Student Data Base Design

database

Requirements

Our database is designed to manage information related to students, instructors, courses, enrollments, and grades within our educational institution.Student Information like Name, ID, email address, address should be securely stored, allowing for insertion and updation. Instructor Information also has been stored in different entity allows for tracking of the Instructors in the particular Courses. Courses stores the data of all the Courses offering by the education institute. For Enrollment, it stores every individual Enrollment ID, Course which ther are enrolled, Enrollment date. Grades stores the data of grades of every Student.This database design enables efficient management of student records, course information, enrollment data, and grading details, facilitating smooth operations within the educational institution.

Conceptul Design

Entities

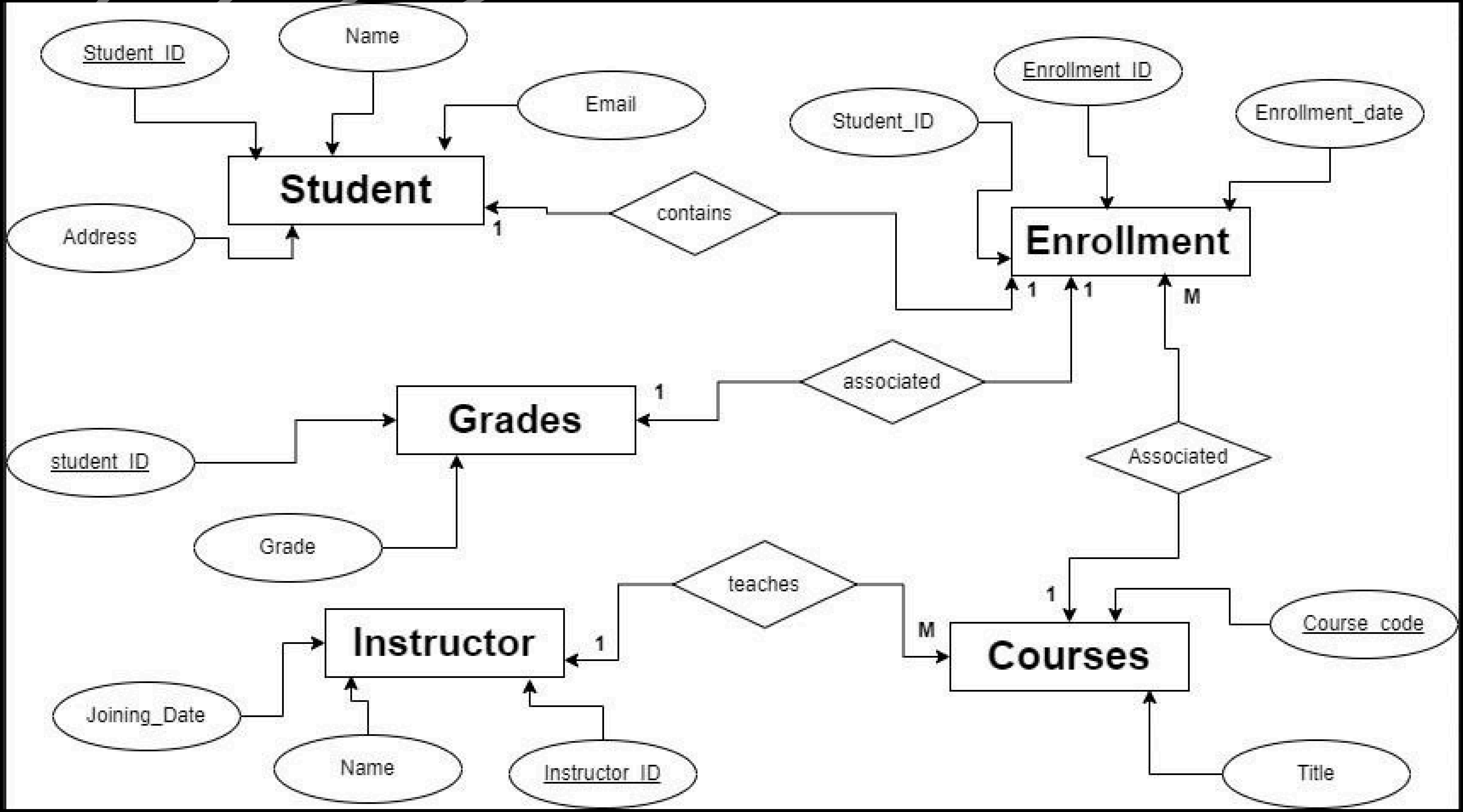
- Student
- Enrollment
- Instructor
- Courses
- Grades

Conceptual Design

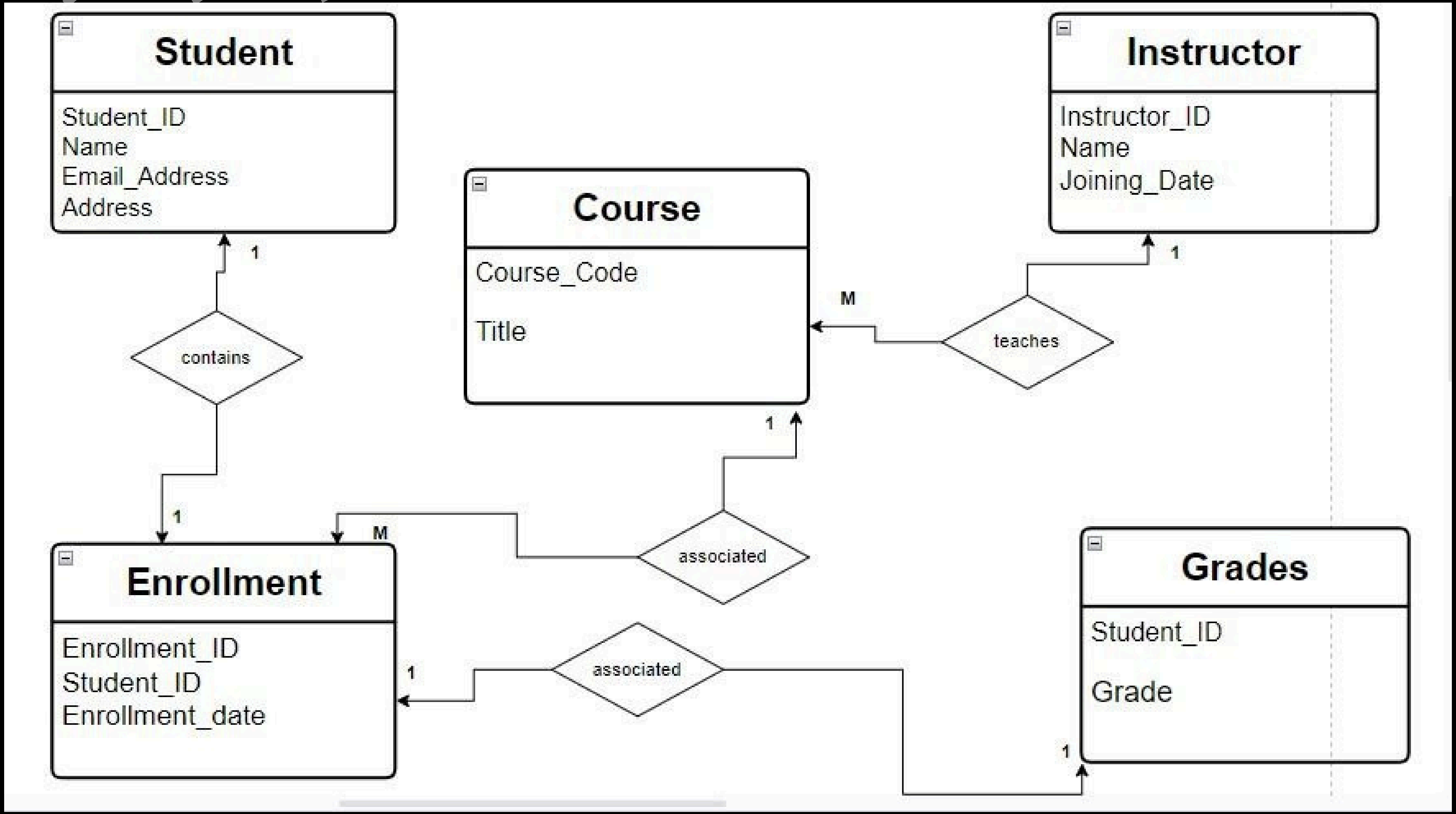
RELATIONSHIPS

- Student and Enrollment: One to One (Each student has One Enrollment ID)
- Courses and Enrollment: One to Many (Each course can have many Enrollments)
- Grades and Enrollment: One to One (Every Enrollment has assigned a grade)
- Instructor and Courses: One to Many (Each Instructor can teach more than one courses)

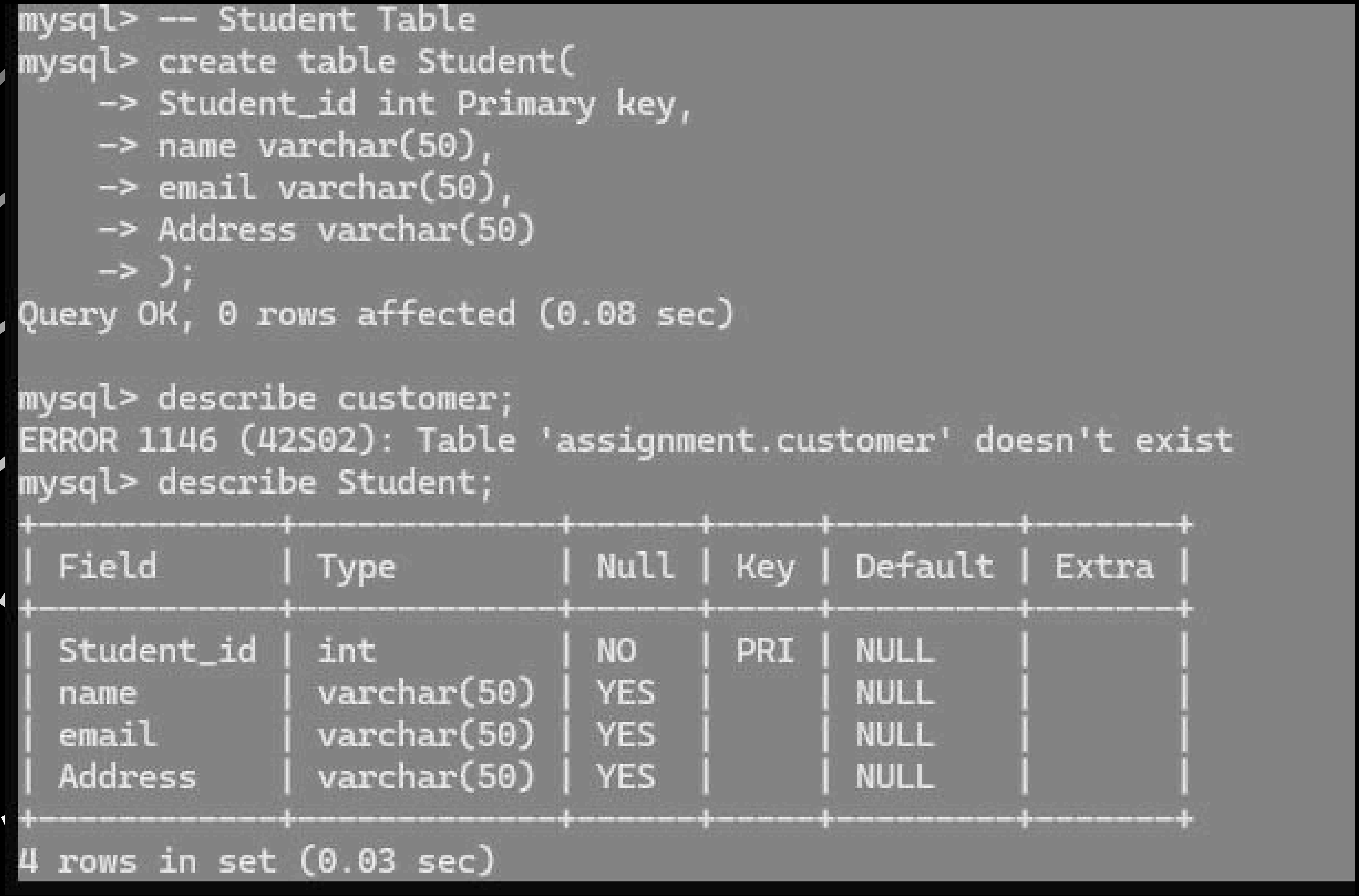
CONCEPTUAL DESIGN



LOGICAL DESIGN

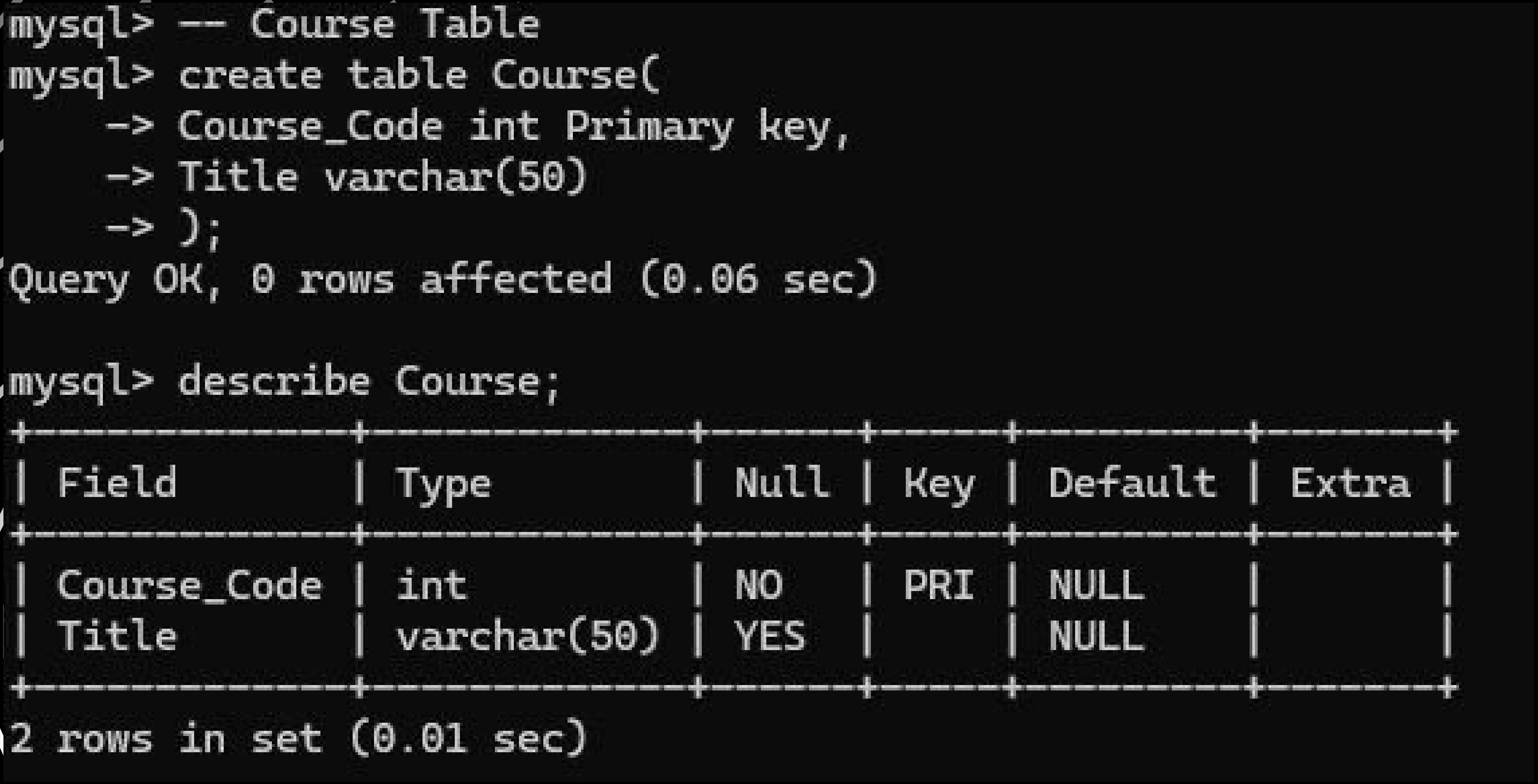


Physical Design (implementation)



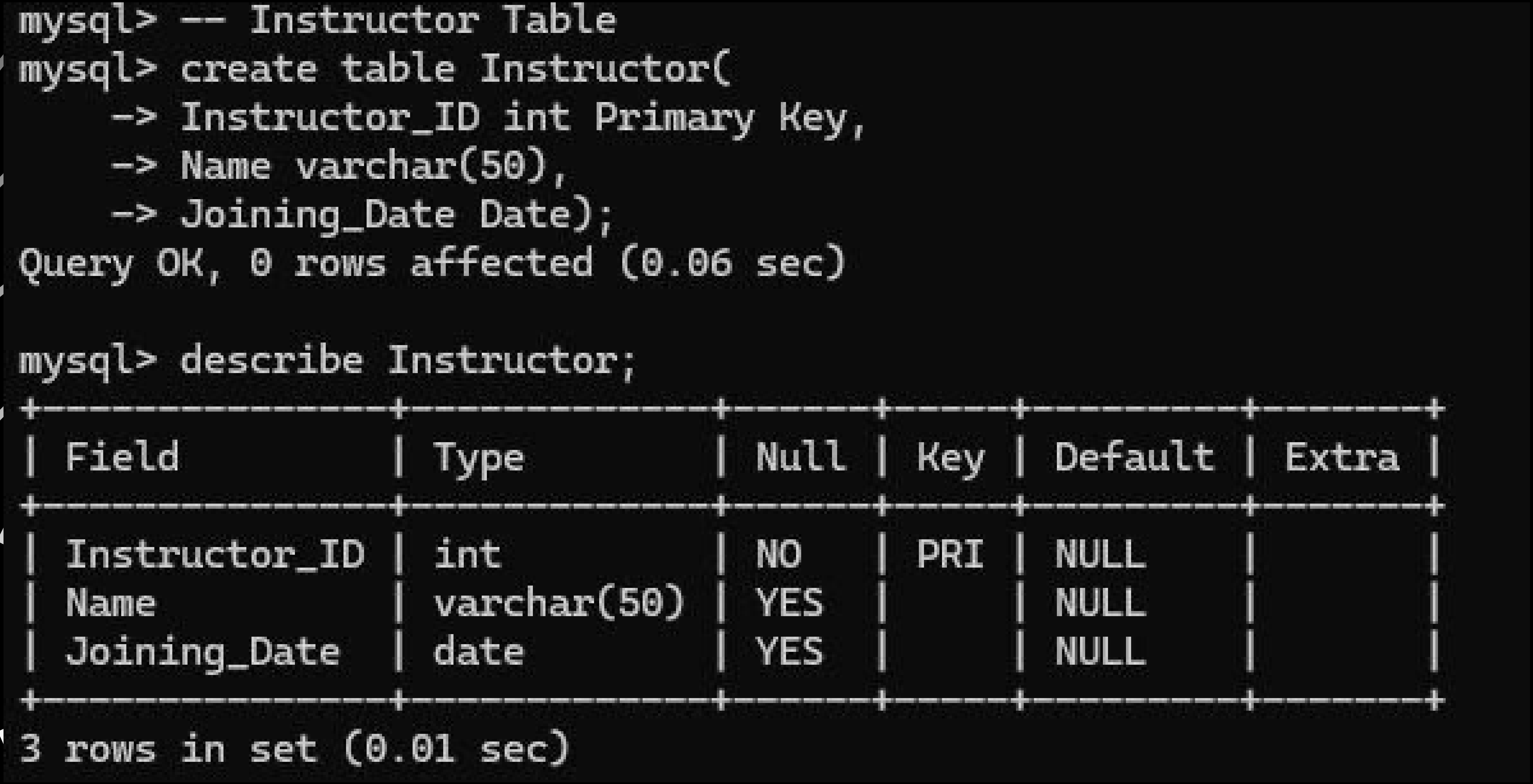
```
mysql> -- Student Table
mysql> create table Student(
    -> Student_id int Primary key,
    -> name varchar(50),
    -> email varchar(50),
    -> Address varchar(50)
    -> );
Query OK, 0 rows affected (0.08 sec)

mysql> describe customer;
ERROR 1146 (42S02): Table 'assignment.customer' doesn't exist
mysql> describe Student;
+------------+-------------+------+-----+---------+-------+
| Field      | Type        | Null | Key | Default | Extra |
+------------+-------------+------+-----+---------+-------+
| Student_id | int         | NO   | PRI | NULL    |       |
| name       | varchar(50) | YES  |     | NULL    |       |
| email      | varchar(50) | YES  |     | NULL    |       |
| Address    | varchar(50) | YES  |     | NULL    |       |
+------------+-------------+------+-----+---------+-------+
4 rows in set (0.03 sec)
```

Physical Design (implementation)



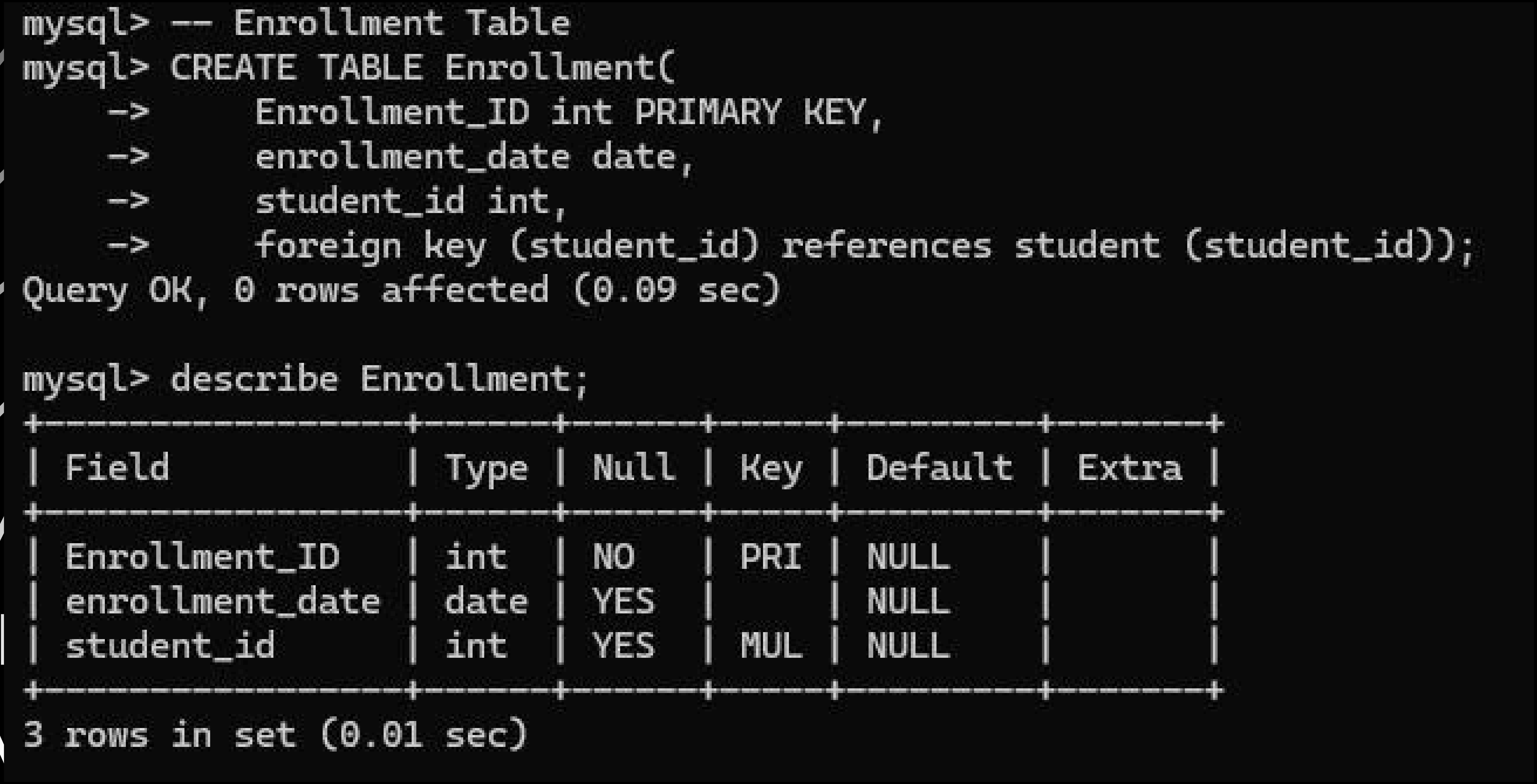
```
mysql> -- Course Table
mysql> create table Course(
    -> Course_Code int Primary key,
    -> Title varchar(50)
    -> );
Query OK, 0 rows affected (0.06 sec)

mysql> describe Course;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| Course_Code | int         | NO   | PRI | NULL    |       |
| Title       | varchar(50) | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
2 rows in set (0.01 sec)
```

Physical Design (implementation)



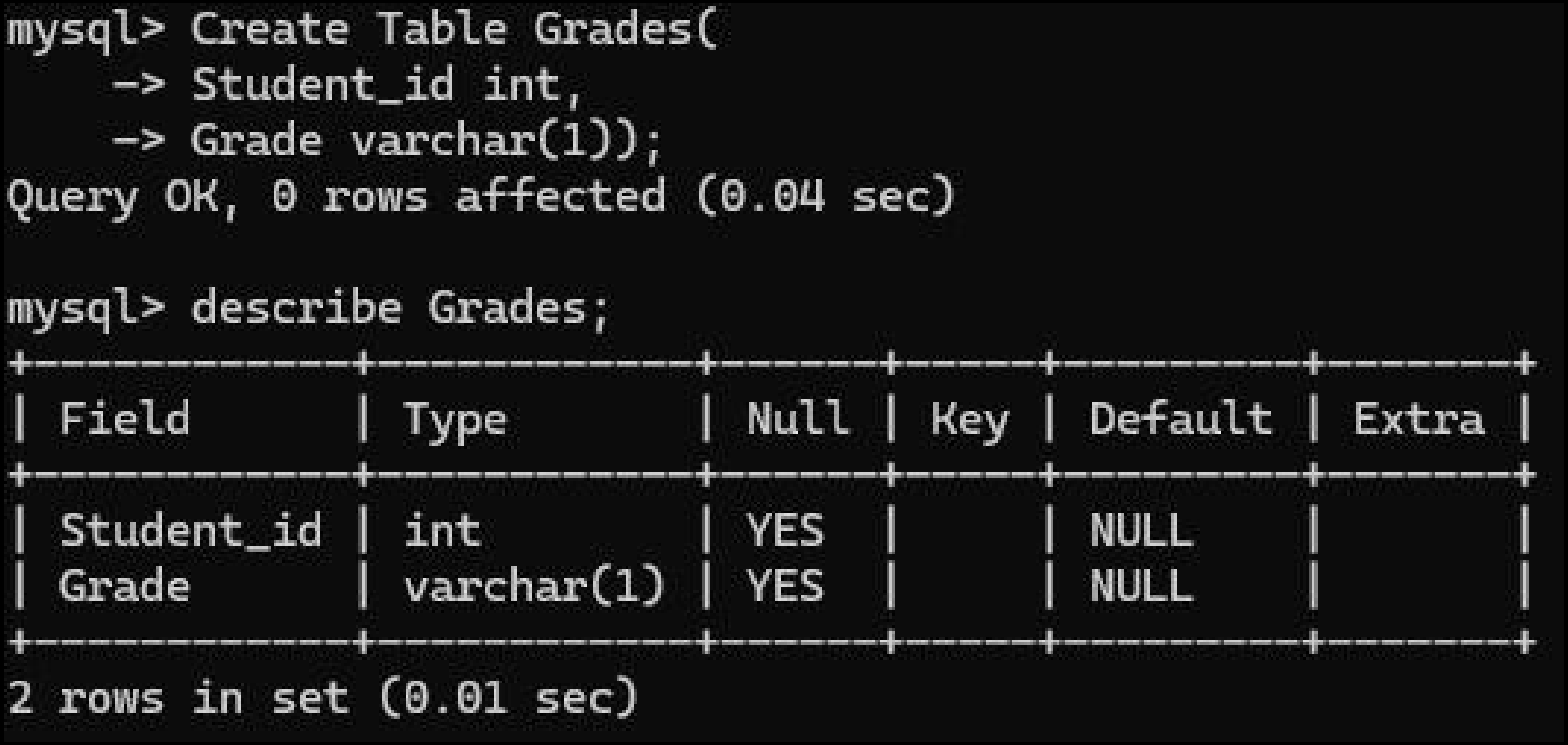
```
mysql> -- Instructor Table
mysql> create table Instructor(
    -> Instructor_ID int Primary Key,
    -> Name varchar(50),
    -> Joining_Date Date);
Query OK, 0 rows affected (0.06 sec)

mysql> describe Instructor;
+---------------+-------------+------+-----+---------+-------+
| Field         | Type        | Null | Key | Default | Extra |
+---------------+-------------+------+-----+---------+-------+
| Instructor_ID | int         | NO   | PRI | NULL    |       |
| Name          | varchar(50) | YES  |     | NULL    |       |
| Joining_Date  | date        | YES  |     | NULL    |       |
+---------------+-------------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

Physical Design (implementation)



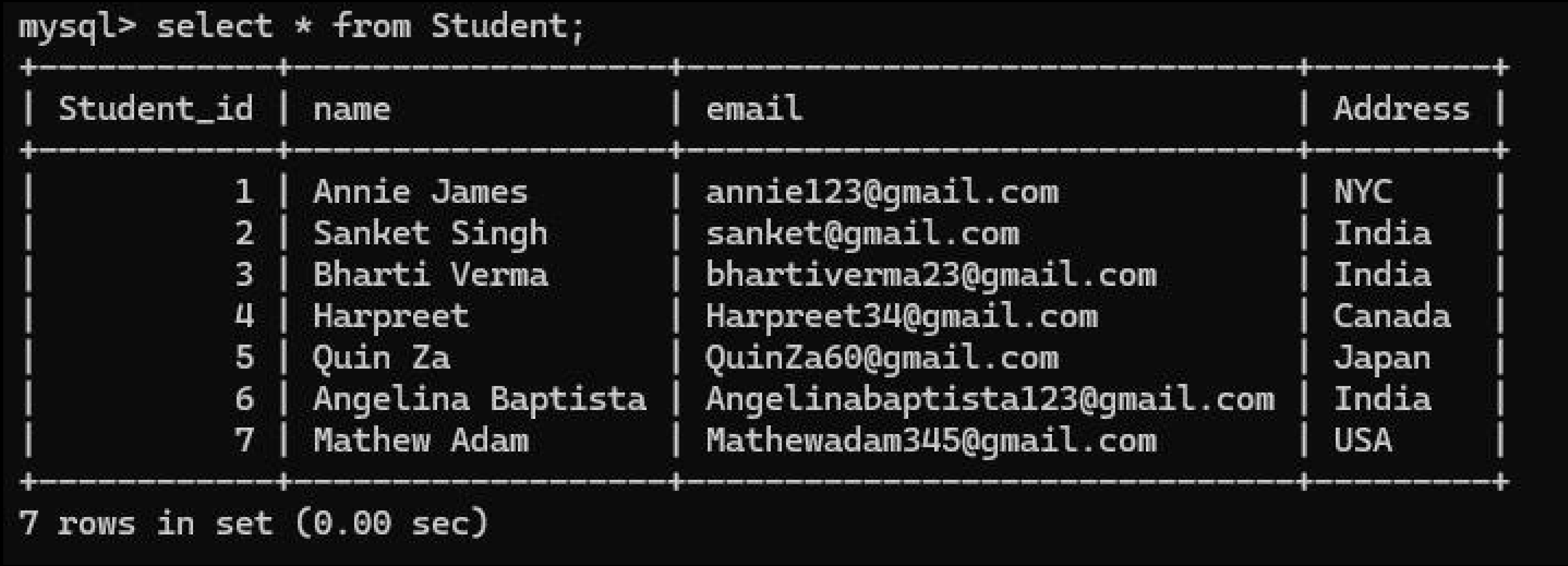
```
mysql> -- Enrollment Table
mysql> CREATE TABLE Enrollment(
    ->     Enrollment_ID int PRIMARY KEY,
    ->     enrollment_date date,
    ->     student_id int,
    ->     foreign key (student_id) references student (student_id));
Query OK, 0 rows affected (0.09 sec)

mysql> describe Enrollment;
+-----------------+------+------+-----+---------+-------+
| Field           | Type | Null | Key | Default | Extra |
+-----------------+------+------+-----+---------+-------+
| Enrollment_ID   | int  | NO   | PRI | NULL    |       |
| enrollment_date | date | YES  |     | NULL    |       |
| student_id      | int  | YES  | MUL | NULL    |       |
+-----------------+------+------+-----+---------+-------+
3 rows in set (0.01 sec)
```

Physical Design (implementation)



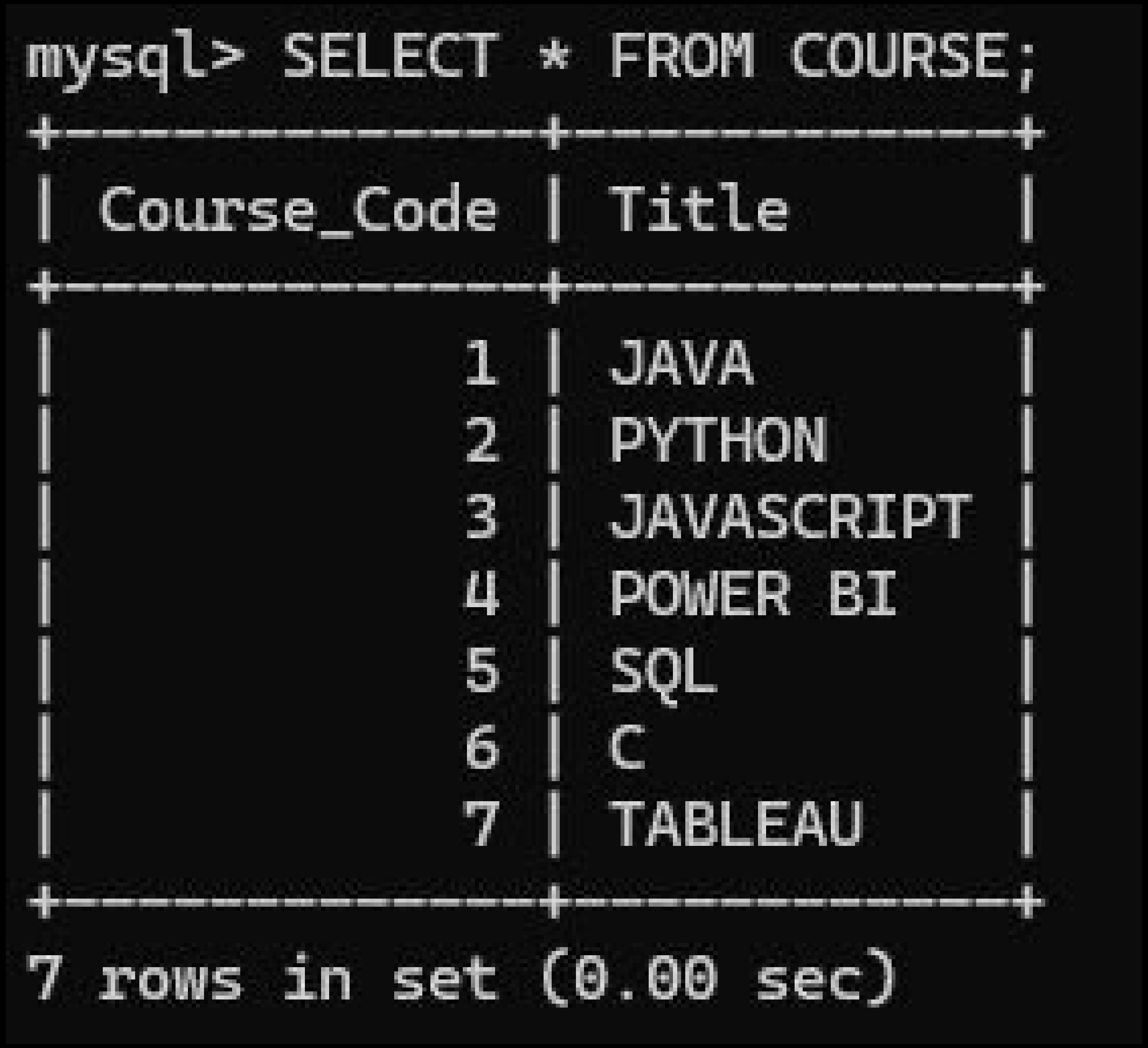
```
mysql> Create Table Grades(
    -> Student_id int,
    -> Grade varchar(1));
Query OK, 0 rows affected (0.04 sec)

mysql> describe Grades;
+------------+------------+------+-----+---------+-------+
| Field      | Type       | Null | Key | Default | Extra |
+------------+------------+------+-----+---------+-------+
| Student_id | int        | YES  |     | NULL    |       |
| Grade      | varchar(1) | YES  |     | NULL    |       |
+------------+------------+------+-----+---------+-------+
2 rows in set (0.01 sec)
```

Physical Design (implementation)

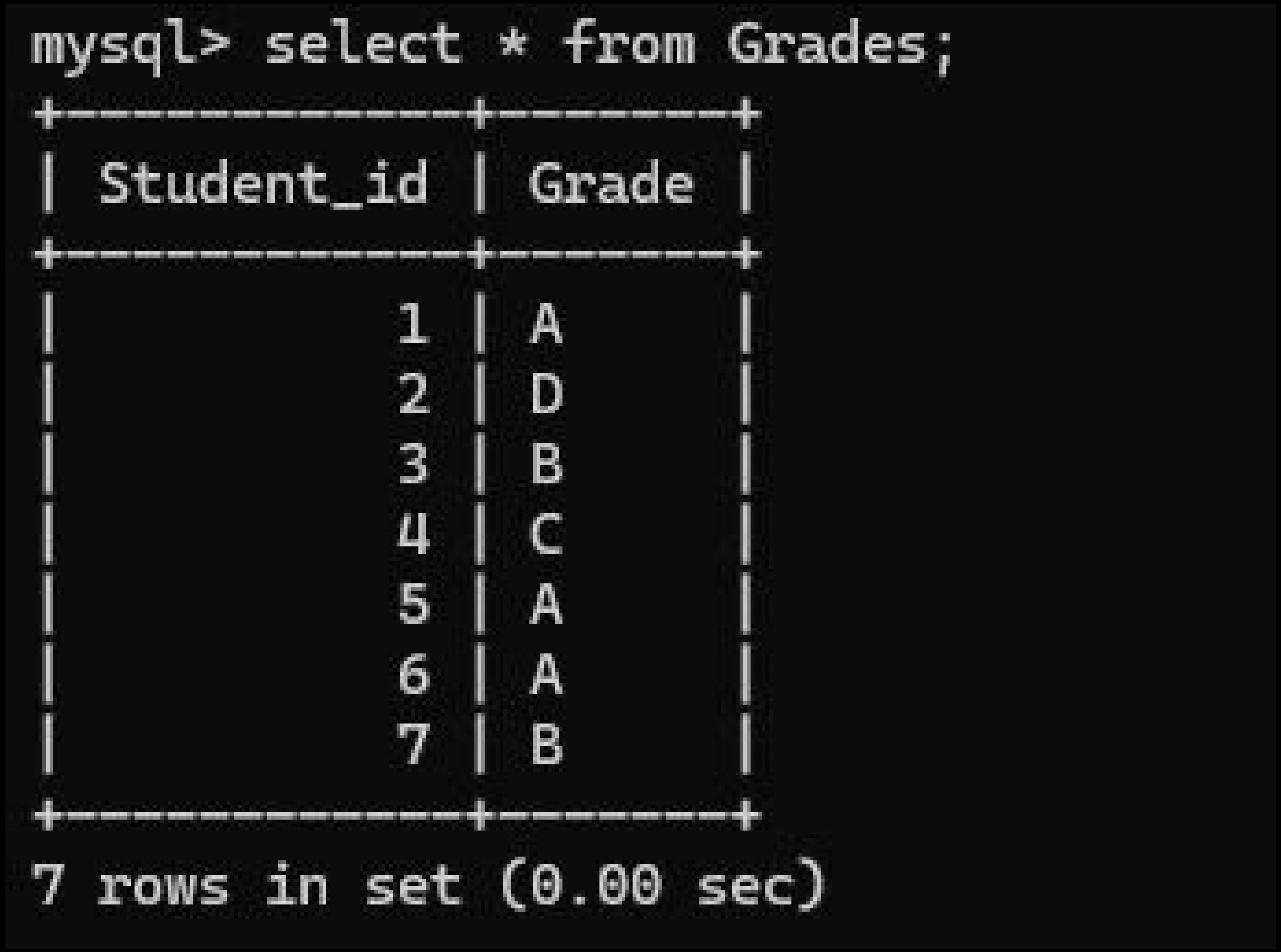


```
mysql> select * from Student;
+------------+-------------------+-------------------------------+---------+
| Student_id | name              | email                         | Address |
+------------+-------------------+-------------------------------+---------+
|          1 | Annie James       | annie123@gmail.com            | NYC     |
|          2 | Sanket Singh      | sanket@gmail.com              | India   |
|          3 | Bharti Verma      | bhartiverma23@gmail.com       | India   |
|          4 | Harpreet          | Harpreet34@gmail.com          | Canada  |
|          5 | Quin Za           | QuinZa60@gmail.com            | Japan   |
|          6 | Angelina Baptista | Angelinabaptista123@gmail.com | India   |
|          7 | Mathew Adam       | Mathewadam345@gmail.com       | USA     |
+------------+-------------------+-------------------------------+---------+
7 rows in set (0.00 sec)
```

Physical Design (implementation)

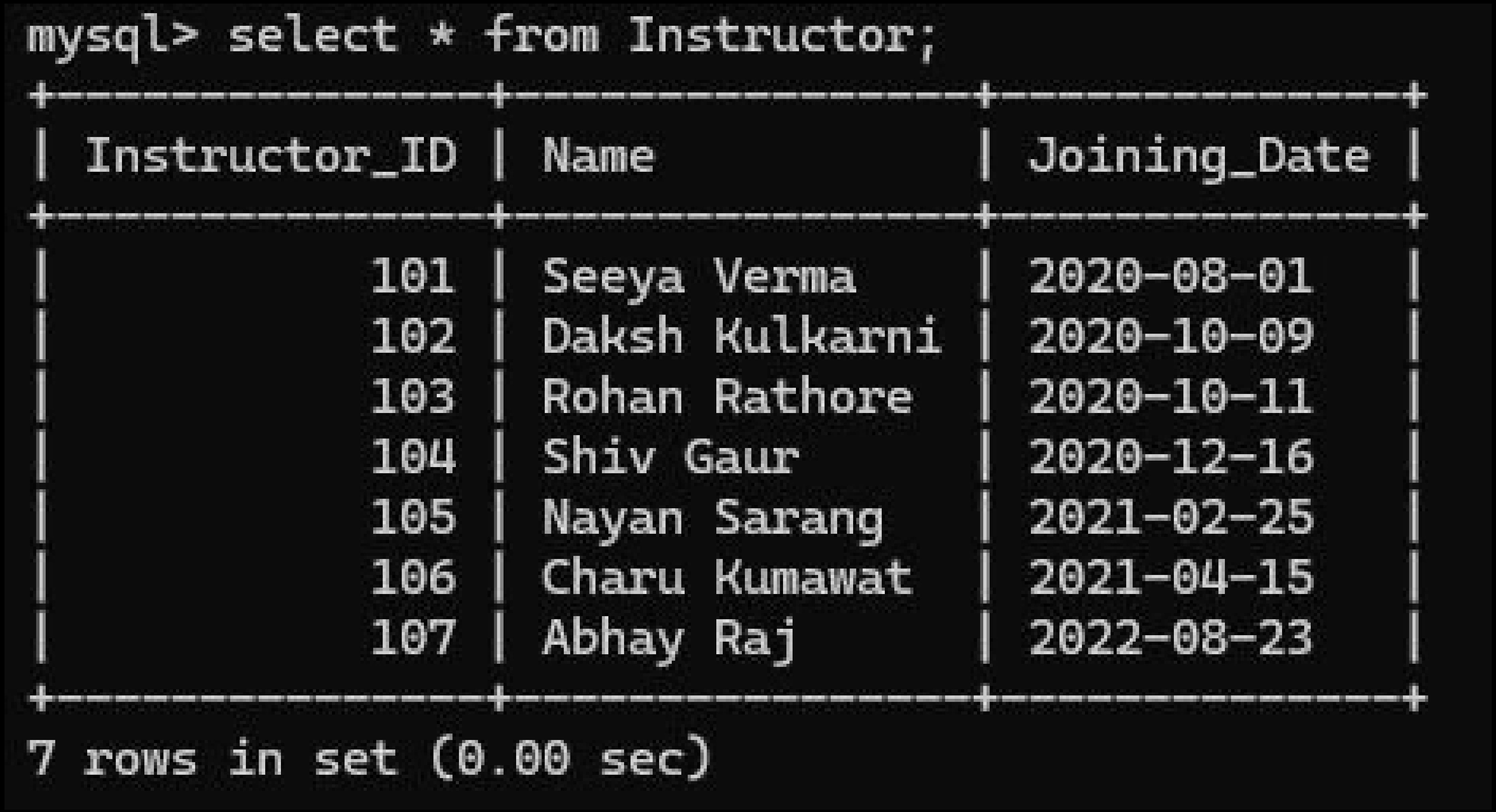


```
mysql> SELECT * FROM COURSE;
+-------------+------------+
| Course_Code | Title      |
+-------------+------------+
|           1 | JAVA       |
|           2 | PYTHON     |
|           3 | JAVASCRIPT |
|           4 | POWER BI   |
|           5 | SQL        |
|           6 | C          |
|           7 | TABLEAU    |
+-------------+------------+
7 rows in set (0.00 sec)
```



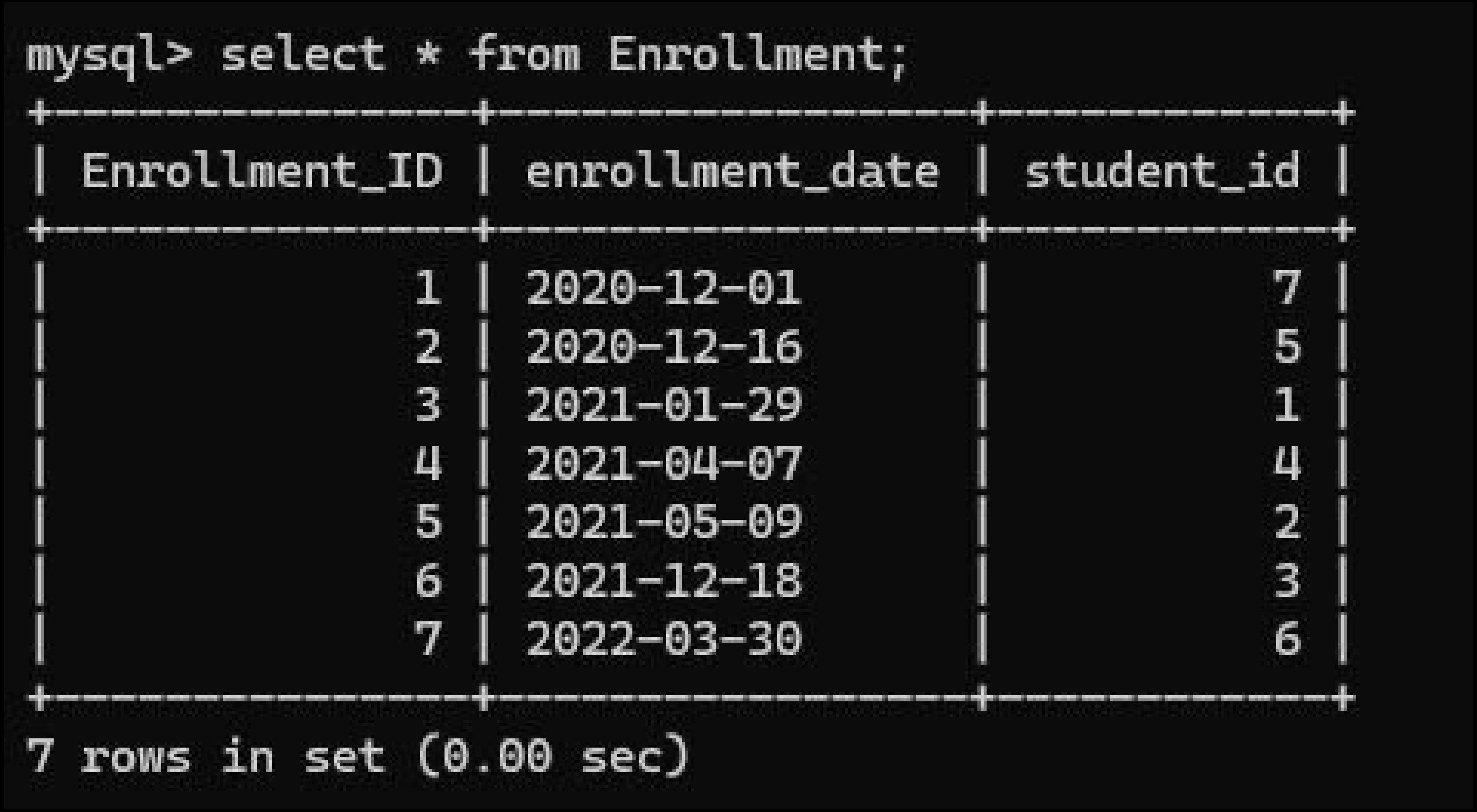
```
mysql> select * from Grades;
+------------+-------+
| Student_id | Grade |
+------------+-------+
|          1 | A     |
|          2 | D     |
|          3 | B     |
|          4 | C     |
|          5 | A     |
|          6 | A     |
|          7 | B     |
+------------+-------+
7 rows in set (0.00 sec)
```

Physical Design (implementation)



```
mysql> select * from Instructor;
+---------------+----------------+--------------+
| Instructor_ID | Name           | Joining_Date |
+---------------+----------------+--------------+
|           101 | Seeya Verma    | 2020-08-01   |
|           102 | Daksh Kulkarni | 2020-10-09   |
|           103 | Rohan Rathore  | 2020-10-11   |
|           104 | Shiv Gaur      | 2020-12-16   |
|           105 | Nayan Sarang   | 2021-02-25   |
|           106 | Charu Kumawat  | 2021-04-15   |
|           107 | Abhay Raj      | 2022-08-23   |
+---------------+----------------+--------------+
7 rows in set (0.00 sec)
```

Physical Design (implementation)



```
mysql> select * from Enrollment;
+---------------+-----------------+------------+
| Enrollment_ID | enrollment_date | student_id |
+---------------+-----------------+------------+
|             1 | 2020-12-01      |          7 |
|             2 | 2020-12-16      |          5 |
|             3 | 2021-01-29      |          1 |
|             4 | 2021-04-07      |          4 |
|             5 | 2021-05-09      |          2 |
|             6 | 2021-12-18      |          3 |
|             7 | 2022-03-30      |          6 |
+---------------+-----------------+------------+
7 rows in set (0.00 sec)
```

CONCLUSION

The student database design outlined in this project provides a robust framework for efficiently managing student information within an educational institution. By carefully considering the various entities, attributes, and relationships, we have created a structured database schema that captures essential data about students, courses, grades, and Instructors. Through the use of appropriate data types, constraints, and normalization techniques, we ensure data integrity, minimize redundancy, and optimize query performance. Additionally, by incorporating features such as authentication and access control, we enhance security and privacy.