

```

    Statement stmt = db.con.createStatement();

```

```

// Checking if the birth star is valid or not
ResultSet result = stmt.executeQuery("Select star from star_list where star = '" + birth_star +
    """);

    if (!result.next()) {
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Invalid Birth Star Inputed !!!!");
        System.out.println("Pooja Not Added !!!!");
        System.out.println();
        System.out.println("=====");
        System.out.println();
        return;

    } else {
        int total = 0;
        int count = 0;
        Statement stmt1 = db.con.createStatement();
        stmt1.executeUpdate(
            "create table temp_records(pooja_id int NOT NULL AUTO_INCREMENT PRIMARY KEY,
pooja_name varchar(50) NOT NULL, rate int NOT NULL, devotee_name varchar(50) NOT NULL,
birth_star varchar(30) NOT NULL, pooja_date varchar(50) NOT NULL);");
        while (true) {
            count += 1;
            System.out.println("Enter Pooja Name : " + count);
            pooja_name = sc.next();
            pooja_name = pooja_name.toLowerCase();
            try {
                ResultSet result1 = stmt.executeQuery("Select * from pooja_list where pooja_name ='"
+ pooja_name + """);

                int rate = 0;
                while (result1.next()) {
                    rate = result1.getInt(2);
                }
                char spe_date;
                String pooja_date;
                System.out.println("Specific date For Pooja ?\nY -> Yes\nN -> No");
                spe_date = sc.next().charAt(0);
                spe_date = Character.toLowerCase(spe_date);
                if (spe_date == 'y') {
                    System.out.print("Enter Date In YYYY-MM-DD format");
                    pooja_date = sc.next();
                } else {
                    LocalDate date = LocalDate.now();
                    pooja_date = date.toString();
                }

                LocalDate today = LocalDate.now();
                String created_at = today.toString();

```



```

        // Inserting in the temporary table
        stmt1.executeUpdate(
            "Insert into temp_records
(pooja_name,rate,devotee_name,birth_star,pooja_date) values('"
            + pooja_name + "','" + rate + "','" + devotee_name + "','" + birth_star
            + "','" + pooja_date + "');"");

        // Inserting as permanenet record
        int ins = stmt.executeUpdate(
            "Insert into all_pooja_records
(pooja_name,rate,devotee_name,birth_star,pooja_date,created_at) values('"
            + pooja_name + "','" + rate + "','" + devotee_name + "','" + birth_star
            + "','" + pooja_date + "','" + created_at + "');"");
        if (ins > 0) {
            System.out.println();
            total += rate;
            System.out.println("Pooja Added.");
            System.out.print("Add More Pooja?\nY -> Yes.\nN -> No.");
            Choice = sc.next().charAt(0);
            Choice = Character.toLowerCase(Choice);
            if (Choice == 'n') {
                break;
            }
        } else {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println("Unable To Add Pooja !!!!");
            System.out.println();
            System.out.println("=====");
            System.out.println();
        }
    } catch (Exception e) {
        System.out.println(e);
    }
}
display_bill(total);
}
} catch (Exception e) {
    System.out.println(e);
}
}
}

```

// Funtion where one pooja has multiple devotees

```

void one_pooja_mul_dev() {
    String pooja_name, devotee_name, birth_star;
    char Choice;
    Scanner sc = new Scanner(System.in);
    System.out.println();
}

```

```

System.out.println("=====");
System.out.println();
System.out.print("Enter Pooja Name : ");
pooja_name = sc.next();
pooja_name = pooja_name.toLowerCase();
DB_connect db = new DB_connect();
try {
    Statement stmt = db.con.createStatement();
    ResultSet result = stmt
        .executeQuery("Select * from pooja_list where pooja_name = '" + pooja_name + "'");
    int rate = 0;
    while (result.next()) {
        rate = result.getInt(2);
    }
    char spe_date;
    String pooja_date;
    int total = 0, count = 0;
    Statement stmt1 = db.con.createStatement();
    stmt1.executeUpdate(
        "create table temp_records(pooja_id int NOT NULL AUTO_INCREMENT PRIMARY KEY,
pooja_name varchar(50) NOT NULL, rate int NOT NULL, devotee_name varchar(50) NOT NULL,
birth_star varchar(30) NOT NULL, pooja_date varchar(50) NOT NULL);");
    while (true) {
        count += 1;
        System.out.println();
        System.out.println("Enter Devotee Name : " + count);
        devotee_name = sc.next();
        devotee_name = devotee_name.toLowerCase();
        System.out.print("Enter Devotee Birth Star : ");
        birth_star = sc.next();
        birth_star = birth_star.toLowerCase();
        ResultSet result1 = stmt
            .executeQuery("Select star from star_list where star = '" + birth_star + "'");

        if (!result1.next()) {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println("Invalid Birth Star Inputed !!!!");
            System.out.println("Pooja Not Added !!!!");
            System.out.println();
            System.out.println("=====");
            System.out.println();
            return;
        } else {
            System.out.println("Specific date For Pooja ?\nY -> Yes\nN -> No");
            spe_date = sc.next().charAt(0);

            if (spe_date == 'y') {
                System.out.print("Enter Date In YYYY-MM-DD format");
            }
        }
    }
}

```

```

        pooja_date = sc.next();
    } else {
        LocalDate date = LocalDate.now();
        pooja_date = date.toString();
    }

    LocalDate today = LocalDate.now();
    String created_at = today.toString();

    // Inserting in the temporary table
    stmt1.executeUpdate(
        "Insert into temp_records (pooja_name,rate,devotee_name,birth_star,pooja_date)
values("
        + pooja_name + "," + rate + "," + devotee_name + "," + birth_star + ","
        + pooja_date + ");");

    // Inserting as permanenet record
    int ins = stmt.executeUpdate(
        "Insert into all_pooja_records
(pooja_name,rate,devotee_name,birth_star,pooja_date,created_at) values("
        + pooja_name + "," + rate + "," + devotee_name + "," + birth_star + ","
        + pooja_date + "," + created_at + ");");

    if (ins > 0) {
        System.out.println();
        total += rate;
        System.out.println("Devotee Added.");
        System.out.println("Add More Pooja?\nY -> Yes.\nN -> No.");
        Choice = sc.next().charAt(0);
        Choice = Character.toLowerCase(Choice);
        if (Choice == 'n') {
            break;
        }
    } else {
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Unable To Add Pooja !!!!");
        System.out.println();
        System.out.println("=====");
        System.out.println();
    }
}
}
}
display_bill(total);
} catch (Exception e) {
    System.out.println(e);
}
}

```

```

// Function to update pooja list
void update_records() {
    String devotee_name, created_date;

    Scanner sc = new Scanner(System.in);
    System.out.println();
    System.out.println("=====");
    System.out.println();
    System.out.println("Details For Update : ");
    System.out.print("Enter Devotee Name : ");
    devotee_name = sc.next();
    devotee_name = devotee_name.toLowerCase();
    System.out.print("Enter Entry Date (YYYY-MM-DD) : ");
    created_date = sc.next();

    // Checking if details of the devotee is present or not
    if ((devotee_details(devotee_name, created_date) > 0)) {
        int col_num, pooja_id;
        System.out.print("Enter Sr.no Of The Record You Want To Update : ");
        pooja_id = sc.nextInt();

        System.out.print("Enter Column Number Of The Record (Valid Inputs : 2,4,5,6) : ");
        col_num = sc.nextInt();

        try {
            DB_connect db = new DB_connect();
            Statement stmt = db.con.createStatement();
            if (col_num == 2) {
                String new_pooja_name;
                System.out.println();
                System.out.print("Enter New Pooja Name : ");
                new_pooja_name = sc.next();
                new_pooja_name = new_pooja_name.toLowerCase();
                int new_rate = 0;
                ResultSet result2 = stmt
                    .executeQuery("Select rate from pooja_list where pooja_name = '" +
new_pooja_name + "';");
                while (result2.next()) {
                    new_rate = result2.getInt(1);
                }

                System.out.println("Hi");
                int up_pooja_name = stmt.executeUpdate("Update all_pooja_records set pooja_name
= '"
                    + new_pooja_name + "' where pooja_id = " + pooja_id + "';");
                if (up_pooja_name > 0) {
                    int up_rate = stmt.executeUpdate("Update all_pooja_records set rate = " + new_rate
                        + " where pooja_id = " + pooja_id);
                    if (up_rate > 0) {
                        System.out.println();
                    }
                }
            }
        }
    }
}

```

```

        System.out.println("=====");
        System.out.println();
        System.out.println("Pooja Name of " + pooja_id + " Updated !!!!");
        System.out.println();
        System.out.println("=====");
        System.out.println();
        return;
    } else {
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Pooja Name of " + pooja_id + " Not Updated !!!!");
        System.out.println();
        System.out.println("=====");
        System.out.println();
        return;
    }
} else {
    System.out.println();
    System.out.println("=====");
    System.out.println();
    System.out.println("Pooja Name of " + pooja_id + " Not Updated !!!!");
    System.out.println();
    System.out.println("=====");
    System.out.println();
    return;
}
} else if (col_num == 4) {
    String new_dev_name;
    System.out.print("Enter New Value For The Devotee Name : ");
    new_dev_name = sc.next();
    new_dev_name = new_dev_name.toLowerCase();
    int upd_res = stmt.executeUpdate("Update all_pooja_records set devotee_name = '" +
new_dev_name
        + "' where pooja_id = " + pooja_id + ";");

    if (upd_res > 0) {
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Devotee Name of " + pooja_id + " Updated");
        System.out.println();
        System.out.println("=====");
        System.out.println();
        return;
    } else {
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Devotee Name of " + pooja_id + " Not Updated");
        System.out.println();
    }
}

```

```

        System.out.println("=====");
        System.out.println();
        return;
    }
} else if (col_num == 5) {
    String new_star;
    System.out.print("Enter New Value For Birth Star : ");
    new_star = sc.next();
    new_star = new_star.toLowerCase();
    ResultSet result = stmt.executeQuery("Select * from star_list where star = '" + new_star +
";");

    if (!result.next()) {
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Invalid Birth Star !!!");
        System.out.println();
        System.out.println("=====");
        System.out.println();
        return;
    } else {
        int upd_res = stmt.executeUpdate("Update all_pooja_records set birth_star = '" +
new_star
        + "' where pooja_id = " + pooja_id + ";");

        if (upd_res > 0) {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println("Birth Star of " + pooja_id + " Updated");
            System.out.println();
            System.out.println("=====");
            System.out.println();
            return;
        } else {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println("Birth Star of " + pooja_id + " Not Updated");
            System.out.println();
            System.out.println("=====");
            System.out.println();
            return;
        }
    }
} else if (col_num == 6) {
    String new_date;
    System.out.print("Enter New Pooja Date (YYYY-MM-DD) format: ");
    new_date = sc.next();

```

```
int upd_res = stmt.executeUpdate("Update all_pooja_records set pooja_date = '" +  
new_date + "' where pooja_id = " + pooja_id + ";");
```

```
if (upd_res > 0) {  
    System.out.println();  
    System.out.println("=====");  
    System.out.println();  
    System.out.println("Pooja Date of " + pooja_id + " Updated");  
    System.out.println();  
    System.out.println("=====");  
    System.out.println();  
    return;  
} else {  
    System.out.println();  
    System.out.println("=====");  
    System.out.println();  
    System.out.println("Pooja Date of " + pooja_id + " Not Updated");  
    System.out.println();  
    System.out.println("=====");  
    System.out.println();  
  
    return;  
}
```

```
} else {  
    System.out.println();  
    System.out.println("=====");  
    System.out.println();  
    System.out.println("Invalid Column Number !!!!");  
    System.out.println();  
    System.out.println("=====");  
    System.out.println();  
  
    return;  
}  
} catch (Exception e) {  
    System.out.println(e);  
}
```

```
}
```

```
}
```

```
// Function to delete a record
```

```
void delete_pooja() {  
    Scanner sc = new Scanner(System.in);  
    String devotee_name, created_date;  
  
    System.out.println();  
    System.out.println("=====");
```

```

System.out.println();
System.out.println("Details For Delete : ");
System.out.print("Enter Devotee Name : ");
devotee_name = sc.next();
devotee_name = devotee_name.toLowerCase();
System.out.print("Enter Entry Date (YYYY-MM-DD) : ");
created_date = sc.next();

if ((devotee_details(devotee_name, created_date) > 0)) {
    DB_connect db = new DB_connect();
    try {
        Statement stmt = db.con.createStatement();
        int pooja_id;
        System.out.print("Enter Sr.no Of The Record You Want To Delete : ");
        pooja_id = sc.nextInt();
        int del_res = stmt.executeUpdate("Delete From all_pooja_records where pooja_id = " +
pooja_id + ";");

        if (del_res > 0) {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println(pooja_id + " Entry Deleted !!!!");
            System.out.println();
            System.out.println("=====");
            System.out.println();
            return;
        } else {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println(pooja_id + " Entry Not Deleted !!!!");
            System.out.println();
            System.out.println("=====");
            System.out.println();
            return;
        }
    } catch (Exception e) {
        System.out.println(e);
    }
}

}

}

// Class to manage Donations
class Manage_donation {
    // View All Donation Records
    void all_donatins() {

```



```

DB_connect db = new DB_connect();
try {
    Statement stmt = db.con.createStatement();
    ResultSet result = stmt.executeQuery("Select * From all_donations");
    System.out.println();
    System.out.println("=====");
    System.out.println();
    System.out.println("All Donations : ");
    System.out.println();
    // System.out.println("Sr.no.\tDonar Name\tAmount\tCreated_at");
    while (result.next()) {
        System.out.print(result.getInt(1) + "\t");
        System.out.print(result.getString(2) + "\t");
        System.out.print(result.getInt(3) + "\t");
        System.out.print(result.getString(4));
    }
    System.out.println();
    System.out.println("=====");

} catch (Exception e) {
    System.out.println(e);
}
}

// Get Records For update and delete
int donar_details(String donar_name, String created_at) {
    DB_connect db = new DB_connect();
    try {
        Statement stmt = db.con.createStatement();
        ResultSet result = stmt.executeQuery("Select * from all_donations where donar_name = '" +
donar_name
        + "' and created_at = '" + created_at + "'");

        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Donar Details : ");
        System.out.println();
        // System.out.println("Sr.no.\tDonar Name\tAmount");
        while (result.next()) {
            System.out.print(result.getInt(1) + "\t");
            System.out.print(result.getString(2) + "\t");
            System.out.print(result.getInt(3) + "\t");
        }

        System.out.println();
        System.out.println("=====");
        return 1;
    } catch (Exception e) {
        System.out.println(e);
    }
}

```

```

        return 0;
    }
}

// Function to insert new donation
void add_donation() {
    String donar_name;
    int amount;

    Scanner sc = new Scanner(System.in);

    LocalDate date = LocalDate.now();
    String created_at = date.toString();

    System.out.println();
    System.out.println("=====");
    System.out.println();
    System.out.print("Enter Donar Name : ");
    donar_name = sc.next();
    donar_name = donar_name.toLowerCase();
    System.out.print("Enter Donation Amount : ");
    amount = sc.nextInt();

    DB_connect db = new DB_connect();

    try {
        Statement stmt = db.con.createStatement();
        int don_ins = stmt.executeUpdate("Insert into all_donations
(donar_name,amount,created_at) values("
        + donar_name + " , " + amount + " , " + created_at + "));");

        if (don_ins > 0) {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println("Donation Added !!!!");
            System.out.println();
            System.out.println("=====");
            System.out.println();

            return;
        } else {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println("Donation Not Added !!!!");
            System.out.println();
            System.out.println("=====");
            System.out.println();
            return;
        }
    }
}

```

```

    } catch (Exception e) {
        System.out.println(e);
    }
}

void update_donation() {
    String donar_name, created_date;

    Scanner sc = new Scanner(System.in);
    System.out.println();
    System.out.println("=====");
    System.out.println();
    System.out.println("Details For Update : ");
    System.out.print("Enter Donar Name : ");
    donar_name = sc.next();
    donar_name = donar_name.toLowerCase();
    System.out.print("Enter Entry Date (YYYY-MM-DD) : ");
    created_date = sc.next();

    if (donar_details(donar_name, created_date) > 0) {
        int col_num, donation_id;
        System.out.print("Enter Sr.no Of The Record You Want To Update : ");
        donation_id = sc.nextInt();

        System.out.print("Enter Column Number Of The Record (Valid Inputs : 2,3) : ");
        col_num = sc.nextInt();

        try {
            DB_connect db = new DB_connect();
            Statement stmt = db.con.createStatement();
            if (col_num == 2) {
                String new_donar_name;
                System.out.println();
                System.out.print("Enter New Value For Donar Name : ");
                new_donar_name = sc.next();
                new_donar_name = new_donar_name.toLowerCase();
                int upd_donar = stmt.executeUpdate("Update all_donations set donar_name = '" +
new_donar_name
                + "' where donation_id = " + donation_id + "");

                if (upd_donar > 0) {
                    System.out.println();
                    System.out.println("=====");
                    System.out.println();
                    System.out.println("Donation Record Updated !!!!");
                    System.out.println();
                    System.out.println("=====");
                    System.out.println();
                    return;
                } else {
                    System.out.println();

```

```

        System.out.println("=====");
        System.out.println();
        System.out.println("Donation Record Not Updated !!!!");
        System.out.println();
        System.out.println("=====");
        System.out.println();
        return;
    }
}
} catch (Exception e) {
    System.out.println(e);
}
}
}
}

```

// Function to delete donation record

```

void delete_donation() {
    Scanner sc = new Scanner(System.in);
    String donar_name, created_date;

    System.out.println();
    System.out.println("=====");
    System.out.println();
    System.out.println("Details For Delete : ");
    System.out.print("Enter Donar Name : ");
    donar_name = sc.next();
    donar_name = donar_name.toLowerCase();
    System.out.print("Enter Entry Date (YYYY-MM-DD) : ");
    created_date = sc.next();

    if ((donar_details(donar_name, created_date)) > 0) {
        DB_connect db = new DB_connect();

        try {
            Statement stmt = db.con.createStatement();
            int donation_id;
            System.out.print("Enter Sr.no. Of The Record You Want To Delete : ");
            donation_id = sc.nextInt();

            int del_res = stmt.executeUpdate("Delete From all_donations where donation_id = " +
donation_id + ";");

            if (del_res > 0) {
                System.out.println();
                System.out.println("=====");
                System.out.println();
                System.out.println(donation_id + " Entry Deleted !!!!");
                System.out.println();
                System.out.println("=====");
                System.out.println();
                return;
            }
        }
    }
}

```

```

        } else {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println(donation_id + " Entry Not Deleted !!!!");
            System.out.println();
            System.out.println("=====");
            System.out.println();
            return;
        }
    } catch (Exception e) {
        System.out.println(e);
    }
}

}
}

class Manage_users {
    // Function to login
    void login() {
        Scanner sc = new Scanner(System.in);
        String username, password;
        DB_connect db = new DB_connect();
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Login");
        System.out.println();
        System.out.print("Enter Username : ");
        username = sc.next();
        System.out.print("Enter Password : ");
        password = sc.next();

        try {
            Statement stmt = db.con.createStatement();
            ResultSet result = stmt.executeQuery("Select username, password from all_users");
            int found = 0;
            while (result.next()) {
                if (username.equals(result.getString(1))) {
                    if (password.equals(result.getString(2))) {
                        System.out.println();
                        System.out.println("=====");
                        System.out.println();
                        System.out.println(" Welcome " + username);
                        System.out.println();
                        System.out.println("=====");
                        System.out.println();
                        found = 1;
                        return;
                    }
                }
            }
        }
    }
}

```

```

        } else {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println(" Incorrect Login Details !!!!");
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println();
            System.exit(0);
            return;
        }
    }
    if (found == 0) {
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println(" User Not Found !!!!");
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println();
        System.exit(0);
        return;
    }
} catch (Exception e) {
    System.out.println(e);
}
}

// Function to update username
void update_username() {
    Scanner sc = new Scanner(System.in);
    String old_password, old_username;
    DB_connect db = new DB_connect();
    System.out.println();
    System.out.println("=====");
    System.out.println();
    System.out.print("Enter Current Username : ");
    old_username = sc.next();
    System.out.print("Enter Current Password : ");
    old_password = sc.next();
    try {
        Statement stmt = db.con.createStatement();
        ResultSet result = stmt.executeQuery("Select username, password from all_users;");
        int found = 0;
        while (result.next()) {
            if (old_username.equals(result.getString(1))) {
                if (old_password.equals(result.getString(2))) {
                    String new_username;
                    System.out.println();
                    System.out.print("Enter New Username : ");
                    new_username = sc.next();
                }
            }
        }
    }
}

```

```

        int upd_usr = stmt.executeUpdate("Update all_users set username = '" +
new_username
        + "' where username ='" + old_username + "';");
        if (upd_usr > 0) {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println(" Username Updated !!!!");
            System.out.println();
            System.out.println("=====");
            System.out.println();
            found = 1;
            return;
        } else {
            System.out.println();
            System.out.println("=====");
            System.out.println();
            System.out.println(" Username Not Updated !!!!");
            System.out.println();
            System.out.println("=====");
            System.out.println();
            return;
        }
    } else {
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println(" Incorrect Login Details !!!!");
        System.out.println();
        System.out.println("=====");
        System.out.println();
        return;
    }
} else {
    System.out.println();
    System.out.println("=====");
    System.out.println();
    System.out.println(" Incorrect Login Details !!!!");
    System.out.println();
    System.out.println("=====");
    System.out.println();
    return;
}
}
if (found == 0) {
    System.out.println();
    System.out.println("=====");
    System.out.println();
    System.out.println(" User Not Found !!!!");
    System.out.println();
    System.out.println("=====");
}

```

```

        System.out.println();
        return;
    }
} catch (Exception e) {
    System.out.println(e);
}
}

// Function to update password
void update_password() {
    Scanner sc = new Scanner(System.in);
    String old_password, old_username;
    DB_connect db = new DB_connect();
    System.out.println();
    System.out.println("=====");
    System.out.println();
    System.out.print("Enter Current Username : ");
    old_username = sc.next();
    System.out.print("Enter Current Password : ");
    old_password = sc.next();
    try {
        Statement stmt = db.con.createStatement();
        ResultSet result = stmt.executeQuery("Select username, password from all_users;");
        while (result.next()) {
            if (old_username.equals(result.getString(1))) {
                if (old_password.equals(result.getString(2))) {
                    String new_password;
                    System.out.println();
                    System.out.print("Enter New Password : ");
                    new_password = sc.next();

                    int upd_pwd = stmt.executeUpdate("Update all_users set password = '" +
new_password
                    + "' where username ='" + old_username + "';");
                    if (upd_pwd > 0) {
                        System.out.println();
                        System.out.println("=====");
                        System.out.println();
                        System.out.println(" Password Updated !!!!");
                        System.out.println();
                        System.out.println("=====");
                        System.out.println();
                        return;
                    } else {
                        System.out.println();
                        System.out.println("=====");
                        System.out.println();
                        System.out.println(" Password Not Updated !!!!");
                        System.out.println();
                        System.out.println("=====");
                        System.out.println();
                        return;
                    }
                }
            }
        }
    }
}

```



```

    }
    }
    }
    } catch (Exception e) {
        System.out.println(e);
    }
}
}

```

// Class for main function

```

public class Temple_management {
    public static void main(String[] args) {
        Manage_pooja pooja = new Manage_pooja();
        Manage_donation donation = new Manage_donation();
        Manage_users users = new Manage_users();
        // Greetings
        System.out.println("=====");
        System.out.println();
        System.out.println("SHREE AYYAPPA SEVA SAMITI TRUST");
        // Creating object of user management class
        Manage_users user = new Manage_users();
        // Initiating Login
        user.login();
        int Choice;
        Scanner sc = new Scanner(System.in);
        while (true) {
            System.out.println("=====");
            System.out.println();
            System.out.println(
                "Enter Choice : \n1 -> View Pooja Menu Section.\n2 -> View Donation Menu.\n3 -> User
Management Menu.");
            Choice = sc.nextInt();
            if (Choice == 1) {
                int pooja_menu;
                System.out.println();
                System.out.println("=====");
                System.out.println("Pooja Menu");
                System.out.println();
                System.out.println(
                    "Enter Choice : \n1 -> Add By Pooja Name.\n2 -> Add By Devotee Name\n3 -> Update
Pooja Records\n4 -> Delete Pooja Records\n5 -> Add New Pooja Type\n6 -> Update Available Pooja
Type\n7 -> Delete Available Pooja Type\n8 -> View All Available Poojas Type\n9 -> View All Devotee
Poojas");
                pooja_menu = sc.nextInt();
                if (pooja_menu == 1) {
                    pooja.one_pooja_mul_dev();
                } else if (pooja_menu == 2) {
                    pooja.one_dev_mul_pooja();
                } else if (pooja_menu == 3) {
                    pooja.update_records();
                }
            }
        }
    }
}

```

```

    } else if (pooja_menu == 4) {
        pooja.delete_pooja();
    } else if (pooja_menu == 5) {
        pooja.insert_new_pooja();
    } else if (pooja_menu == 6) {
        pooja.update_pooja();
    } else if (pooja_menu == 7) {
        pooja.delete_p_list();
    } else if (pooja_menu == 8) {
        pooja.show_all_pooja();
    } else if (pooja_menu == 9) {
        pooja.show_all();
    } else {
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Exiting Pooja Menu !!!!");
        System.out.println();
        System.out.println("=====");
        System.out.println();
    }
} else if (Choice == 2) {
    int donation_menu;
    System.out.println();
    System.out.println("=====");
    System.out.println("Donation Management Menu");
    System.out.println();
    System.out.println(
        "Enter Choice : \n1 -> Add Donation\n2 -> Update Donation\n3 -> Delete Donation\n4 -
> Show All Donations");
    donation_menu = sc.nextInt();
    if (donation_menu == 1) {
        donation.add_donation();
    } else if (donation_menu == 2) {
        donation.update_donation();
    } else if (donation_menu == 3) {
        donation.delete_donation();
    } else if (donation_menu == 4) {
        donation.all_donations();
    } else {
        System.out.println();
        System.out.println("=====");
        System.out.println();
        System.out.println("Exiting Donation Management Menu !!!!");
        System.out.println();
        System.out.println("=====");
        System.out.println();
    }
} else if (Choice == 3) {
    int user_menu;
    System.out.println();

```

```

System.out.println("=====");
System.out.println("User Management Menu");
System.out.println();
System.out.println("Enter Choice : \n1 -> Update Username\n2 -> Update Password");
user_menu = sc.nextInt();

if (user_menu == 1) {
    user.update_username();
} else if (user_menu == 2) {
    user.update_password();
} else {
    System.out.println();
    System.out.println("=====");
    System.out.println();
    System.out.println("Exiting User Management Menu !!!!");
    System.out.println();
    System.out.println("=====");
    System.out.println();
}
} else {
    System.out.println();
    System.out.println("=====");
    System.out.println();
    System.out.println("Exiting !!!!");
    System.out.println("Thank You For Using Temple Management !!!!");
    System.out.println();
    System.out.println("=====");
    System.out.println();
    break;
}
}
}
}

```

## Output:-

=====

SHREE AYYAPPA SEVA SAMITI TRUST

=====

Login

Enter Username : admin

Enter Password : admin

=====

Welcome admin

=====

Enter Choice :

1 -> View Pooja Menu Section.

2 -> View Donation Menu.

3 -> User Management Menu.

1

=====

Pooja Menu

Enter Choice :

1 -> Add By Pooja Name.

2 -> Add By Devotee Name

3 -> Update Pooja Records

4 -> Delete Pooja Records

5 -> Add New Pooja Type

6 -> Update Available Pooja Type

7 -> Delete Available Pooja Type

8 -> View All Available Poojas Type

9 -> View All Devotee Poojas

1

=====

Enter Pooja Name : bhagavati\_seva

Enter Devotee Name : 1

aditya

Enter Devotee Birth Star : aayilyam

Specific date For Pooja ?

Y -> Yes

N -> No

n

Devotee Added.

Add More Pooja?

Y -> Yes.

N -> No.

y

Enter Devotee Name : 2

ananth

Enter Devotee Birth Star : avittam

Specific date For Pooja ?

Y -> Yes

N -> No

n

Devotee Added.

Add More Pooja?

Y -> Yes.

N -> No.

n

=====

Total Bill :

1	bhagavati_seva	100	aditya aayilyam	2024-04-29
2	bhagavati_seva	100	ananth avittam	2024-04-29

Total Amount : 200

=====

=====

Enter Choice :

1 -> View Pooja Menu Section.

2 -> View Donation Menu.

3 -> User Management Menu.

1

=====

Pooja Menu

Enter Choice :

1 -> Add By Pooja Name.

2 -> Add By Devotee Name

3 -> Update Pooja Records

4 -> Delete Pooja Records

5 -> Add New Pooja Type

6 -> Update Available Pooja Type

7 -> Delete Available Pooja Type

8 -> View All Available Poojas Type

9 -> View All Devotee Poojas

3

=====

Details For Update :

Enter Devotee Name : ananth

Enter Entry Date (YYYY-MM-DD) : 2024-04-29

=====

Devotee Details :

10 bhagavati\_seva 100 ananth avittam 2024-04-29

=====

Enter Sr.no Of The Record You Want To Update : 10

Enter Column Number Of The Record (Valid Inputs : 2,4,5,6) : 5

Enter New Value For Birth Star : chothi

=====

Birth Star of 10 Updated

=====

=====

Enter Choice :

1 -> View Pooja Menu Section.

2 -> View Donation Menu.

3 -> User Management Menu.

1

=====

Pooja Menu

Enter Choice :

1 -> Add By Pooja Name.

2 -> Add By Devotee Name

3 -> Update Pooja Records

4 -> Delete Pooja Records

5 -> Add New Pooja Type

6 -> Update Available Pooja Type

7 -> Delete Available Pooja Type

8 -> View All Available Poojas Type

9 -> View All Devotee Poojas

4

=====

Details For Delete :

Enter Devotee Name : ananth

Enter Entry Date (YYYY-MM-DD) : 2024-04-29

=====

Devotee Details :

10   bhagavati\_seva 100   ananth chothi 2024-04-29

=====

Enter Sr.no Of The Record You Want To Delete : 10

=====

10 Entry Deleted !!!!

=====

=====

Enter Choice :

1 -> View Pooja Menu Section.

2 -> View Donation Menu.

3 -> User Management Menu.

1

=====

Pooja Menu

Enter Choice :

1 -> Add By Pooja Name.

2 -> Add By Devotee Name

3 -> Update Pooja Records

4 -> Delete Pooja Records

5 -> Add New Pooja Type

6 -> Update Available Pooja Type

7 -> Delete Available Pooja Type

8 -> View All Available Poojas Type

9 -> View All Devotee Poojas

5

=====

New Pooja :

Enter Name For New Pooja : noorum\_palum

Enter Rate Of New Pooja : 70

=====

New Pooja Name Inserted !!!!

=====

=====

Enter Choice :

1 -> View Pooja Menu Section.

2 -> View Donation Menu.

3 -> User Management Menu.

1

=====

Pooja Menu

Enter Choice :

1 -> Add By Pooja Name.

2 -> Add By Devotee Name

3 -> Update Pooja Records

4 -> Delete Pooja Records

5 -> Add New Pooja Type

6 -> Update Available Pooja Type

7 -> Delete Available Pooja Type

8 -> View All Available Poojas Type

9 -> View All Devotee Poojas

6

=====

Update Pooja :

Enter Old Pooja Name : noorum\_palum

Available Poojas

noorum\_palum 70

Enter Column Number You Want To Update ( 1 or 2 ) :

2

Enter New Price : 100

=====

Pooja Name Updated !!!!

=====

=====

Enter Choice :

1 -> View Pooja Menu Section.

2 -> View Donation Menu.

3 -> User Management Menu.

1

=====

Pooja Menu

Enter Choice :

1 -> Add By Pooja Name.

2 -> Add By Devotee Name

3 -> Update Pooja Records

4 -> Delete Pooja Records

5 -> Add New Pooja Type

6 -> Update Available Pooja Type

7 -> Delete Available Pooja Type

8 -> View All Available Poojas Type

9 -> View All Devotee Poojas

7

=====

Delete Pooja :

Enter Pooja Name : noorum\_palum

=====

noorum\_palum Pooja Deleted !!!!

=====

=====

Enter Choice :

1 -> View Pooja Menu Section.

2 -> View Donation Menu.

3 -> User Management Menu.

1

=====

Pooja Menu

Enter Choice :

1 -> Add By Pooja Name.

2 -> Add By Devotee Name

3 -> Update Pooja Records

4 -> Delete Pooja Records

5 -> Add New Pooja Type

6 -> Update Available Pooja Type

7 -> Delete Available Pooja Type

8 -> View All Available Poojas Type

9 -> View All Devotee Poojas

8

=====

Available Poojas

bhagavati\_seva 100

pushpanjali 10

=====

=====

Enter Choice :

1 -> View Pooja Menu Section.

2 -> View Donation Menu.

3 -> User Management Menu.

1

=====

Pooja Menu

Enter Choice :

1 -> Add By Pooja Name.

2 -> Add By Devotee Name

3 -> Update Pooja Records

4 -> Delete Pooja Records

5 -> Add New Pooja Type

6 -> Update Available Pooja Type

7 -> Delete Available Pooja Type

8 -> View All Available Poojas Type

9 -> View All Devotee Poojas

9

=====



All Pooja Records :

1	bhagavati_seva	100	aditya aayilyam	2024-04-29	2024-04-29
2	pushpanjali	10	aditya aayilyam	2024-04-29	2024-04-29
3	pushpanjali	10	aditya aayilyam	2024-04-29	2024-04-29
4	pushpanjali	10	aditya aayilyam	2024-04-29	2024-04-29
6	pushpanjali	10	aditya aayilyam	2024-04-29	2024-04-29
7	pushpanjali	10	aditya aayilyam	2024-04-29	2024-04-29
8	pushpanjali	10	aditya aayilyam	2024-04-29	2024-04-29
9	bhagavati_seva	100	aditya aayilyam	2024-04-29	2024-04-29

=====  
=====

Enter Choice :

1 -> View Pooja Menu Section.  
2 -> View Donation Menu.  
3 -> User Management Menu.  
2

=====  
Donation Management Menu

Enter Choice :

1 -> Add Donation  
2 -> Update Donation  
3 -> Delete Donation  
4 -> Show All Donations  
1

=====

Enter Donar Name : ayush  
Enter Donation Amount : 500

=====

Donation Added !!!!

=====

=====

Enter Choice :

1 -> View Pooja Menu Section.  
2 -> View Donation Menu.  
3 -> User Management Menu.  
2

=====  
Donation Management Menu

Enter Choice :

1 -> Add Donation  
2 -> Update Donation  
3 -> Delete Donation  
4 -> Show All Donations  
2

=====

Details For Update :

Enter Donar Name : ayush  
Enter Entry Date (YYYY-MM-DD) : 2024-04-29

=====

Donar Details :

2     ananth   500

=====

Enter Sr.no Of The Record You Want To Update : 2

Enter Column Number Of The Record (Valid Inputs : 2,3) : 3

Enter New Donation Amount :

1500

=====

Donation Record Updated !!!!

=====

=====

Enter Choice :

1 -> View Pooja Menu Section.

2 -> View Donation Menu.

3 -> User Management Menu.

2

=====

Donation Management Menu

Enter Choice :

1 -> Add Donation

2 -> Update Donation

3 -> Delete Donation

4 -> Show All Donations

3

=====

Details For Delete :

Enter Donar Name : aditya

Enter Entry Date (YYYY-MM-DD) : 2024-04-29

=====

Donar Details :

3     aditya   1000

=====

Enter Sr.no. Of The Record You Want To Delete : 3

=====

3 Entry Deleted !!!!

=====

=====

Enter Choice :

1 -> View Pooja Menu Section.

2 -> View Donation Menu.

3 -> User Management Menu.

3

=====

User Management Menu

Enter Choice :  
1 -> Update Username  
2 -> Update Password  
1

=====

Enter Current Username : admin  
Enter Current Password : admin

Enter New Username : admin1

=====

Username Updated !!!!

=====

=====

Enter Choice :  
1 -> View Pooja Menu Section.  
2 -> View Donation Menu.  
3 -> User Management Menu.  
3

=====

User Management Menu

Enter Chosie :  
1 -> Update Username  
2 -> Update Password  
2

=====

Enter Current Username : admin  
Enter Current Password : admin

Enter New Password : admin1

=====

Password Updated !!!!

=====

=====

Enter Choice :  
1 -> View Pooja Menu Section.  
2 -> View Donation Menu.  
3 -> User Management Menu.  
4

=====

Exiting !!!!  
Thank You For Using Temple Management !!!!

=====