DBMS Project

Title:-Subject Management System

Aditya Tomar (202051012)

Aniket Yadav (202051025)

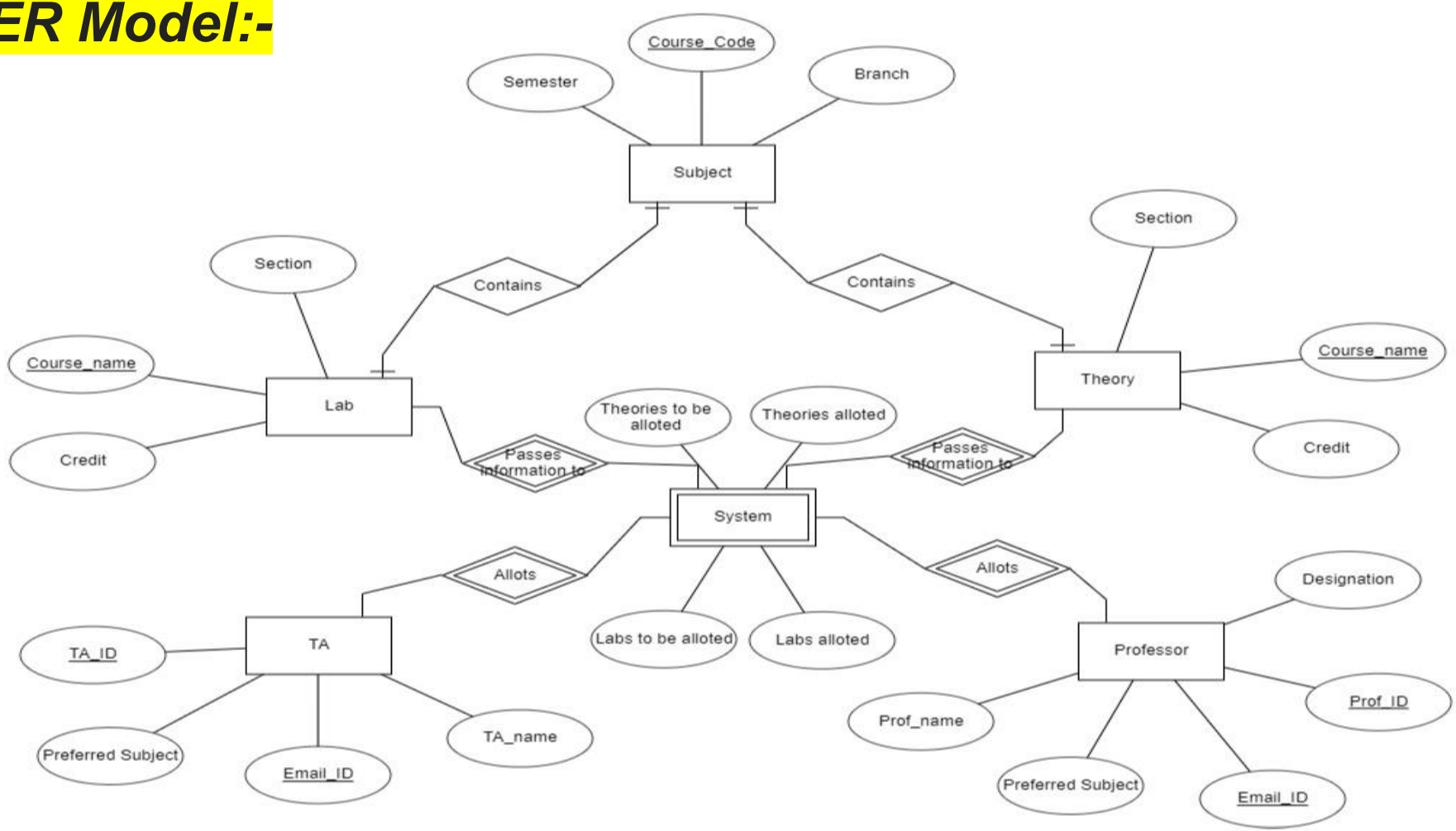
Avnit Kumar Anand (202051040)

Kewal Delhiwala (202051060)

Vikram Kumar (202052346)

TA:-Amit Dwivedi

ER Model:-


Course_Code
Semester
Branch
Subject
Section
Contains
Contains
Section
Course_name
Lab
Theory
Course_name
Credit
Credit
Theories to be alloted
Theories alloted
Passes information to
Passes information to
System
Allots
Allots
Designation
TA
Labs to be alloted
Labs alloted
Professor
TA_ID
Prof_ID
Prof_name
Preferred Subject
TA_name
Email_ID
Preferred Subject
Email_ID

EER Model Concepts Used:-

1- Sub Class and Super Class

2- Specialization and Generalization

3- Attribute and relationship inheritance

4-Aggregation

Steps followed to convert ER and EER model concepts to the Relational Model:-

1- Mapping Entity

Mapping Process (Algorithm)

- Create table for each entity.
- Entity's attributes should become fields of tables with their respective data types.
- Declare primary key.

2- Mapping Relationship

Mapping Process

- Create table for a relationship.
- Add the primary keys of all participating Entities as fields of table with their respective data types.
- If relationship has any attribute, add each attribute as field of table.
- Declare a primary key composing all the primary keys of participating entities.
- Declare all foreign key constraints.

4- Mapping Hierarchical Entities

Mapping Process

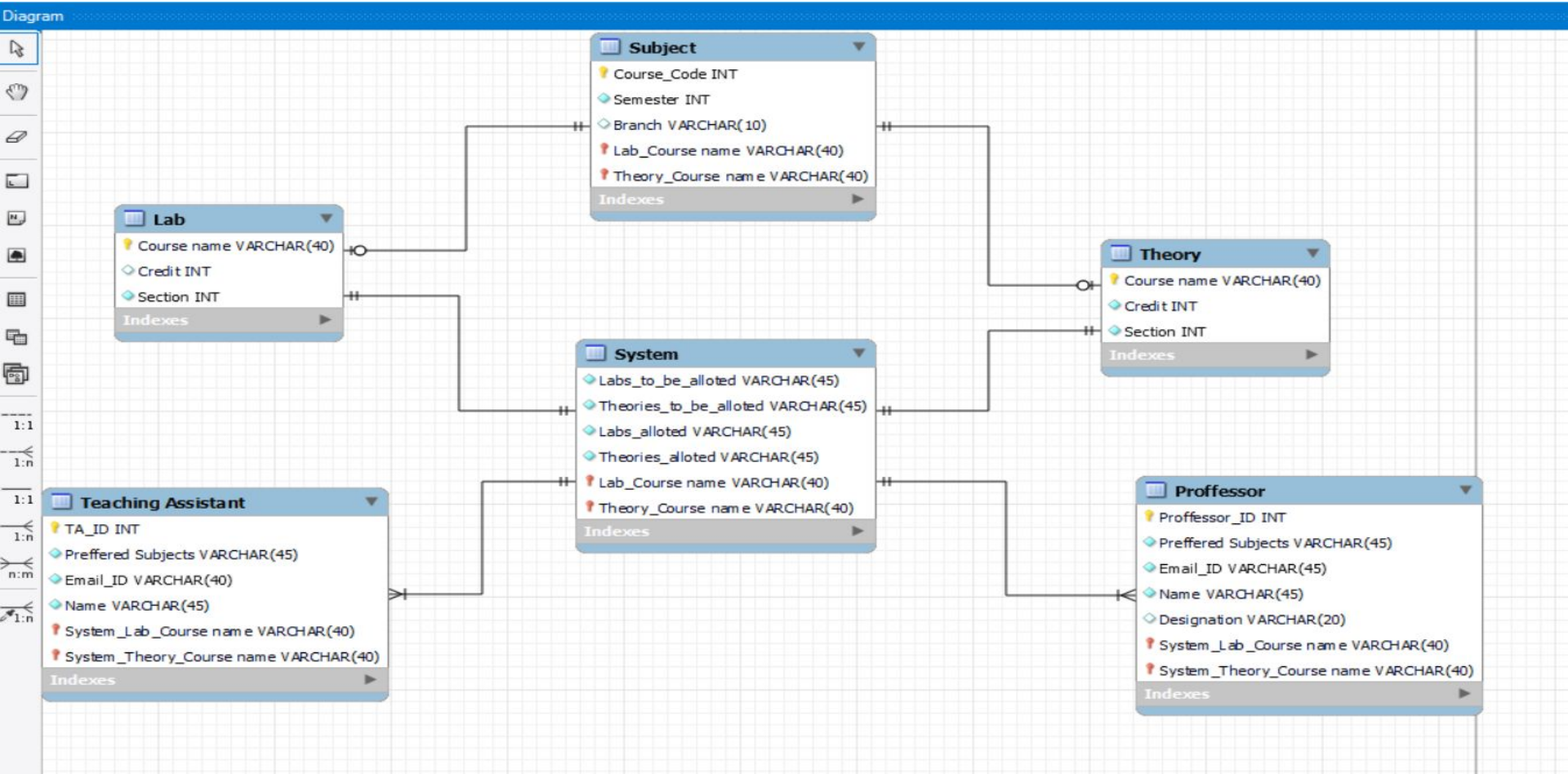
- Create tables for all higher-level entities.
- Create tables for lower-level entities.
- Add primary keys of higher-level entities in the table of lower-level entities.
- In lower-level tables, add all other attributes of lower-level entities.
- Declare primary key of higher-level table and the primary key for lower-level table.
- Declare foreign key constraints.

3-Mapping Weak Entity Sets

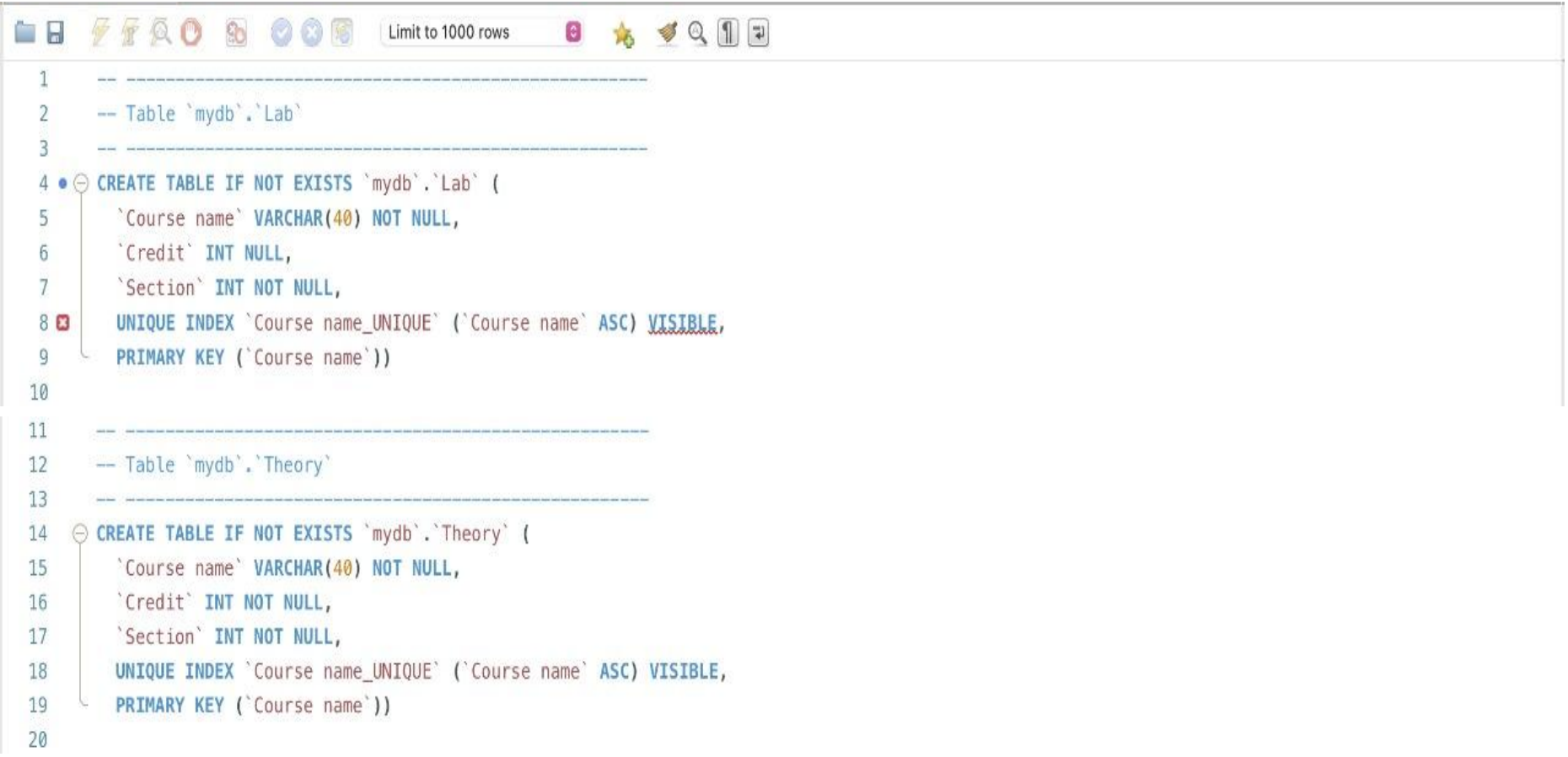
Mapping Process

- Create table for weak entity set.
- Add all its attributes to table as field.
- Add the primary key of identifying entity set.
- Declare all foreign key constraints.

Relational Schema Diagram



Screenshots of all the queries executed to implement your Relational Model



```
-- -----------------------------------------------------
-- Table `mydb`.`System`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`System` (
  `Labs_to_be_alloted` VARCHAR(45) NOT NULL,
  `Theories_to_be_alloted` VARCHAR(45) NOT NULL,
  `Labs_alloted` VARCHAR(45) NOT NULL,
  `Theories_alloted` VARCHAR(45) NOT NULL,
  `Lab_Course name` VARCHAR(40) NOT NULL,
  `Theory_Course name` VARCHAR(40) NOT NULL,
  UNIQUE INDEX `Subjects_to_allot_UNIQUE` (`Labs_to_be_alloted` ASC) VISIBLE,
  UNIQUE INDEX `Subjects_alloted_UNIQUE` (`Theories_to_be_alloted` ASC) VISIBLE,
  UNIQUE INDEX `Labs_alloted_UNIQUE` (`Labs_alloted` ASC) VISIBLE,
  UNIQUE INDEX `Theories_alloted_UNIQUE` (`Theories_alloted` ASC) VISIBLE,
  PRIMARY KEY (`Lab_Course name`, `Theory_Course name`),
  INDEX `fk_System_Theory1_idx` (`Theory_Course name` ASC) VISIBLE,
  CONSTRAINT `fk_System_Lab1`
    FOREIGN KEY (`Lab_Course name`)
    REFERENCES `mydb`.`Lab` (`Course name`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_System_Theory1`
    FOREIGN KEY (`Theory_Course name`)
    REFERENCES `mydb`.`Theory` (`Course name`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
```

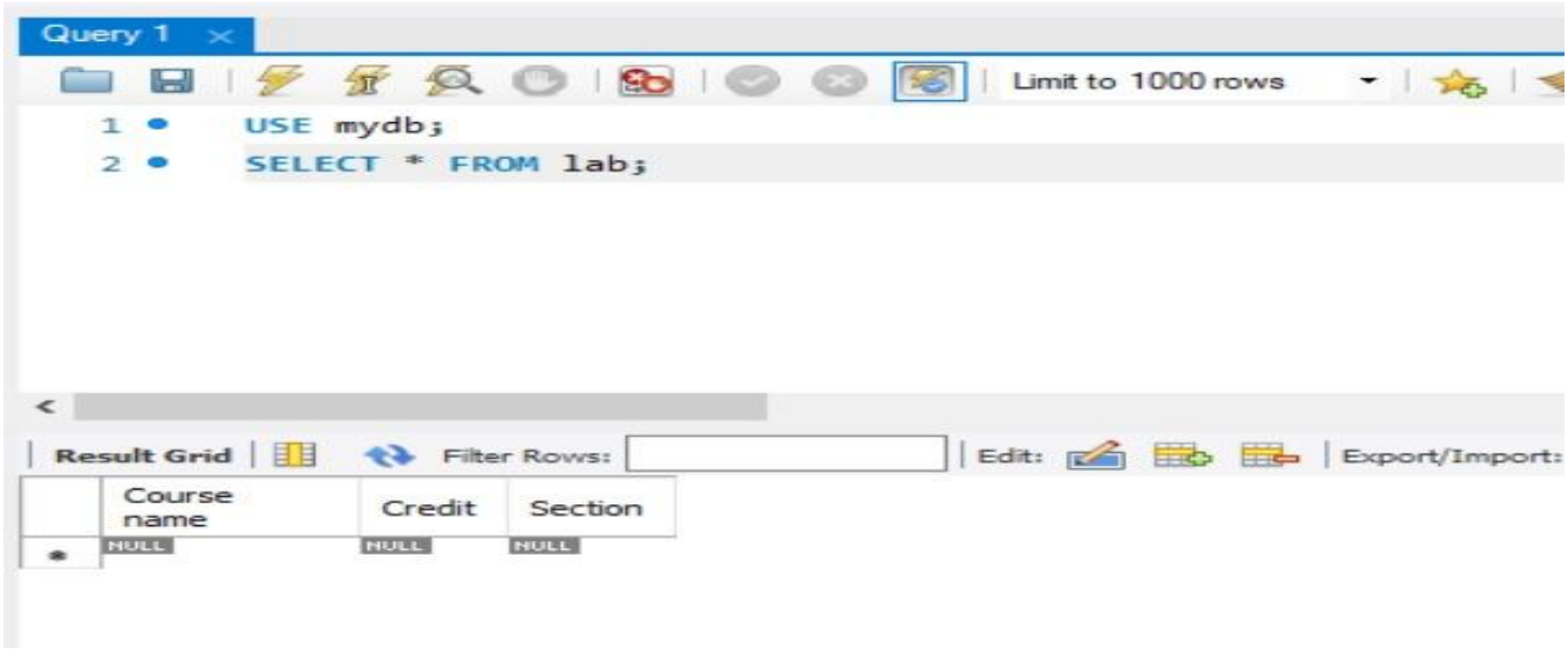
```sql
-- -----------------------------------------------------
-- Table `mydb`.`Teaching Assistant`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Teaching Assistant` (
  `TA_ID` INT NOT NULL,
  `Preffered Subjects` VARCHAR(45) NOT NULL,
  `Email_ID` VARCHAR(40) NOT NULL,
  `Name` VARCHAR(45) NOT NULL,
  `System_Lab_Course name` VARCHAR(40) NOT NULL,
  `System_Theory_Course name` VARCHAR(40) NOT NULL,
  PRIMARY KEY (`TA_ID`, `System_Lab_Course name`, `System_Theory_Course name`),
  UNIQUE INDEX `TA_ID_UNIQUE` (`TA_ID` ASC) VISIBLE,
  UNIQUE INDEX `Email_ID_UNIQUE` (`Email_ID` ASC) VISIBLE,
  INDEX `fk_Teaching Assistant_System1_idx` (`System_Lab_Course name` ASC, `System_Theory_Course name` ASC) VISIBLE,
  CONSTRAINT `fk_Teaching Assistant_System1`
    FOREIGN KEY (`System_Lab_Course name` , `System_Theory_Course name`)
    REFERENCES `mydb`.`System` (`Lab_Course name` , `Theory_Course name`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)

```

```sql
-- -----------------------------------------------------
-- Table `mydb`.`Proffessor`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Proffessor` (
  `Proffessor_ID` INT NOT NULL,
  `Preffered Subjects` VARCHAR(45) NOT NULL,
  `Email_ID` VARCHAR(45) NOT NULL,
  `Name` VARCHAR(45) NOT NULL,
  `Designation` VARCHAR(20) NULL,
  `System_Lab_Course name` VARCHAR(40) NOT NULL,
  `System_Theory_Course name` VARCHAR(40) NOT NULL,
  PRIMARY KEY (`Proffessor_ID`, `System_Lab_Course name`, `System_Theory_Course name`),
  UNIQUE INDEX `Proffessor_ID_UNIQUE` (`Proffessor_ID` ASC) VISIBLE,
  UNIQUE INDEX `Email_ID_UNIQUE` (`Email_ID` ASC) VISIBLE,
  INDEX `fk_Proffessor_System1_idx` (`System_Lab_Course name` ASC, `System_Theory_Course name` ASC) VISIBLE,
  CONSTRAINT `fk_Proffessor_System1`
    FOREIGN KEY (`System_Lab_Course name` , `System_Theory_Course name`)
    REFERENCES `mydb`.`System` (`Lab_Course name` , `Theory_Course name`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
```

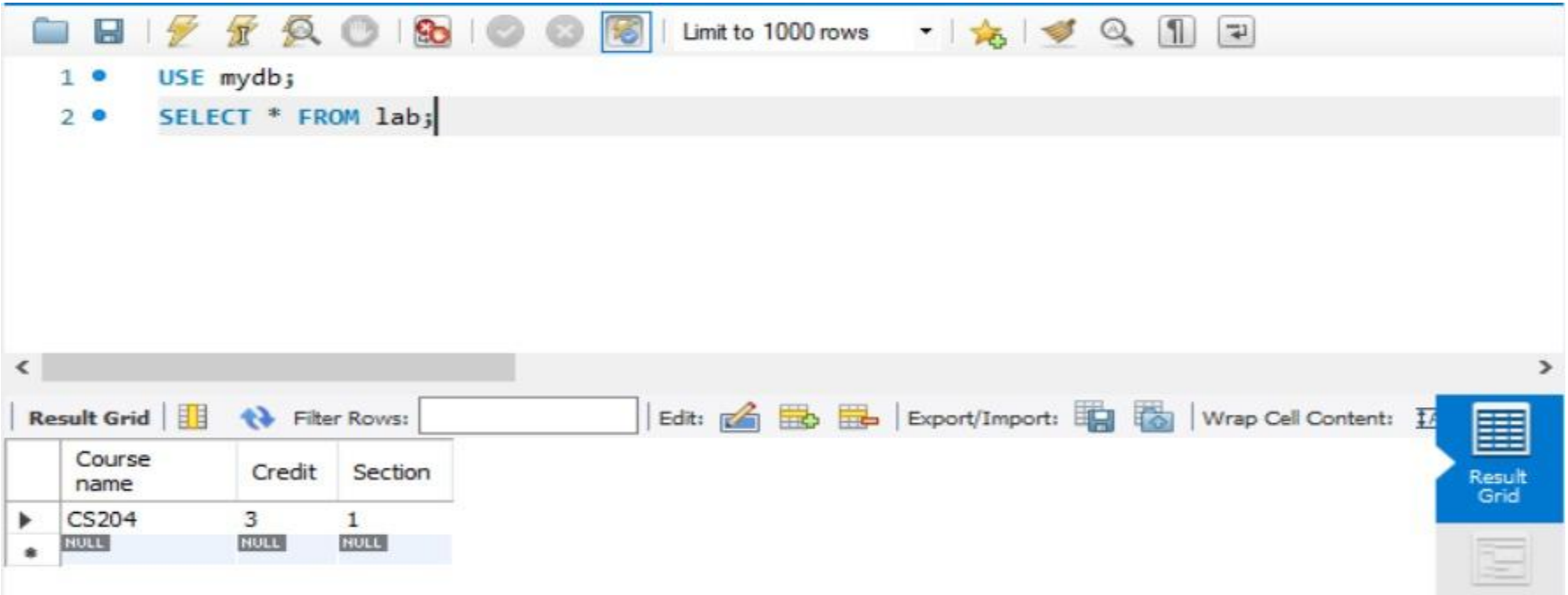
```sql
-- -----------------------------------------------------
-- Table `mydb`.`Subject`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `mydb`.`Subject` (
  `Course_Code` INT NOT NULL,
  `Semester` INT NOT NULL,
  `Branch` VARCHAR(10) NOT NULL,
  `Lab_Course name` VARCHAR(40) NOT NULL,
  `Theory_Course name` VARCHAR(40) NOT NULL,
  PRIMARY KEY (`Course_Code`, `Lab_Course name`, `Theory_Course name`),
  UNIQUE INDEX `Course_Code_UNIQUE` (`Course_Code` ASC) VISIBLE,
  INDEX `fk_Subject_Lab_idx` (`Lab_Course name` ASC) VISIBLE,
  INDEX `fk_Subject_Theory1_idx` (`Theory_Course name` ASC) VISIBLE,
  CONSTRAINT `fk_Subject_Lab`
    FOREIGN KEY (`Lab_Course name`)
    REFERENCES `mydb`.`Lab` (`Course name`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION,
  CONSTRAINT `fk_Subject_Theory1`
    FOREIGN KEY (`Theory_Course name`)
    REFERENCES `mydb`.`Theory` (`Course name`)
    ON DELETE NO ACTION
    ON UPDATE NO ACTION)
```

Insert the queries

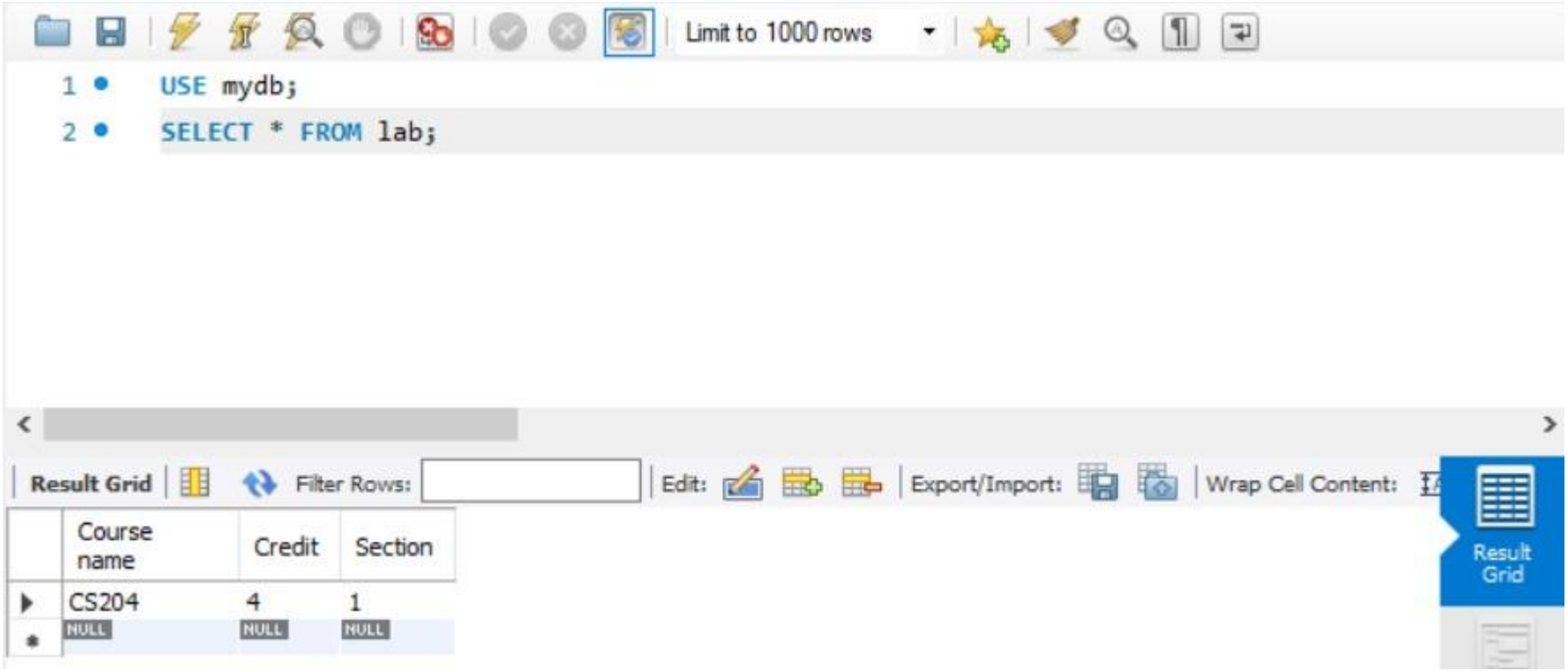
Select the queries



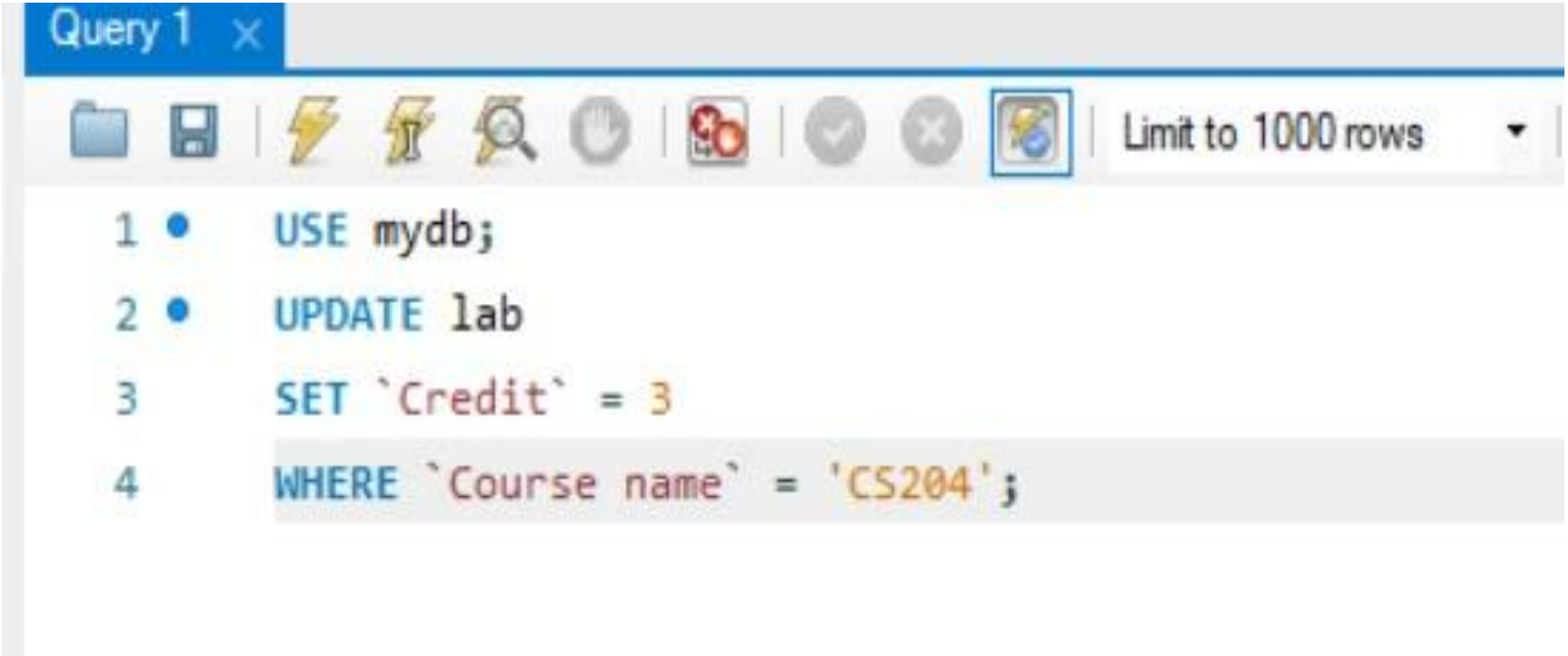
Select the queries



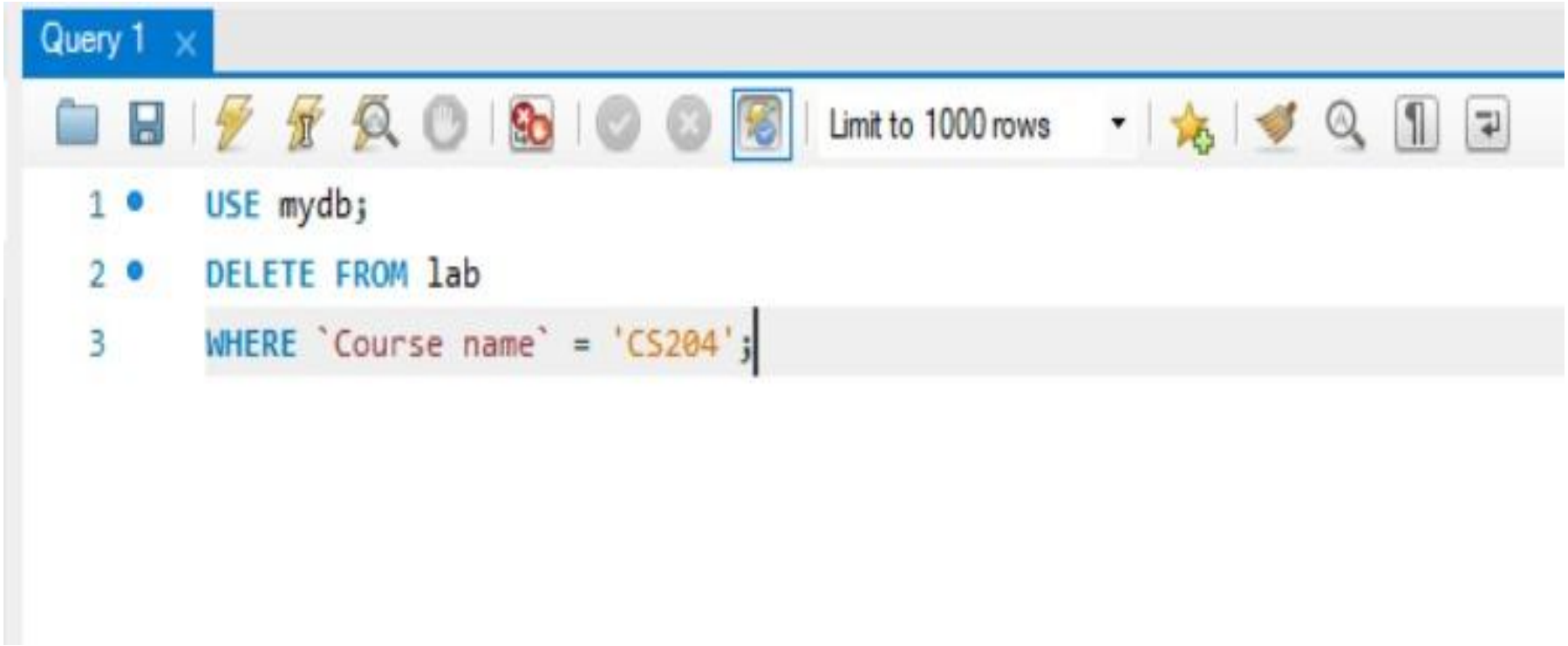
Select the queries



Update the queries



Delete the queries



Output

Action Output

#	Time	Action	Message
1	22:39:45	USE mydb	0 row(s) affected
2	22:39:45	INSERT INTO lab (`Course name`, `Credit`, `Section`) VALUES ('CS204', 4, 1)	1 row(s) affected
3	22:40:02	USE mydb	0 row(s) affected
4	22:40:02	INSERT INTO lab (`Course name`, `Credit`, `Section`) VALUES ('CS204', 4, 1)	Error Code: 1062. Duplicate entry 'CS204' for key 'lab.PRIMARY'
5	22:41:44	USE mydb	0 row(s) affected
6	22:41:44	SELECT * FROM lab LIMIT 0, 1000	1 row(s) returned
7	22:44:46	USE mydb	0 row(s) affected
8	22:44:46	UPDATE lab SET `Credit` = 3 WHERE `Course name` = 'CS204'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
9	22:45:30	USE mydb	0 row(s) affected
10	22:45:30	SELECT * FROM lab LIMIT 0, 1000	1 row(s) returned
11	22:46:29	USE mydb	0 row(s) affected
12	22:46:29	DELETE FROM lab WHERE `Course name` = 'CS204'	1 row(s) affected
13	22:46:57	USE mydb	0 row(s) affected