**School management database design  
all sql statements used in the project attached in this file   
  
**

**1. ER diagram analysis:**

**a. identify all the primary keys and foreign in the schema**

**The primary keys:**

**Entities and Primary Keys**:

1. **classroom → classroom\_id**
2. **grade → grade\_id**
3. **course → course\_id**
4. **student → student\_id**
5. **parent → parent\_id**
6. **teacher → teacher\_id**
7. **classroom\_student → classroom\_id, student\_id**
8. **attendance → date, student\_id**
9. **exam\_type → exam\_type\_id**
10. **exam → exam\_id**
11. **exam\_result → exam\_id, student\_id, course\_id**

**The foreign keys:  
Table, Foreign Key, References Table, and References Column**:

1. classroom
   * Foreign Key: grade\_id → References Table: grade → References Column: grade\_id
   * Foreign Key: teacher\_id → References Table: teacher → References Column: teacher\_id
2. course
   * Foreign Key: grade\_id → References Table: grade → References Column: grade\_id
3. student
   * Foreign Key: parent\_id → References Table: parent → References Column: parent\_id
4. classroom\_student
   * Foreign Key: classroom\_id → References Table: classroom → References Column: classroom\_id
   * Foreign Key: student\_id → References Table: student → References Column: student\_id
5. attendance
   * Foreign Key: student\_id → References Table: student → References Column: student\_id
6. exam
   * Foreign Key: exam\_type\_id → References Table: exam\_type → References Column: exam\_type\_id
7. exam\_result
   * Foreign Key: exam\_id → References Table: exam → References Column: exam\_id
   * Foreign Key: student\_id → References Table: student → References Column: student\_id
   * Foreign Key: course\_id → References Table: course → References Column: course\_id

**2. Review the provided ER diagram:**

**Table overview**:

* **classroom**:  
  Stores details about classrooms, including grade, section, teacher, and additional remarks.
* **grade**:  
  Contains grade-specific information, such as grade name and a brief description.
* **course**:  
  Holds course details, including their associated grades.
* **student**:  
  Manages student data, including personal details, relationships with parents, and enrollment status.
* **parent**:  
  Stores parent or guardian information, including contact details.
* **teacher**:  
  Contains teacher records, including their personal information and contact details.
* **classroom\_student**:  
  Manages many-to-many relationships between students and classrooms.
* **attendance**:  
  Tracks student attendance with dates, statuses ,and remarks.
* **exam\_type**:  
  Defines various exam types, such as quizzes, midterms, and finals.
* **exam**:  
  Stores exam details, including type, date, and start time.
* **exam\_result**:  
  Records students' performance and scores in specific courses.

**3. List the relationships (one-to-one, one-to-many, many-to-many) between entities and explain.**

### Relationships Between Entities

1. **Classroom → Grade**
   * **Relationship Type:** One-to-Many
   * **Foreign Key in Table 1:** grade\_id
   * **References Column in Table 2:** grade\_id
   * **Description:** Each classroom belongs to one grade, but one grade can have many classrooms.
2. **Classroom → Teacher**
   * **Relationship Type:** One-to-One
   * **Foreign Key in Table 1:** teacher\_id
   * **References Column in Table 2:** teacher\_id
   * **Description:** Each classroom is assigned to one teacher, and each teacher manages one class.
3. **Course → Grade**
   * **Relationship Type:** One-to-Many
   * **Foreign Key in Table 1:** grade\_id
   * **References Column in Table 2:** grade\_id
   * **Description:** Each course is assigned to one grade, but one grade can have many courses.
4. **Student → Parent**
   * **Relationship Type:** One-to-Many
   * **Foreign Key in Table 1:** parent\_id
   * **References Column in Table 2:** parent\_id
   * **Description:** Each student has one parent, but one parent can have multiple students.

**SQL Table Relationships**

1. **Classroom\_Student → Classroom**
   * **Relationship Type:** Many-to-One
   * **Foreign Key in Table 1:** classroom\_id
   * **References Column in Table 2:** classroom\_id
   * **Description:** Links multiple students to a single classroom.
2. **Classroom\_Student → Student**
   * **Relationship Type:** Many-to-One
   * **Foreign Key in Table 1:** student\_id
   * **References Column in Table 2:** student\_id
   * **Description:** Links multiple classrooms to a single student.
3. **Attendance → Student**
   * **Relationship Type:** Many-to-One
   * **Foreign Key in Table 1:** student\_id
   * **References Column in Table 2:** student\_id
   * **Description:** Tracks attendance for each student in the system.
4. **Exam → Exam\_Type**
   * **Relationship Type:** One-to-Many
   * **Foreign Key in Table 1:** exam\_type\_id
   * **References Column in Table 2:** exam\_type\_id
   * **Description:** Each exam is categorized under a specific exam type.
5. **Exam\_Result → Exam**
   * **Relationship Type:** Many-to-One
   * **Foreign Key in Table 1:** exam\_id
   * **References Column in Table 2:** exam\_id
   * **Description:** Tracks results for multiple students in one exam.
6. **Exam\_Result → Student**
   * **Relationship Type:** Many-to-One
   * **Foreign Key in Table 1:** student\_id
   * **References Column in Table 2:** student\_id
   * **Description:** Tracks results for each student in various exams.
7. **Exam\_Result → Course**
   * **Relationship Type:** Many-to-One
   * **Foreign Key in Table 1:** course\_id
   * **References Column in Table 2:** course\_id
   * **Description:** Tracks exam results for specific courses.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **table 1** | **table 2** | **relationship Type** | |  | | --- | | **Foreign Key in Table 1** |  |  | | --- | |  | | |  | | --- | | **References Column in Table 2** |  |  | | --- | |  | | | **Description** | | --- |  |  | | --- | |  | |
| classroom | grade | |  | | --- | | One-to-Many |  |  | | --- | |  | | grade\_id | grade\_id | |  | | --- | | Each classroom belongs to one grade, but one grade can have many classrooms. |  |  | | --- | |  | |
| classroom | teacher | One-to-One | teacher\_id | teacher\_id | |  | | --- | | Each classroom is assigned to one teacher, and each teacher manages one class. |  |  | | --- | |  | |
| course | grade | |  | | --- | | One-to-Many |  |  | | --- | |  | | grade\_id | grade\_id | |  | | --- | | Each course is assigned to one grade, but one grade can have many courses. |  |  | | --- | |  |  |  | | --- | |  | |
| student | |  | | --- | | parent |  |  | | --- | |  | | |  | | --- | | One-to-Many |  |  | | --- | |  | | parent\_id | parent\_id | |  | | --- | | Each student has one parent, but one parent can have multiple students. |  |  | | --- | |  | |
| classroom\_student | classroom | |  | | --- | | Many-to-One |  |  | | --- | |  | | classroom\_id | classroom\_id | |  | | --- | | Links multiple students to a single classroom. |  |  | | --- | |  | |
| classroom\_student | student | |  | | --- | | Many-to-One |  |  | | --- | |  | | student\_id | student\_id | |  | | --- | | Links multiple classrooms to a single student. |  |  | | --- | |  | |
| attendance | student | |  | | --- | | Many-to-One |  |  | | --- | |  | | student\_id | student\_id | |  | | --- | | Tracks attendance for each student in the system. |  |  | | --- | |  | |
| exam | exam\_type | |  | | --- | | One-to-Many |  |  | | --- | |  | | exam\_type\_id | exam\_type\_id | |  | | --- | | Each exam is categorized under a specific exam type. |  |  | | --- | |  | |
| exam\_result | exam | |  | | --- | | Many-to-One |  |  | | --- | |  | | exam\_id | exam\_id | |  | | --- | | Tracks results for multiple students in one exam. |  |  | | --- | |  | |
| exam\_result | student | |  | | --- | | Many-to-One |  |  | | --- | |  | | student\_id | student\_id | |  | | --- | | Tracks results for each student in various exams. |  |  | | --- | |  | |
| exam\_result | course | |  | | --- | | Many-to-One |  |  | | --- | |  | | course\_id | course\_id | |  | | --- | |  |  |  | | --- | |  |   Tracks exam results for specific courses. |

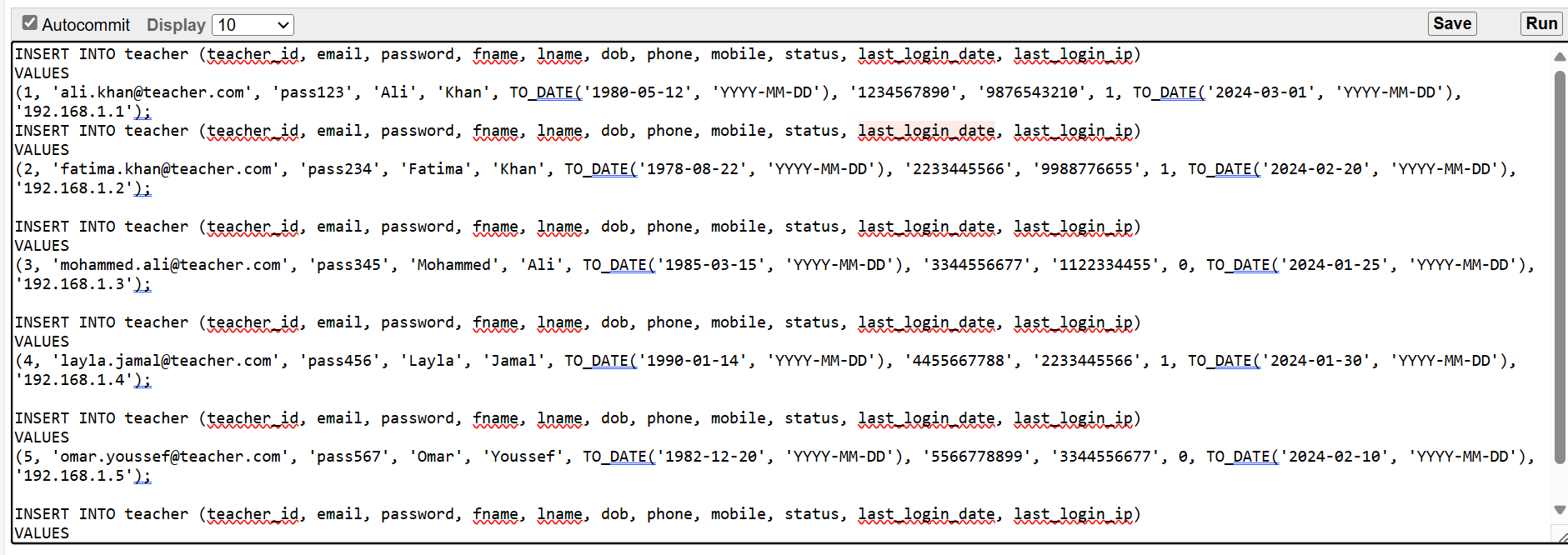
**2. Sql table relation :**

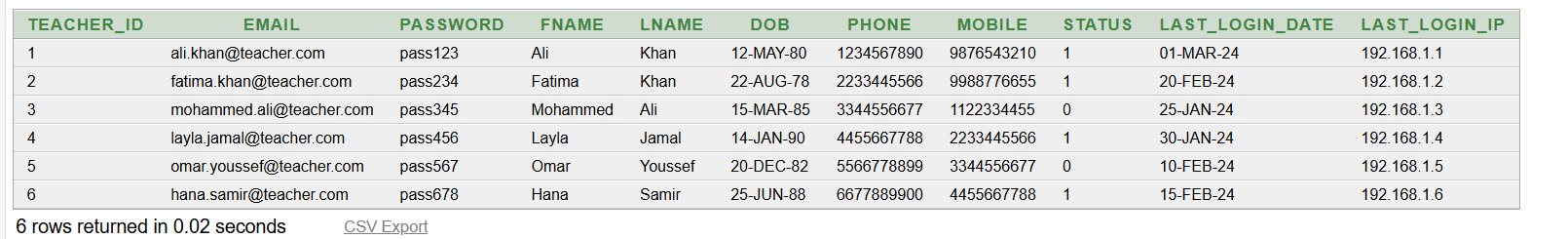
**This file contains the tables, relationships between the tables, constraints, and explanation for each attribute in each table**

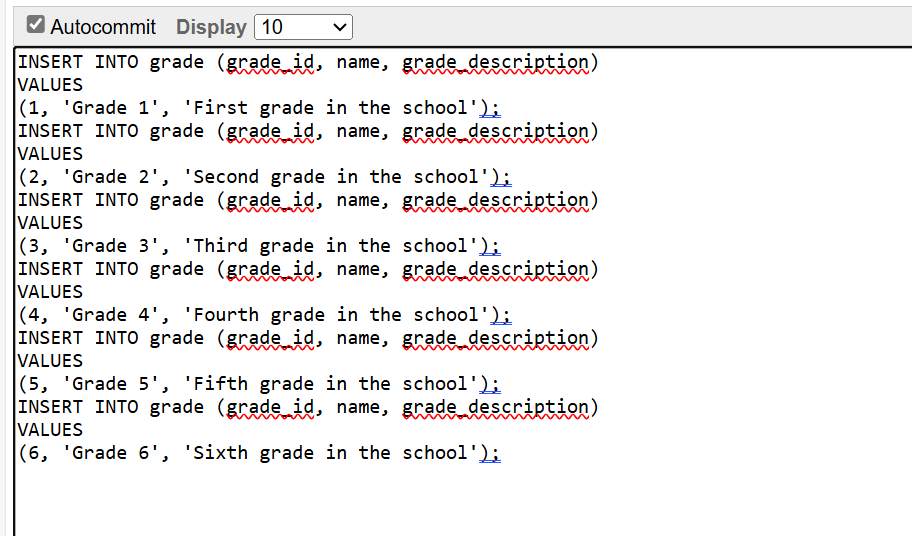
**  
3) Insert sample data for the following tables:**

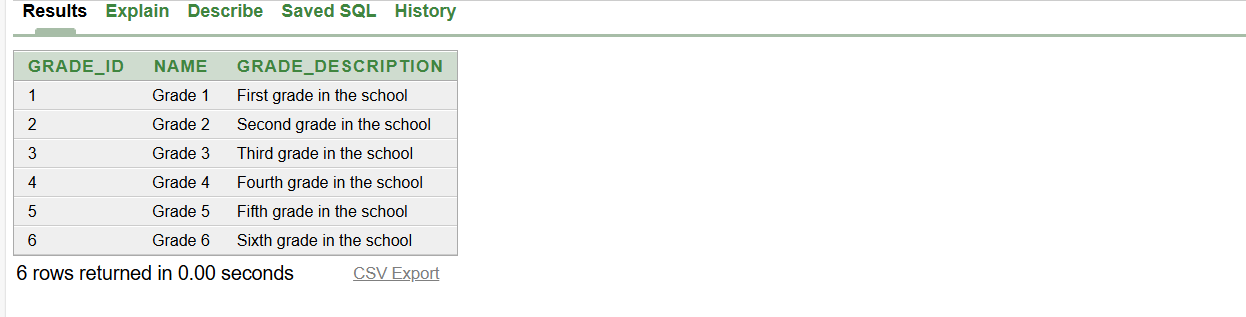
**In the attached file:**

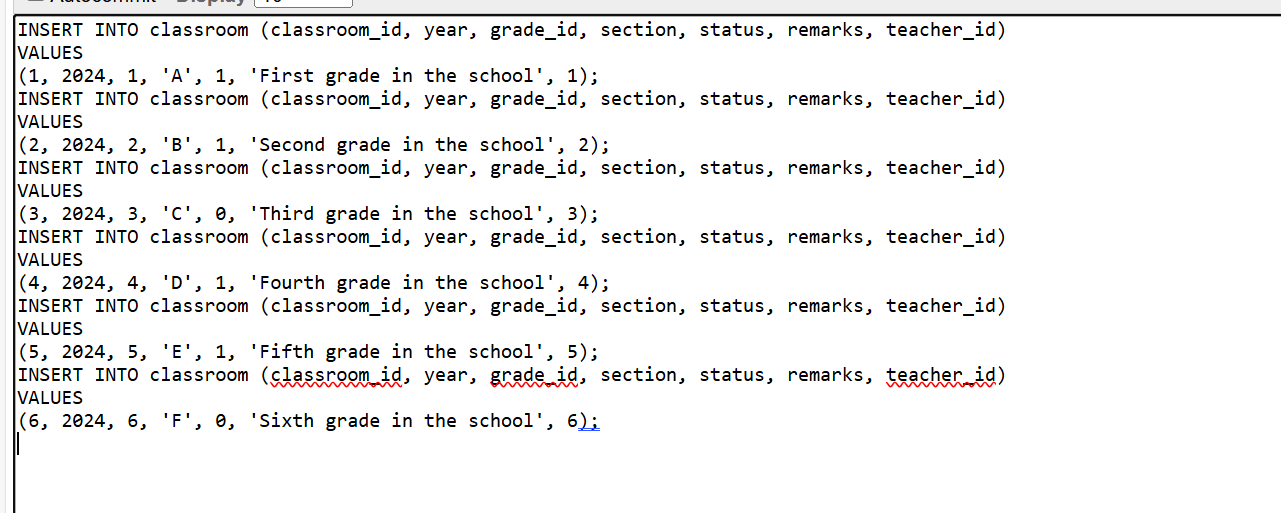


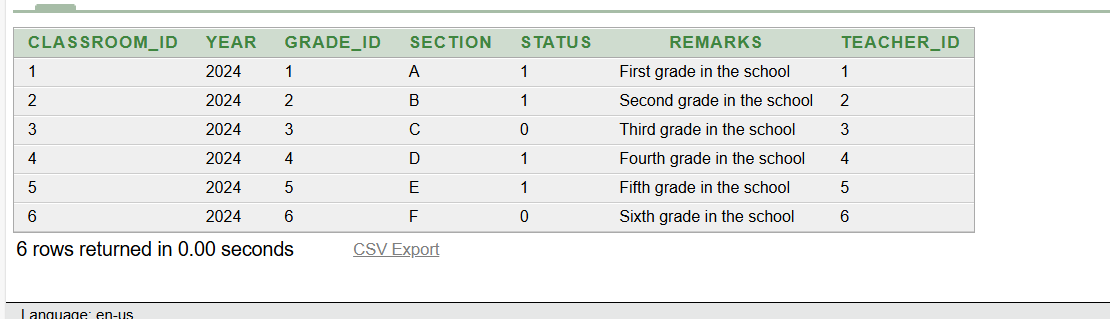


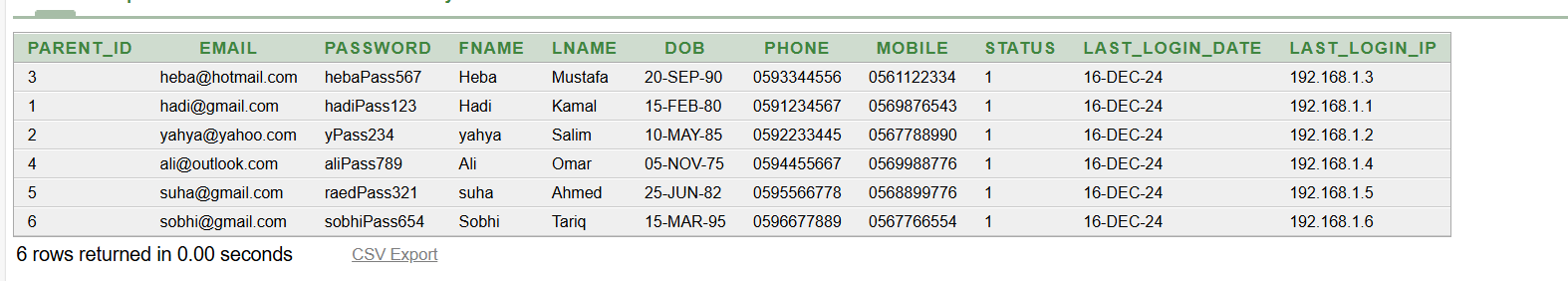
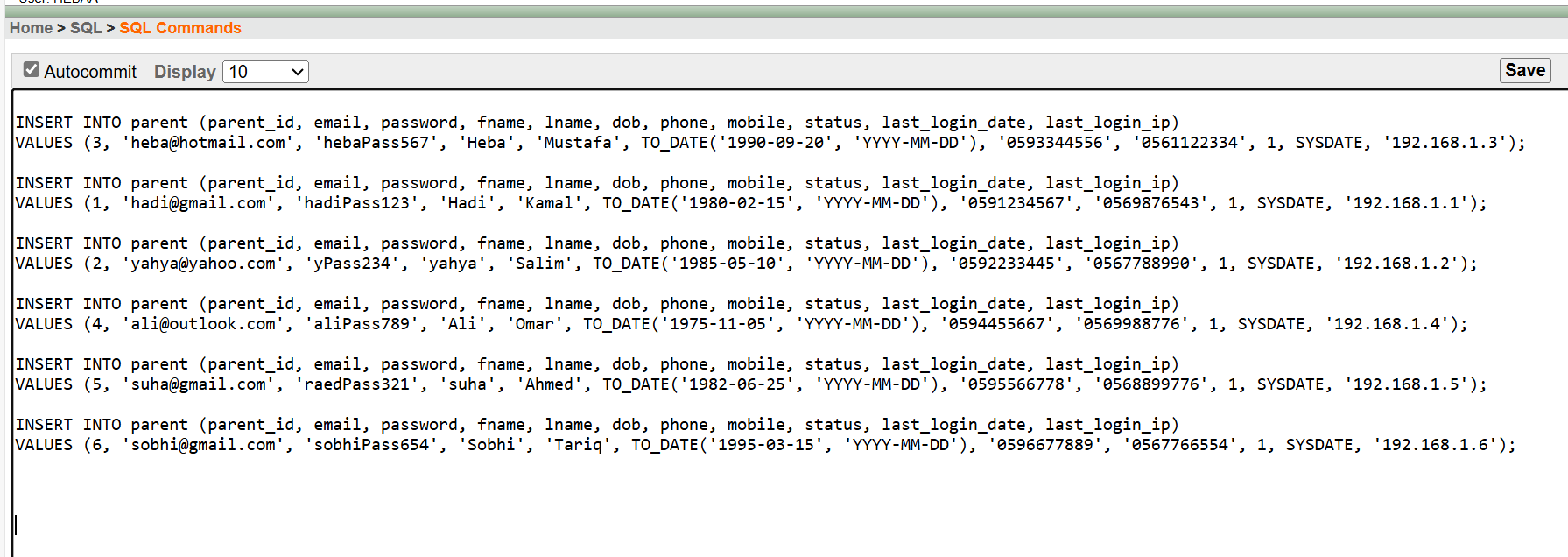




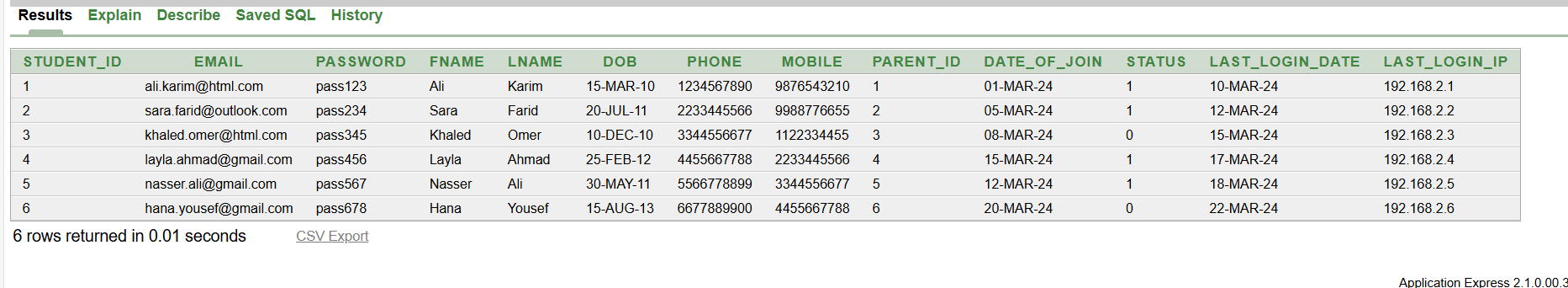


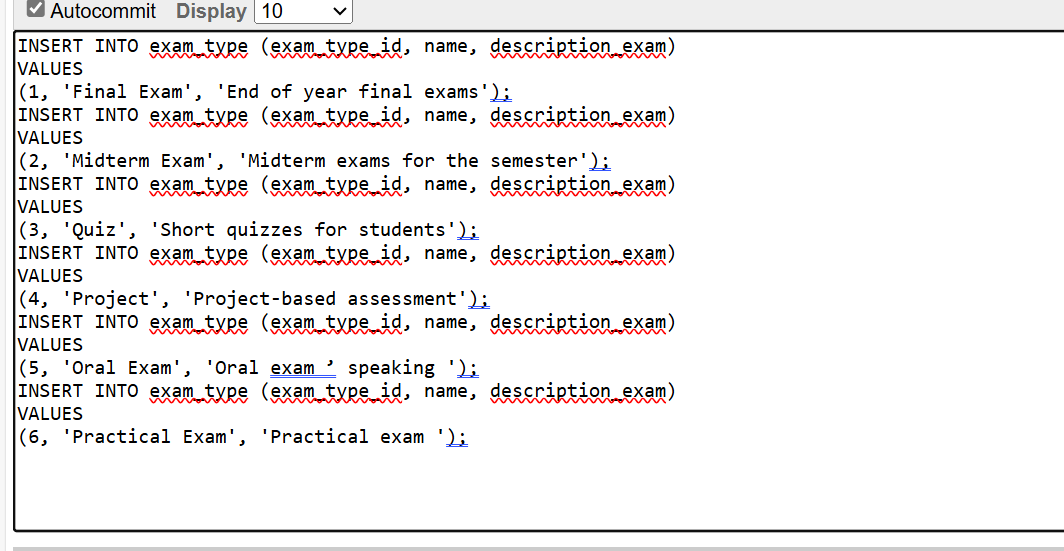


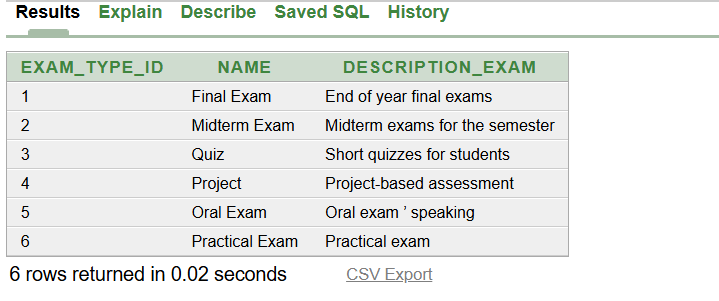


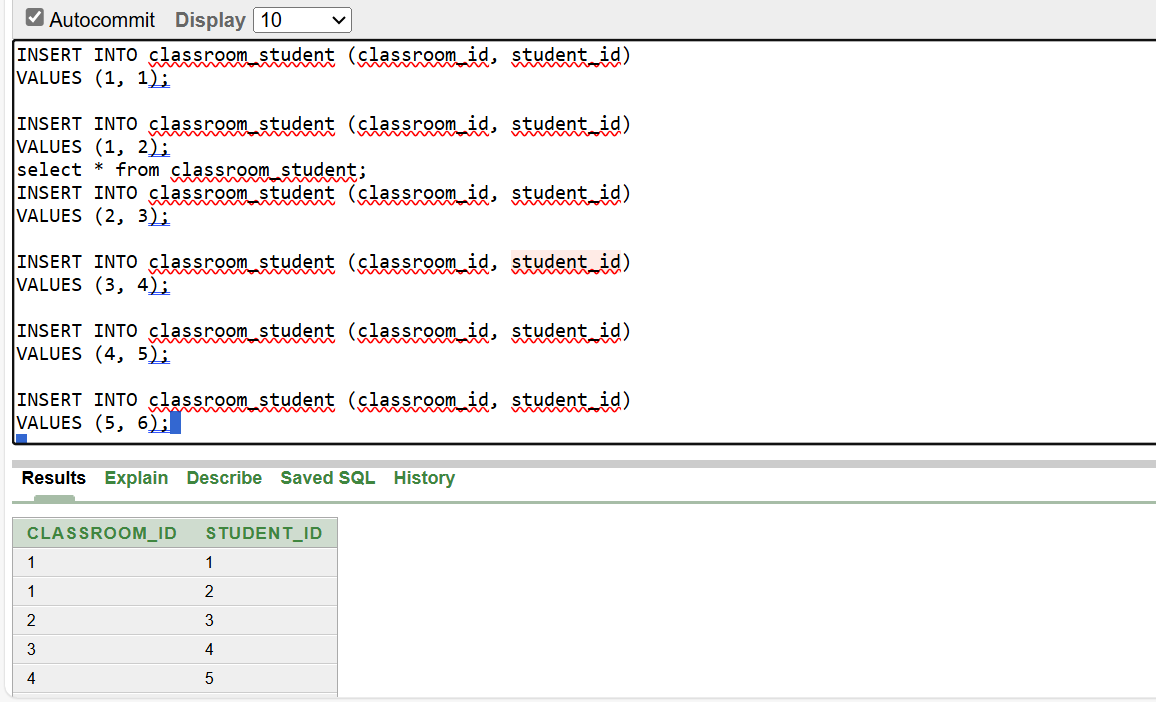


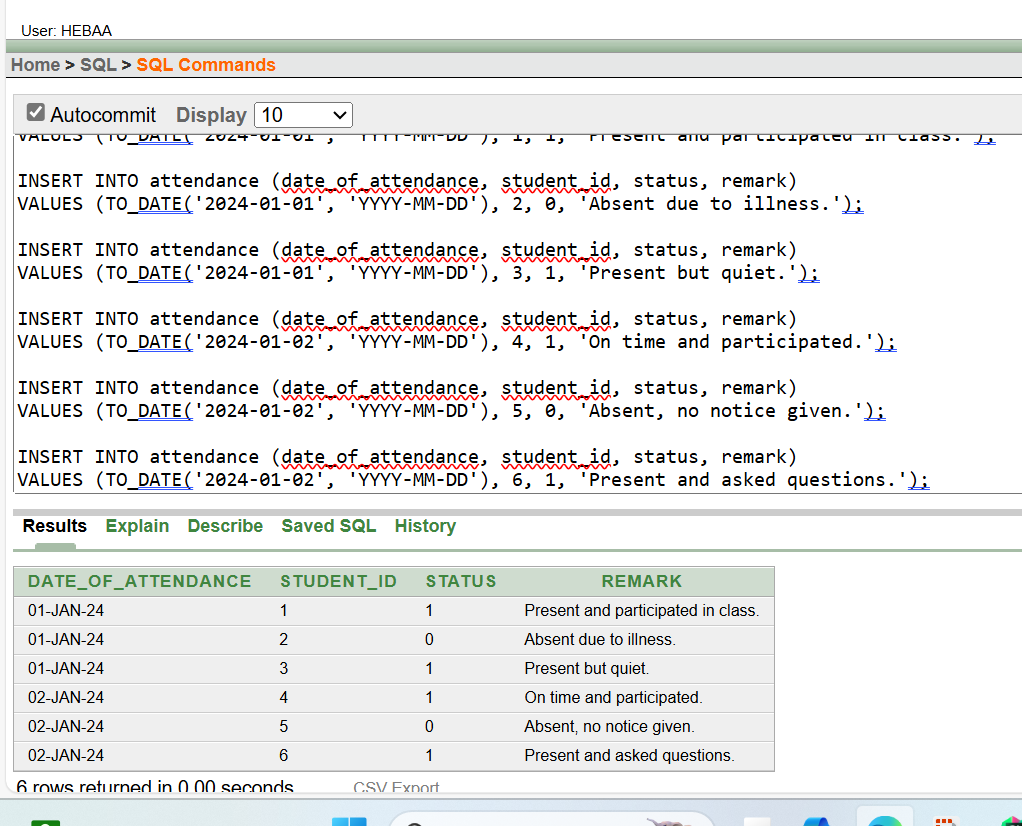










\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
**sql queries in this file**   


1. **Retrieve a list of all students in a specific classroom along with their grade:  
   select**

**c.classroom\_id as classroom\_id,**

**s.fname || ' ' || s.lname as student\_name,**

**g.name as grade\_name,**

**g.grade\_description as grade\_desc**

**from**

**student s**

**join**

**classroom\_student cs on s.student\_id = cs.student\_id**

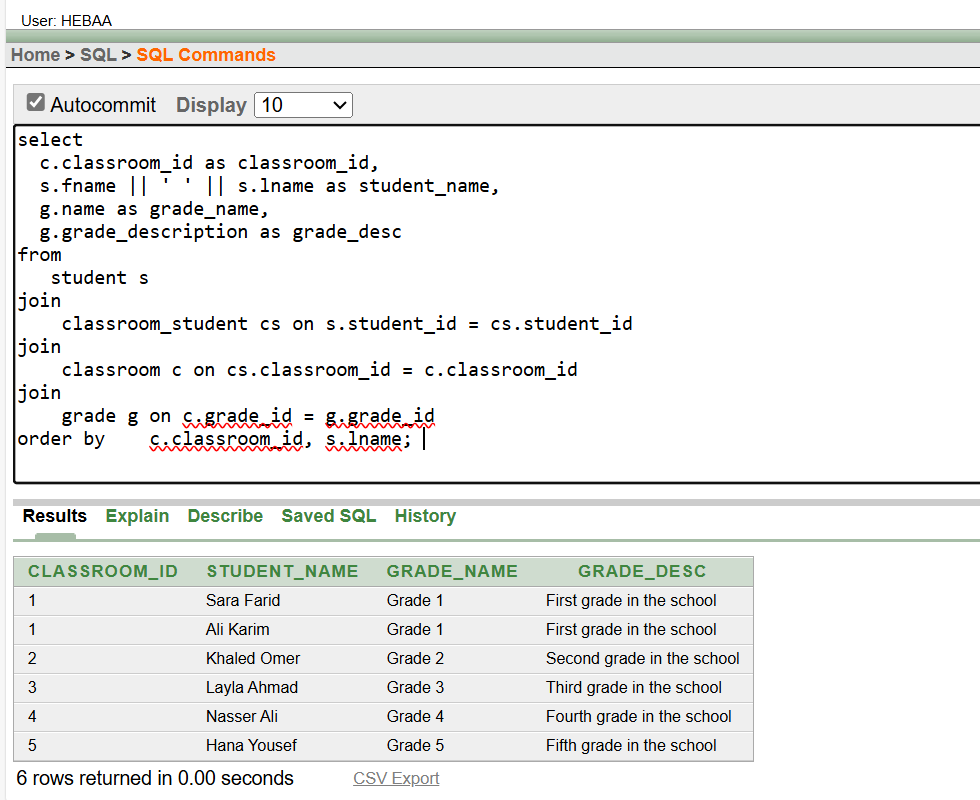
**join**

**classroom c on cs.classroom\_id = c.classroom\_id**

**join**

**grade g on c.grade\_id = g.grade\_id**

**order by c.classroom\_id, s.lname;**



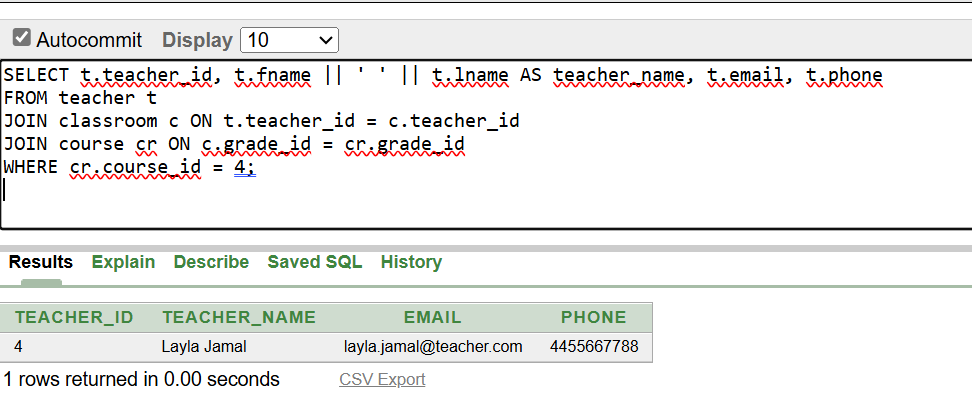
**2)Fetch the details of all teachers teaching a specific course.**

**SELECT t.teacher\_id, t.fname || ' ' || t.lname AS teacher\_name, t.email, t.phone**

**FROM teacher t**

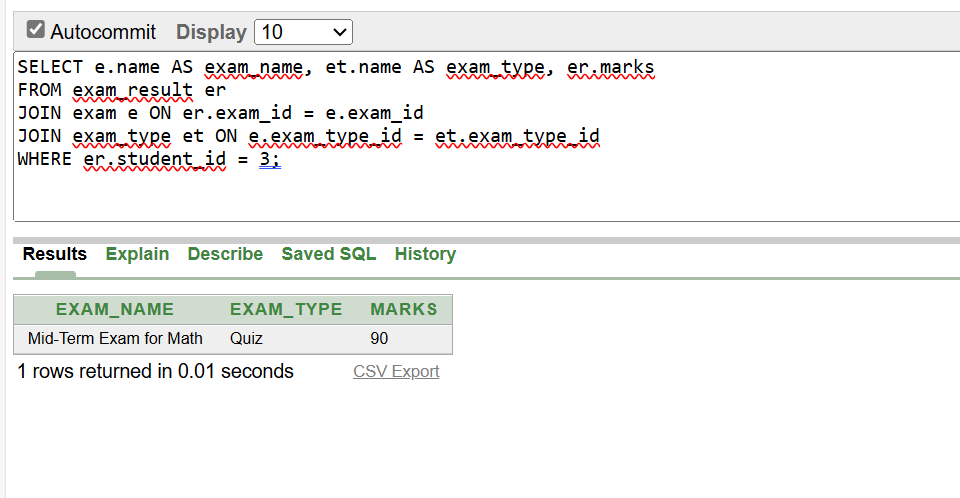
**JOIN classroom c ON t.teacher\_id = c.teacher\_id**

**JOIN course cr ON c.grade\_id = cr.grade\_id  
WHERE cr.course\_id = 4;**

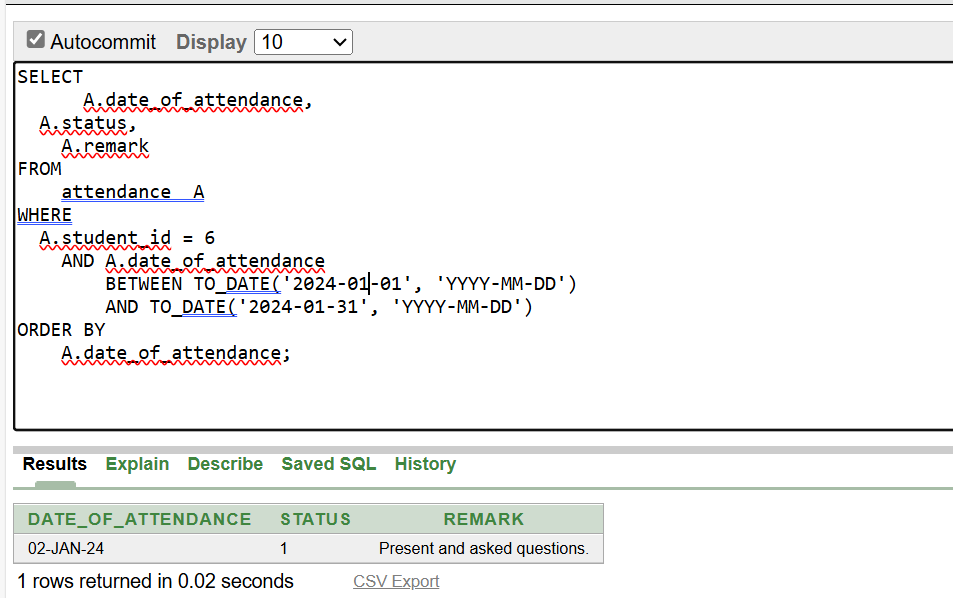


**3)Find all exams taken by a student along with their marks.:**

**SELECT e.name AS exam\_name, et.name AS exam\_type, er.marks  
FROM exam\_result er  
JOIN exam e ON er.exam\_id = e.exam\_id  
JOIN exam\_type et ON e.exam\_type\_id = et.exam\_type\_id  
WHERE er.student\_id = 3;**

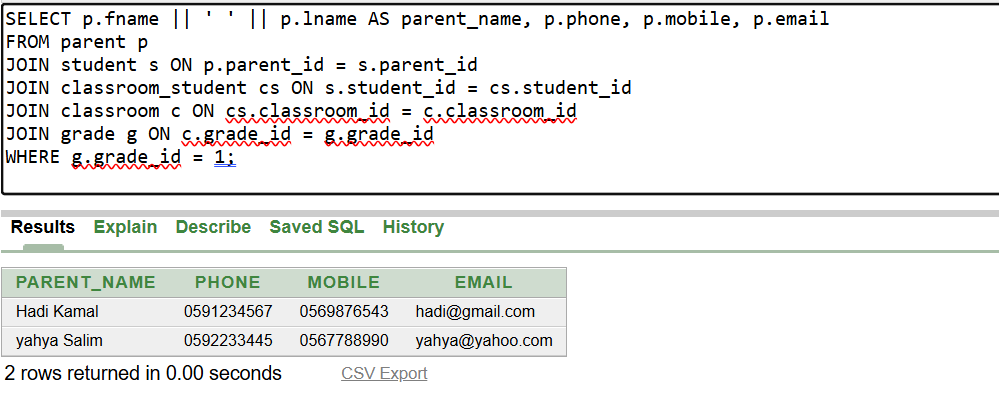
****

**4) . Retrieve the attendance records for a specific student within a given date range.  
SELECT A.date\_of\_attendance,   
A.status,   
A.remark  
FROM   
attendance A  
WHERE   
A.student\_id = 6 AND A.date\_of\_attendance   
 BETWEEN TO\_DATE('2024-01-01', 'YYYY-MM-DD')   
AND TO\_DATE('2024-01-31', 'YYYY-MM-DD')  
ORDER BY   
A.date\_of\_attendance;**

****

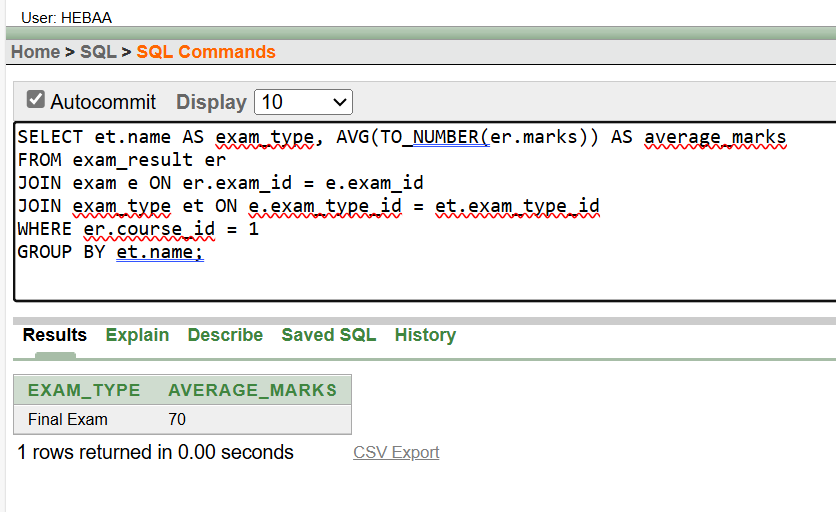
**5). List the parents' contact information for all students in a particular grade**

**SELECT p.fname || ' ' || p.lname AS parent\_name, p.phone, p.mobile, p.email  
FROM parent p  
JOIN student s ON p.parent\_id = s.parent\_id  
JOIN classroom\_student cs ON s.student\_id = cs.student\_id  
JOIN classroom c ON cs.classroom\_id = c.classroom\_id  
JOIN grade g ON c.grade\_id = g.grade\_id  
WHERE g.grade\_id = 1;**

****

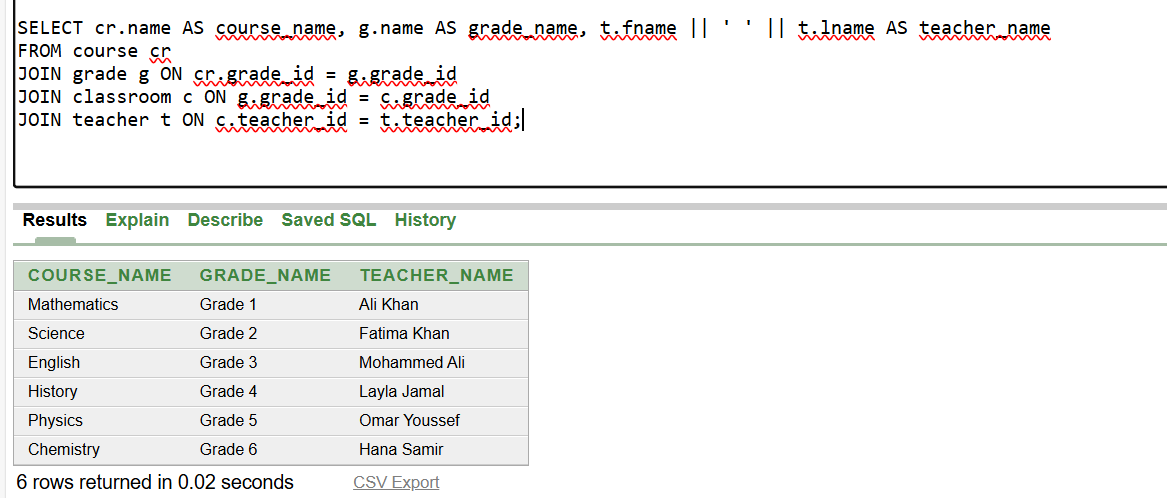
**6). Fetch the average marks for each exam type for a specific course.**

**SELECT et.name AS exam\_type, AVG(TO\_NUMBER(er.marks)) AS average\_marks  
FROM exam\_result er  
JOIN exam e ON er.exam\_id = e.exam\_id  
JOIN exam\_type et ON e.exam\_type\_id = et.exam\_type\_id  
WHERE er.course\_id = 1  
GROUP BY et.name;**

****

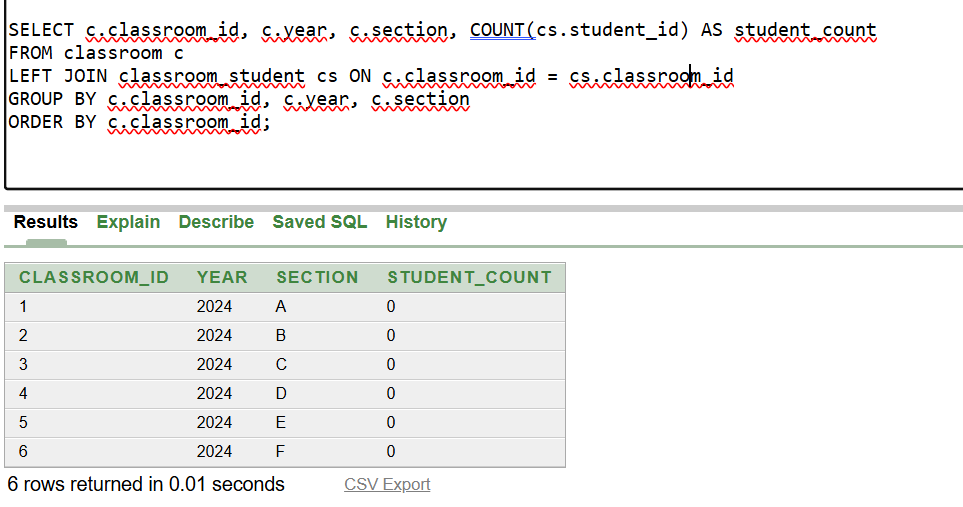
**7). Display all courses along with their associated grade and teacher.**

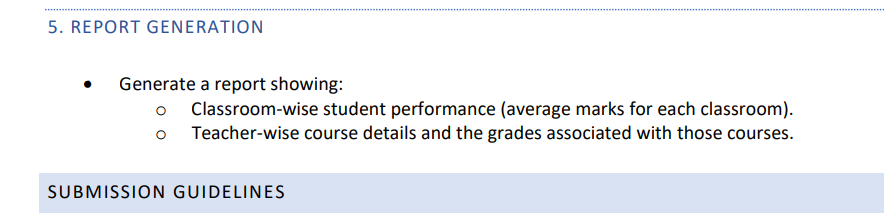
**SELECT cr.name AS course\_name, g.name AS grade\_name, t.fname || ' ' || t.lname AS teacher\_name  
FROM course cr  
JOIN grade g ON cr.grade\_id = g.grade\_id  
JOIN classroom c ON g.grade\_id = c.grade\_id  
JOIN teacher t ON c.teacher\_id = t.teacher\_id;**

****

**8). List all classrooms along with the number of students in each**

**SELECT c.classroom\_id, c.year, c.section, COUNT(cs.student\_id) AS student\_count  
FROM classroom c  
LEFT JOIN classroom\_student cs ON c.classroom\_id = cs.classroom\_id  
GROUP BY c.classroom\_id, c.year, c.section  
ORDER BY c.classroom\_id;**

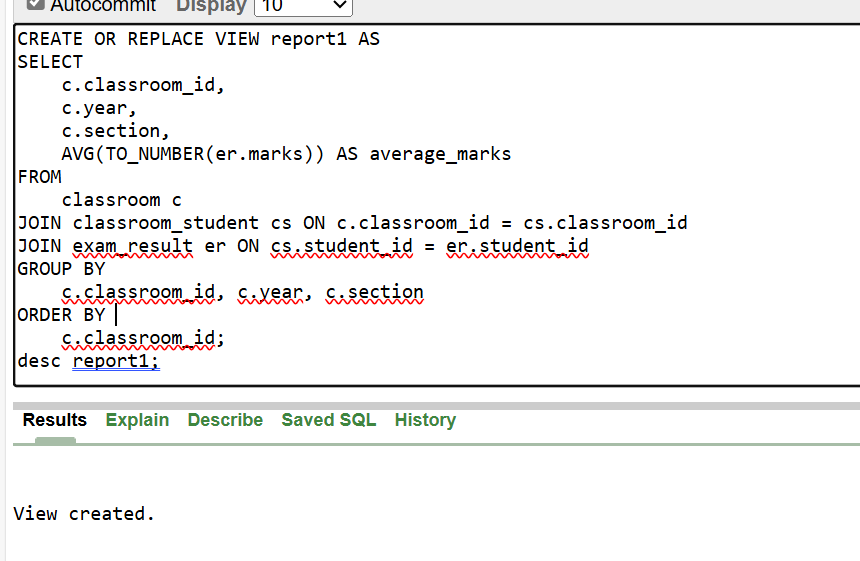
****

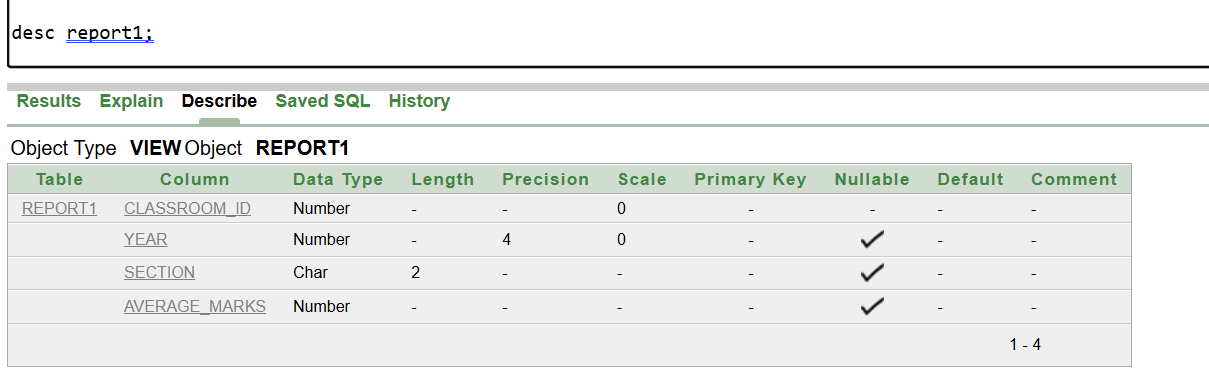


**Sql file**

  
**1.**

**CREATE OR REPLACE VIEW report1 AS  
SELECT   
c.classroom\_id,  
c.year,  
c.section,  
AVG(TO\_NUMBER(er.marks)) AS average\_marks  
FROM   
classroom c  
JOIN classroom\_student cs ON c.classroom\_id = cs.classroom\_id  
JOIN exam\_result er ON cs.student\_id = er.student\_id  
GROUP BY   
c.classroom\_id, c.year, c.section  
ORDER BY   
c.classroom\_id;  
desc report1;  
select \* from report1;**





**2.  
CREATE OR REPLACE VIEW report2 AS  
SELECT   
t.teacher\_id,  
t.fname || ' ' || t.lname AS teacher\_name,  
cr.course\_id,  
cr.name AS course\_name,  
g.grade\_id, g.name AS grade\_name FROM   
teacher t JOIN classroom c ON t.teacher\_id = c.teacher\_id  
JOIN grade g ON c.grade\_id = g.grade\_id  
JOIN course cr ON g.grade\_id = cr.grade\_id ORDER BY t.teacher\_id, cr.course\_id; desc report2; select \* from report2;**

