DATABASE MANAGEMENT SYSTEM LAB

MINI PROJECT

***FITNESS CENTRE MANAGEMENT DATABASE***

Using MYSQL

NAME: Preksha Jain

SAP ID : 500120166

ROLL : R2142230012

FACULTY: Mr. Syed Sajid Hussain

BATCH: 1

**INTRODUCTION:**

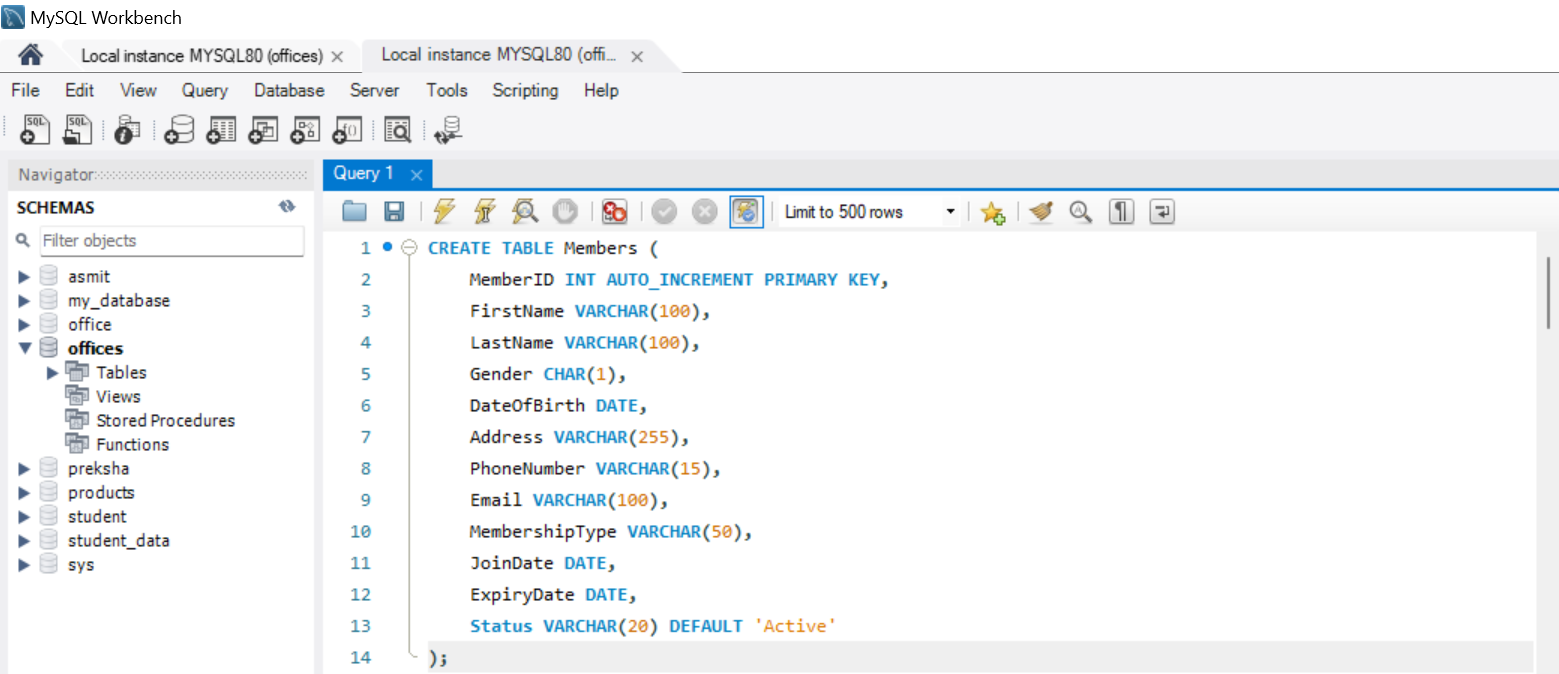
A Fitness Center Management Database is typically designed to handle the administrative and operational tasks of a fitness facility. This kind of database can store a wide variety of information, such as member details, trainer schedules, membership types, payments, class bookings, equipment inventory, and more.

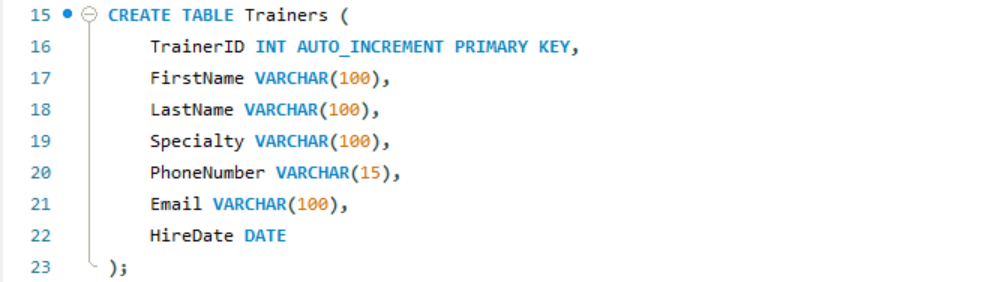
**Key Components of a Fitness Center Database:**

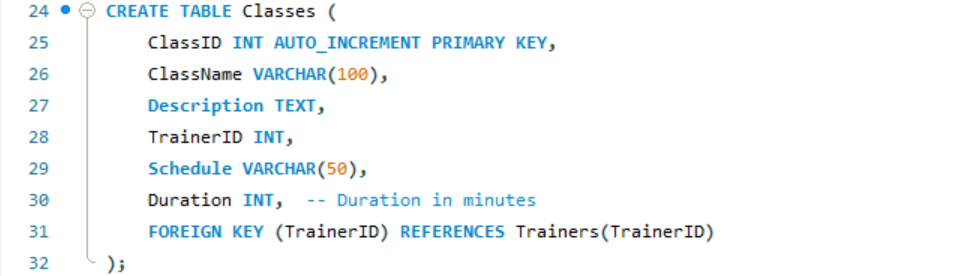
1. **Members Table**: This table stores information about the members who attend the fitness center.
   * **MemberID**: Unique identifier for each member.
   * **FirstName**: Member's first name.
   * **LastName**: Member's last name.
   * **DOB**: Date of birth.
   * **Gender**: Member’s gender.
   * **MembershipType**: Type of membership (e.g., monthly, yearly, family).
   * **JoinDate**: Date the member joined the fitness center.
   * **ExpirationDate**: Date the membership expires.
   * **PhoneNumber**: Contact number.
   * **Email**: Contact email address.
2. **Classes Table**: This table stores details of the fitness classes offered by the gym.
   * **ClassID**: Unique identifier for each class.
   * **ClassName**: Name of the class (e.g., Yoga, Pilates, Zumba).
   * **InstructorID**: Foreign key linking to the Instructor table.
   * **StartTime**: Class start time.
   * **EndTime**: Class end time.
   * **Capacity**: Maximum number of participants allowed in the class.
3. **Instructors Table**: This table holds information about the fitness instructors.
   * **InstructorID**: Unique identifier for each instructor.
   * **FirstName**: Instructor’s first name.
   * **LastName**: Instructor’s last name.
   * **Specialization**: The type of fitness classes the instructor specializes in (e.g., strength training, yoga).
   * **HireDate**: The date the instructor was hired.
   * **Email**: Instructor’s contact email.
4. **Memberships Table**: This table records membership details and payment history.
   * **MembershipID**: Unique identifier for the membership.
   * **MemberID**: Foreign key linking to the Members table.
   * **StartDate**: Start date of the membership.
   * **EndDate**: End date of the membership.
   * **PaymentStatus**: Status of the membership payment (e.g., Paid, Pending).
   * **Fee**: The amount paid for the membership.
5. **Attendance Table**: Tracks member attendance in classes and gym visits.
   * **AttendanceID**: Unique identifier for the attendance record.
   * **MemberID**: Foreign key linking to the Members table.
   * **ClassID**: Foreign key linking to the Classes table (if attending a class).
   * **VisitDate**: The date of the visit or class attended.
   * **Status**: Attendance status (e.g., Attended, Missed).
6. **Payments Table**: This table stores payment transactions for memberships or services.
   * **PaymentID**: Unique identifier for each payment.
   * **MemberID**: Foreign key linking to the Members table.
   * **Amount**: The amount paid.
   * **PaymentDate**: Date when the payment was made.
   * **PaymentMethod**: Method of payment (e.g., Credit Card, Cash, Bank Transfer).
7. **Equipment Table**: This table tracks the equipment used in the gym.
   * **EquipmentID**: Unique identifier for each equipment item.
   * **EquipmentName**: Name of the equipment (e.g., Treadmill, Dumbbell).
   * **Quantity**: Number of available equipment units.
   * **Condition**: Condition of the equipment (e.g., New, Used, Damaged).
8. **Trainer Assignments Table**: This table records which trainers are assigned to which classes.
   * **AssignmentID**: Unique identifier for the assignment.
   * **TrainerID**: Foreign key linking to the Instructors table.
   * **ClassID**: Foreign key linking to the Classes table.
   * **AssignedDate**: Date when the trainer was assigned to the class.

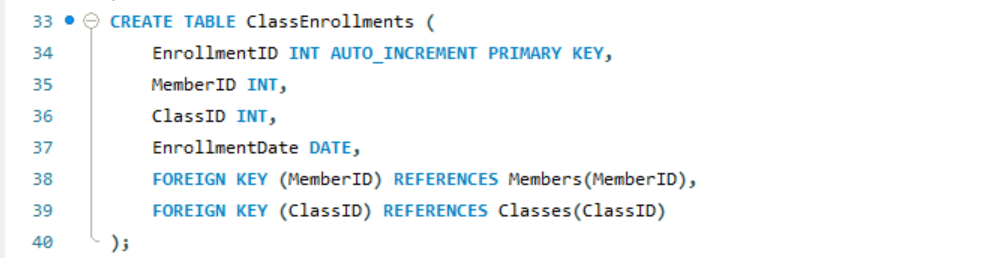
CODE

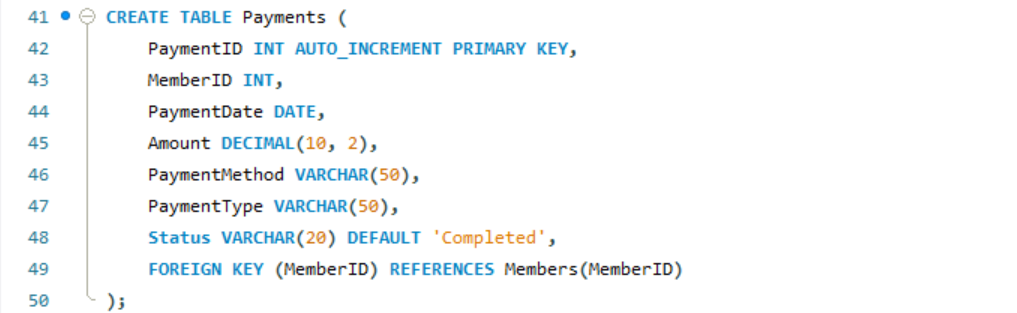
Here, I used a database and created tables that were required in the database.

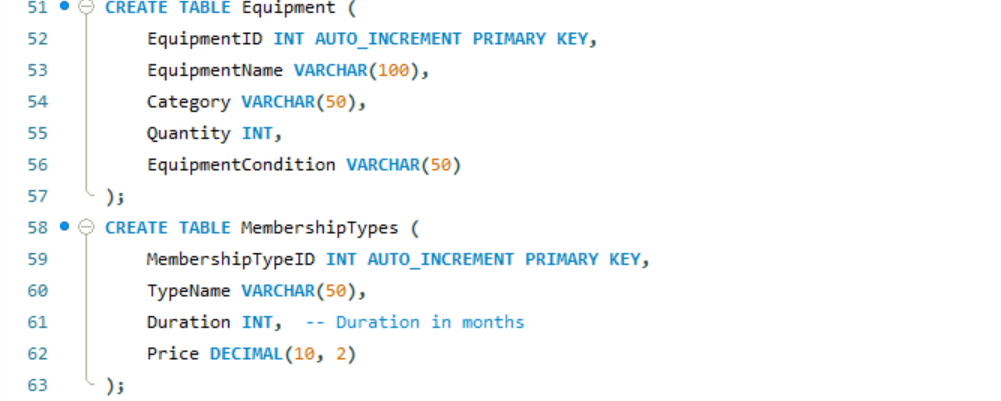




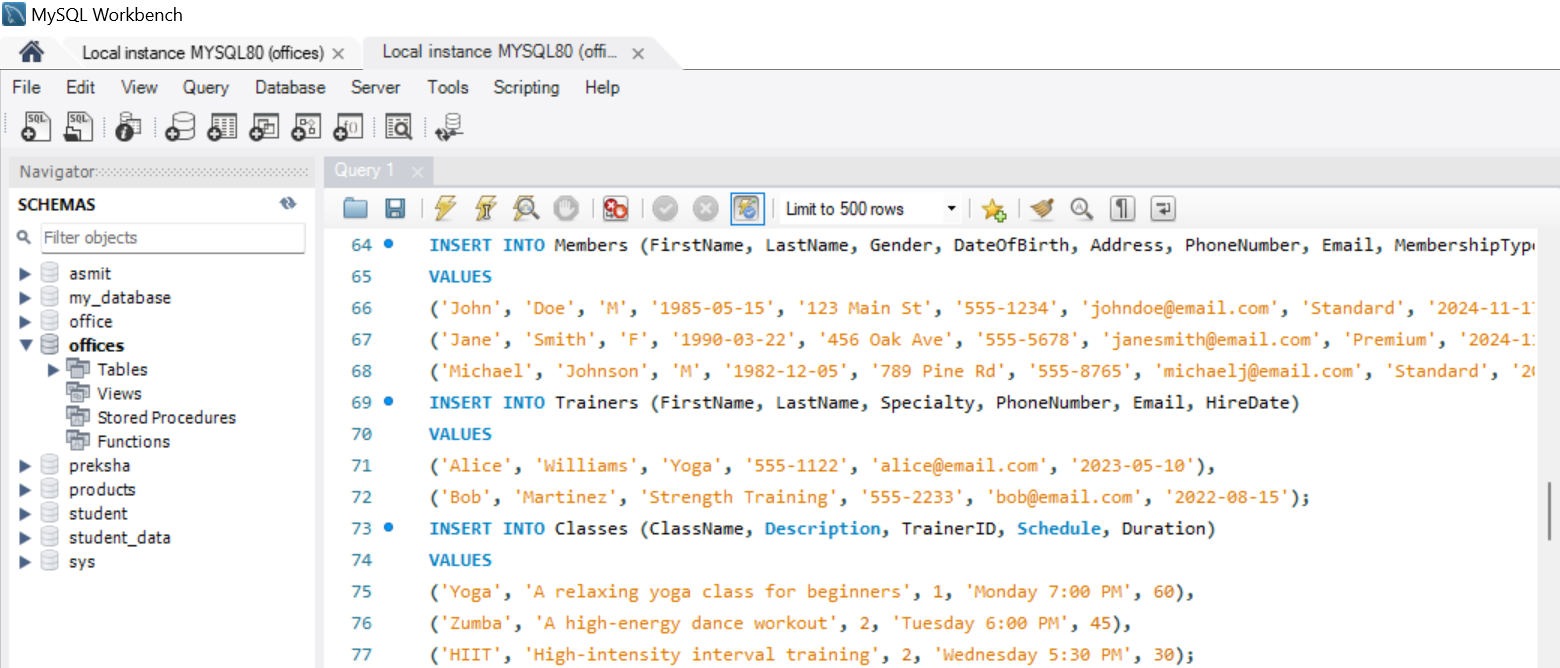


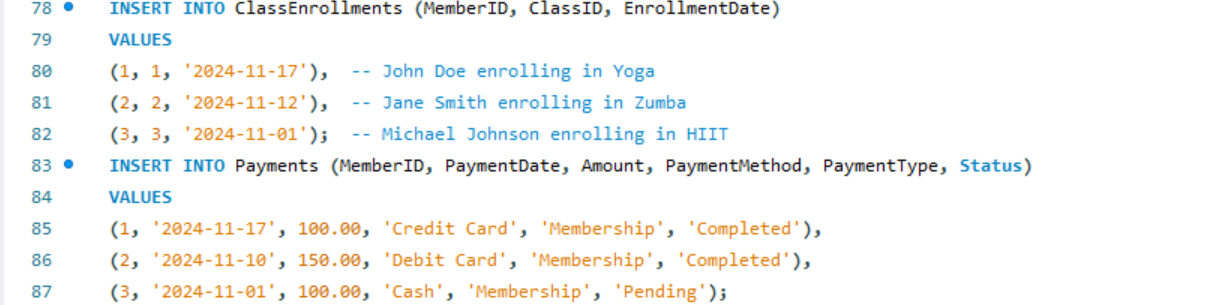


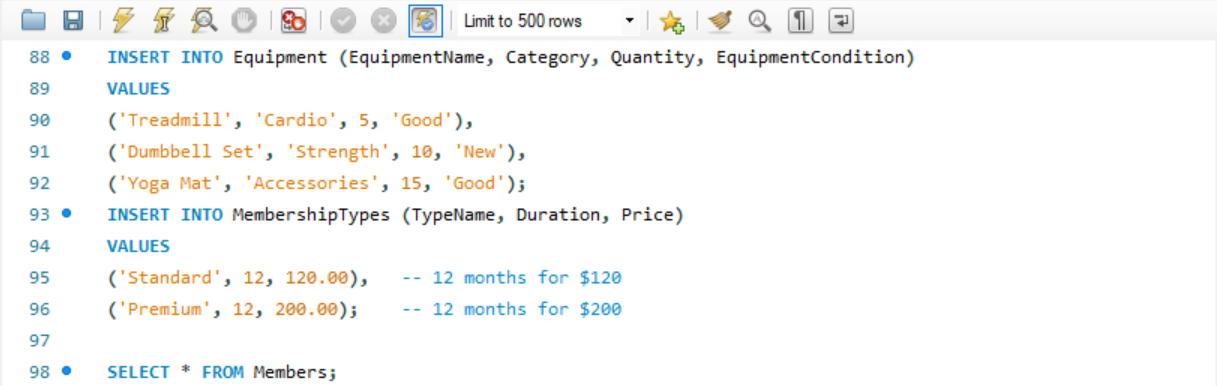




Now, Inserting the values into the tables created.

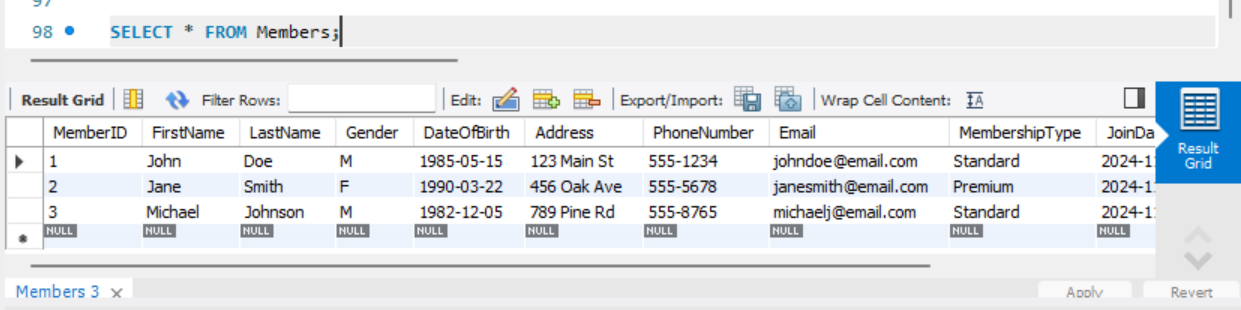




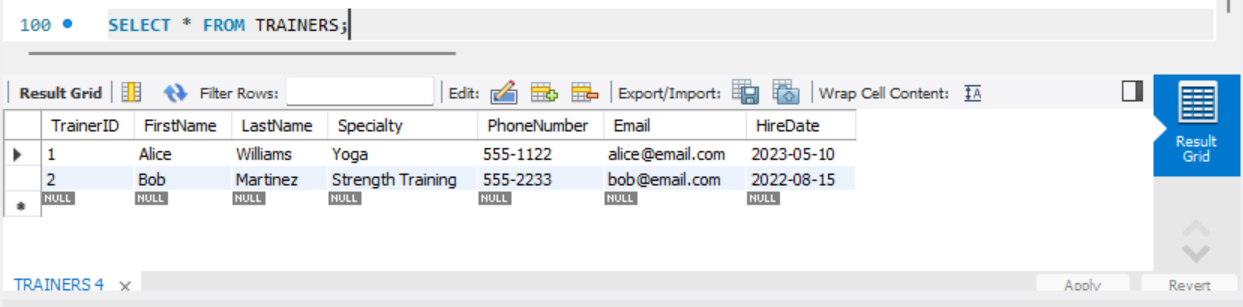


Running Commands,

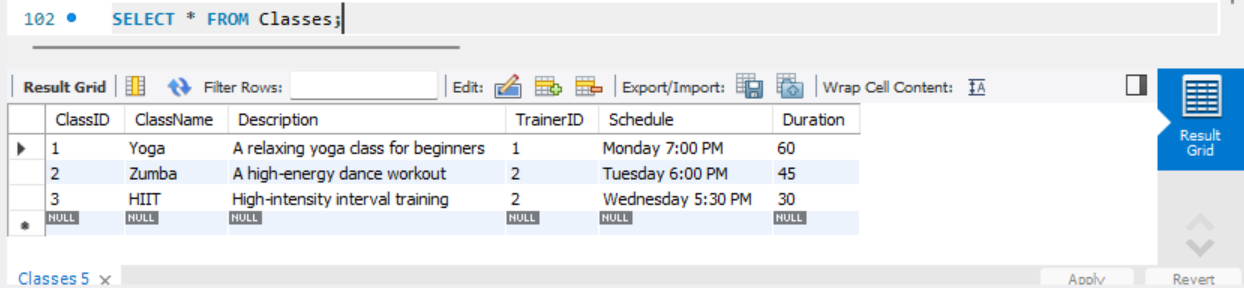
1. Searching the members.



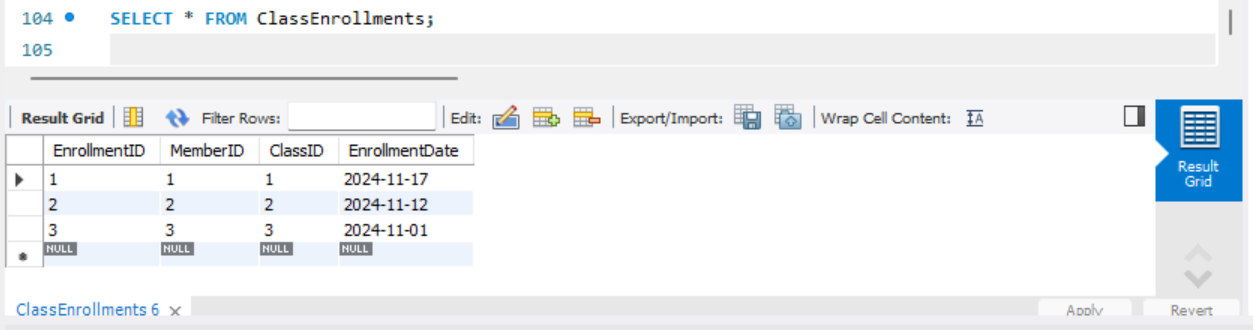
1. Searching the trainers.



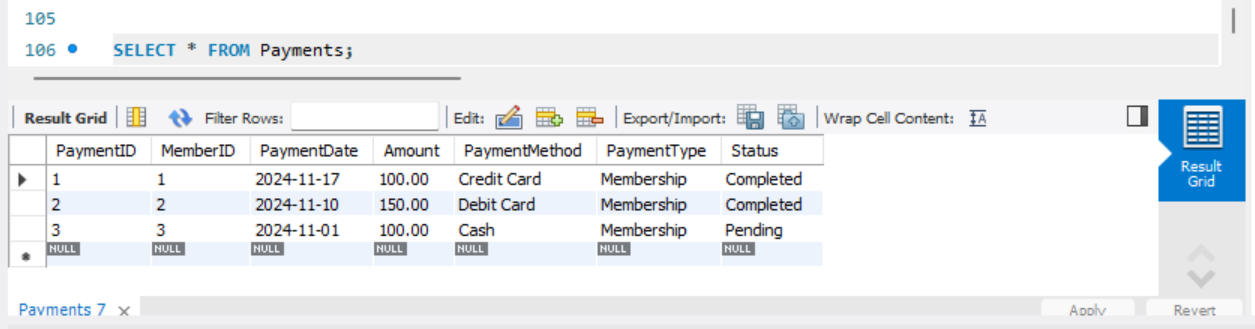
1. Searching the classes.



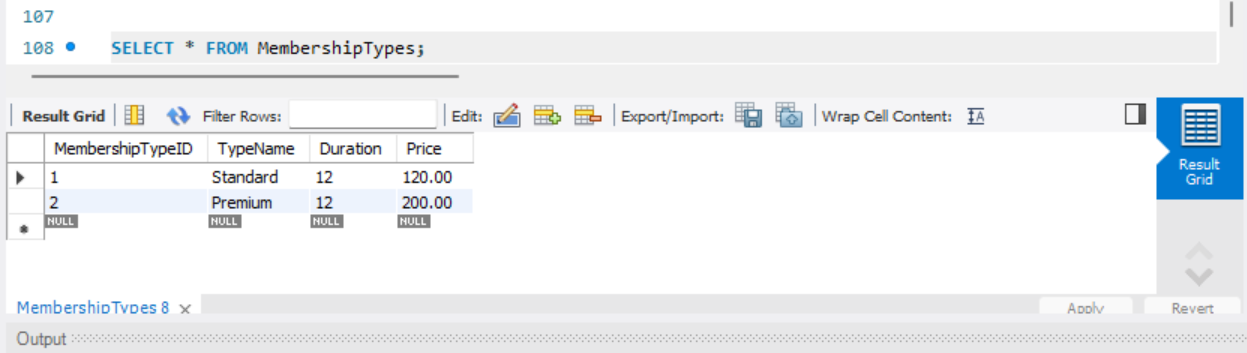
1. To view all class enrolments.



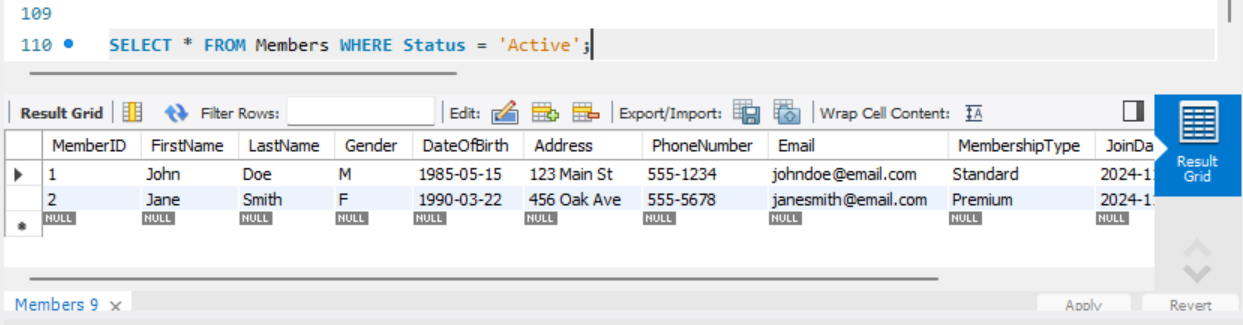
1. To view all payments .



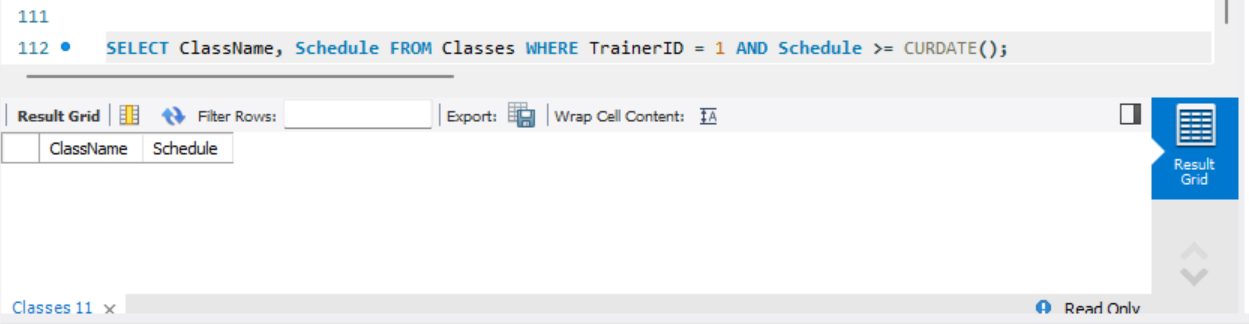
1. To view all membership types.



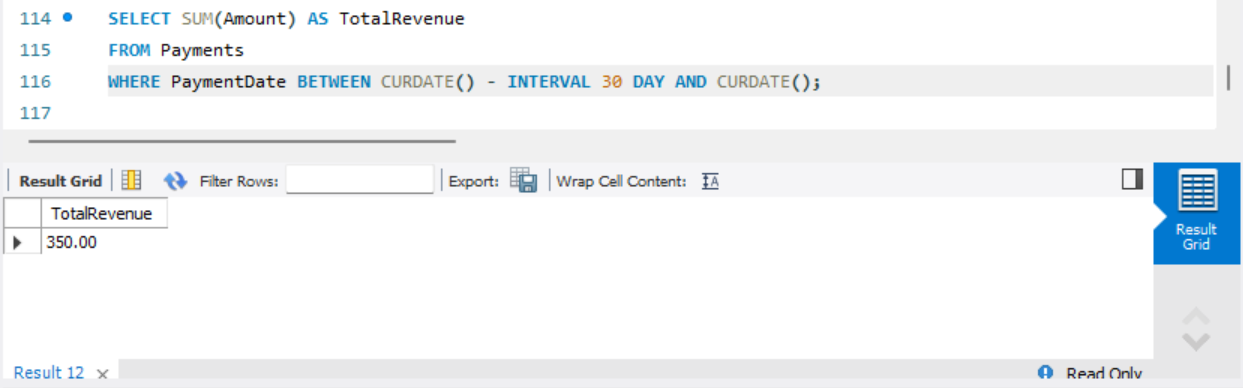
1. Getting all the Active members.



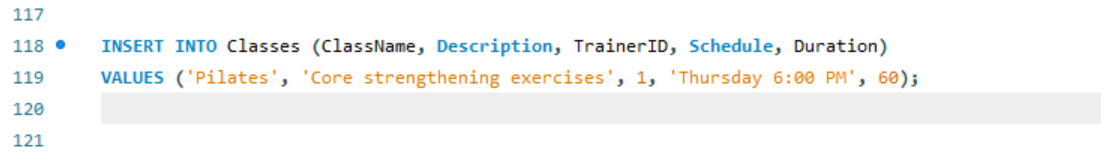
1. All upcoming classes from a specific trainer



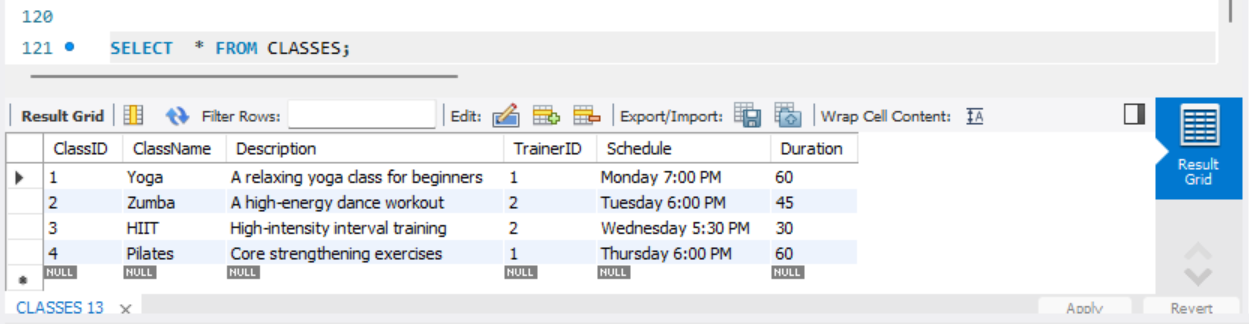
1. Total revenue for last 30 days



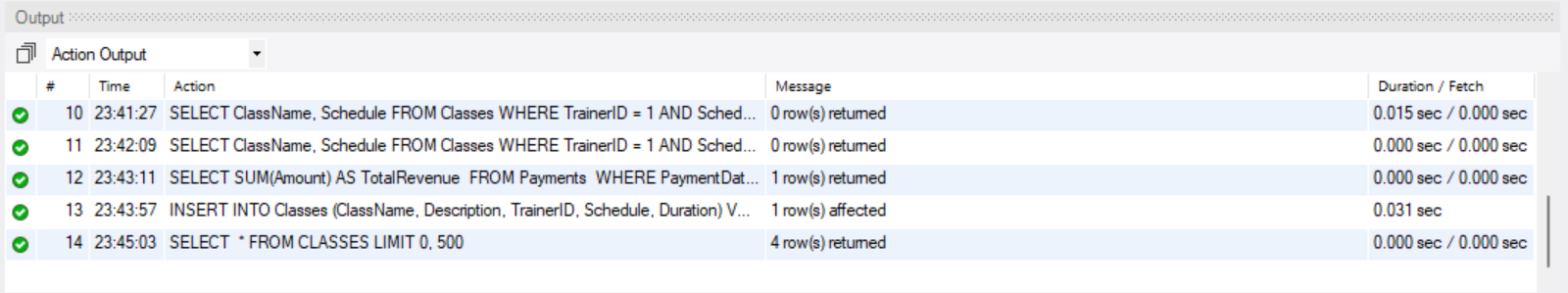
1. Inserting New values

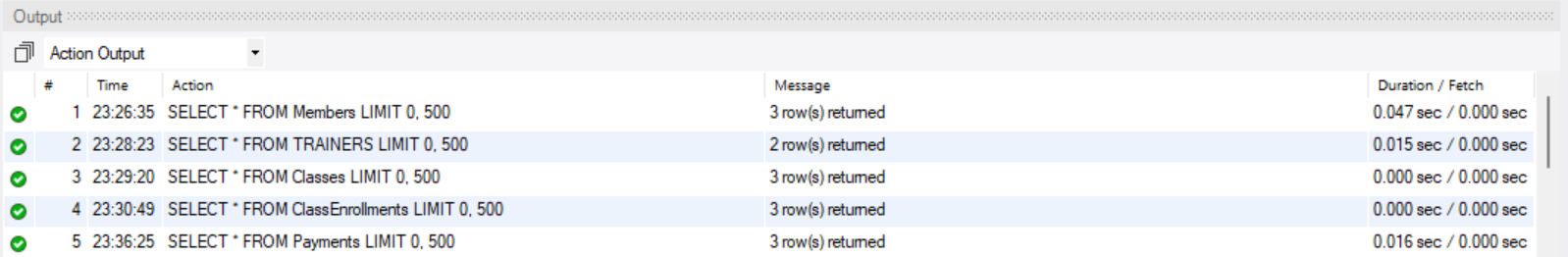
****

**Output**

****

**ACTION OUTPUT:**

****

****

**This shows that our database is working as it is performing all the queries.**