**crMySQL Labs**

**MySQL (Day1):**

|  |  |
| --- | --- |
|  | **Create a database called grades** |
|  | **CREATE database grades CHARACTER SET utf8 COLLATE utf8\_danish\_ci;**  **USE grades;** |
|  | **Create the following tables in the grades database:**  ***courses***  ***course\_id*** *int pk*  *course\_name varchar(100) not null*  *credit\_hour int*  ***students\_courses***  ***course\_id*** *int*  ***student\_id*** *int*  *grade int*  *reg\_date date*  ***students***  ***student\_id*** *int pk*  *student\_name varchar (100) not null*  *email varchar (50)*  *tel varchar (20)* |
|  | **CREATE TABLE IF NOT EXISTS students(**  **student\_id INT(11) NOT NULL AUTO\_INCREMENT,**  **student\_name VARCHAR(100) NOT NULL,**  **email VARCHAR(50),**  **tel VARCHAR(20),**  **PRIMARY KEY (student\_id))**  **ENGINE = INNODB;**  **CREATE TABLE IF NOT EXISTS *courses*(**  **course\_id INT(11) NOT NULL AUTO\_INCREMENT,**  **course\_name VARCHAR(100) NOT NULL,**  **credit\_hour INT,**  **PRIMARY KEY (course\_id))**  **ENGINE = INNODB;**  **CREATE TABLE IF NOT EXISTS *students\_courses*(**  **student\_id INT(11) NOT NULL,**  **course\_id INT(11) NOT NULL,**  **grade INT(11),**  **PRIMARY KEY (student\_id, course\_id),**  **FOREIGN KEY (student\_id) REFERENCES students (student\_id),**  **FOREIGN KEY (course\_id) REFERENCES courses (course\_id))**  **ENGINE = INNODB;** |
| **3** | **Modify the students table to allow for longer Student names (150 char)**  **Confirm your modification.** |
|  | **ALTER TABLE students**  **MODIFY student\_name VARCHAR(150) NOT NULL;** |
| **4** | **Add constraint to force unique email for each student** |
|  | **ALTER TABLE students**  **MODIFY email VARCHAR(50) UNIQUE;** |
| **5** | **Get Time, Date, Current user, MySQL Version using prompt?** |
|  | **SELECT NOW();**  **SELECT CURRENT\_USER();**  **show variables like '%version%';** |
| **6** | **Add gender column for the students table. It holds two value (male or female)** |
|  | **Alter table students add gender enum ('male', 'female');** |
| **7** | **Add birth\_date column for the students table.** |
|  | **Alter table students add birth\_date date;** |
| **8** | **Drop the student\_name column and replace it with first name and last name.** |
|  | **ALTER TABLE students**  **DROP COLUMN student\_name;**  **ALTER TABLE students ADD first\_name VARCHAR(50) NOT NULL;**  **ALTER TABLE students ADD last\_name VARCHAR(50) NOT NULL;** |
| **9** | **Insert your friend’s data into the table students.** |
|  | **INSERT INTO students (student\_id, first\_name, last\_name, tel, email, gender, birth\_date)**  **VALUES**  **(1,”Ahmed”,”Aly”,NULL,NULL,”male”,”1991-10-01”),**  **(2,”Ahmed”,”Ibrahim”,NULL,NULL,”male”,”1991-09-01”),**  **(3,”Ahmed”,”Ossama”,NULL,NULL,”male”, “1992-10-01”),**  **(4,”Hoda”,”Khaled”,NULL,NULL,”female”, “1991-09-01”),**  **(5,”Mona”,”Khalil”,NULL,NULL,”female”, “1992-10-01”);** |
| **10** | **Create a new table (male\_students) based on students table and fill it with the data of male students** |
|  | **CREATE TABLE male\_students SELECT \* FROM students WHERE gender=”male”;** |

**Part II**

**Create another database “OS42”**

**Use OS42**

**Run Lab Script then answer the following**

**CREATE database OS42 CHARACTER SET utf8 COLLATE utf8\_danish\_ci;**

**USE OS42;**

|  |  |
| --- | --- |
| **1** | **Display all students’ information.** |
|  | **select \* from students;** |
| **2** | **Display male students only.** |
|  | **select \* from students**  **WHERE gender = “male”; // or WHERE gender LIKE “male”;** |
| **3** | **Display the number of female students.** |
|  | **select count(student\_id) from students**  **WHERE gender = “female”;** |
| **4** | **Display the students’ data for the students who are born before 1992-10-01.** |
|  | **select \* from students**  **WHERE birth\_date < “1992-10-01”;** |
| **5** | **Display male students who are born before 1991-10-01.** |
|  | **select \* from students**  **WHERE birth\_date < “1992-10-01”**  **AND gender LIKE “male”;** |
| **6** | **Display course\_id and their grades sorted by grades.** |
|  | **select course\_id, grade from students\_courses**  **ORDER BY grade;** |
| **7** | **Display students’ names that begin with A.** |
|  | **select CONCAT(first\_name, “ “, last\_name) from students**  **WHERE first\_name LIKE “A%”;** |
| **8** | **Display the gender, number of males and females.** |
|  | **Select gender, count(student\_id) from students**  **Group by gender;** |
| **9** | **Display the repeated first names and their counts if higher than 2.** |
|  | **Select first\_name, count(student\_id) from students**  **Group by first\_name**  **Having count(student\_id) > 2;** |
| **10** | **Display the subject with highest grade** |
|  | **Select c.course\_name, sum(e.grade) as sum1**  **From courses c, students\_courses e**  **Where c.course\_id = e.course\_id**  **Group by c.course\_name**  **Order by sum1 DESC limit 1;** |