# **BTD210- Lab 4**

We, Muskan, Priya, Monica, declare that the attached assignment is our own work in accordance with the Seneca Academic Policy. No part of this assignment has been copied manually or electronically from any other source (including web sites) **or distributed to other students.**

Please work in **groups** to complete this lab. This lab is worth 2% of the total course grade and will be evaluated through your written submission, as well as the lab demo. During the lab demo, group members are randomly selected to present the answers to each of the lab questions. Group members not present during the lab demo will lose the demo mark.

Please submit the following files through Blackboard. Only one person must submit for the team.

* Lab4.docx

1. Add this declaration on the top of your file.
2. Specify what each member has done towards the completion of this work:

|  |  |  |
| --- | --- | --- |
|  | Name | Task(s) |
| 1 | Muskan | answers |
| 2 | Priya | answers, submission, typing |
| 3 | Monica | answers |

# Part I: SQL

1. To have a database of all groups of students, we will be using two tables: STD (to save student information) and GRP (to save group information). We have identified the following attributes needed in each table:

Table: STD

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute name | Data type needs | Constraints | Contents |
| S\_NUM | 10 characters | Primary key | Student number |
| S\_FIRST | Up to 15 characters | Not NULL and not repeating | First name |
| S\_DOB | Date |  | Date of birth |
| S\_N | Small integer |  | Number of tasks |
| G\_NUM | 2 characters | Foreign key to GRP table | Group number |

Table: GRP

|  |  |  |  |
| --- | --- | --- | --- |
| Attribute name | Data type needs | Constraints | Contents |
| G\_NUM | 2 characters | Primary key | Group number |
| G\_NAME | Up to 10 characters | Not NULL and not repeating | Group name |
| S\_NUM | 10 characters |  | Student number of the main contact in the group |

Note that there is a 1:M *membership* relationship, and a 1:1 *main-contact* relationship between the two tables. Write the SQL commands to create the GRP table and then the STD table..

CREATE TABLE GRP (

G\_NUM CHAR(2) PRIMARY KEY,

G\_NAME VARCHAR(10) NOT NULL UNIQUE,

S\_NUM CHAR(10));

CREATE TABLE STD (

S\_NUM CHAR(10) PRIMARY KEY,

S\_FIRST CHAR(15) NOT NULL UNIQUE,

S\_DOB DATE

S\_N SMALLINT

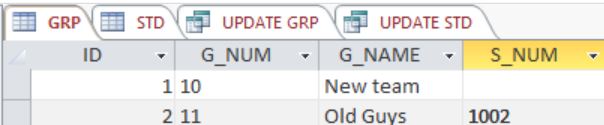
G\_NUM CHAR(2)

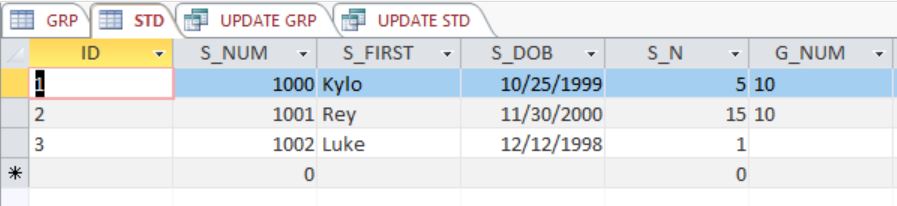
FOREIGN KEY (G\_NUM) REFERENCES GRP);

1. Write the SQL commands to insert the following data into the above tables (Oh well! My son is obsessed with Star Wars!):
   1. INSERT INTO STD VALUES  
      (‘1000’, ‘Kylo’, ‘25-OCT-1999’, ‘5’, NULL);  
      INSERT INTO STD VALUES  
      (‘1001’, ‘Rey’, ‘30-NOV-2000’, ‘15’, NULL);  
      INSERT INTO STD VALUES  
      (‘1002’, ‘Luke’, ‘12-DEC-1998’, ‘1’, NULL);
   2. INSERT INTO GRP VALUES  
      (‘10’, ‘New team’, NULL);  
      INSERT INTO GRP VALUES  
      (‘11’, ‘Old guys’, NULL);

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STD** | | | | |
| **S\_NUM** | **S\_FIRST** | **S\_DOB** | **S\_N** | **G\_NUM** |
| 1000 | Kylo | 10/25/1999 | 5 |  |
| 1001 | Rey | 11/30/2000 | 15 |  |
| 1002 | Luke | 12/12/1998 | 1 |  |

|  |  |  |
| --- | --- | --- |
| **GRP** | | |
| **G\_NUM** | **G\_NAME** | **S\_NUM** |
| 10 | New team |  |
| 11 | Old Guys |  |

1. Write a SQL UPDATE statement to set Luke (with S\_NUM of 1002) as the main contact of ‘Old Guys’.
   1. UPDATE GRP  
      SET S\_NUM = [‘1002’]   
      WHERE G\_NUM = ‘11’;
2. Write SQL UPDATE statements to assign Kylo and Rey to group 1.
   1. UPDATE STD  
      SET G\_NUM = 10  
      WHERE S\_FIRST IN (‘Kylo’, ‘Rey’);
3. Open Microsoft Access and start a new database. Use Create > Query Design > SQL View to copy your SQL commands (one by one) from Question 3 to 6 and run them. Check the spreadsheet view of the tables to check your work. Paste a snapshot here.  
   



# Part II: Joins

The following questions are NOT SQL questions. Show the resulting table after the updates of Question 5 and 6.

1. Find the product of the two tables above.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **STD S\_NUM** | **S\_FIRST** | **S\_DOB** | **S\_N** | **STD G\_NUM** | **GRP G\_NUM** | **G\_NAME** | **GRP S\_NUM** |
| 1000 | Kylo | 25-OCT-1999 | 5 | 10 | 10 | New team | NULL |
| 1000 | Kylo | 25-OCT-1999 | 5 | 10 | 11 | Old Guys | 1002 |
| 1001 | Rey | 30-NOV-2000 | 15 | 10 | 10 | New team | NULL |
| 1001 | Rey | 30-NOV-2000 | 15 | 10 | 11 | Old guys | 1002 |
| 1002 | Luke | 12-DEC-1998 | 1 | NULL | 10 | New team | NULL |
| 1002 | Luke | 12-DEC-1998 | 1 | NULL | 11 | Old Guys | 1002 |

1. Find the resulting table of
   1. The (inner) equijoin between the two tables based on the G\_NUM field. Which rows of the product in question 8 do you select?

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **STD S\_NUM** | **S\_FIRST** | **S\_DOB** | **S\_N** | **STD G\_NUM** | **GRP G\_NUM** | **G\_NAME** | **GRP S\_NUM** |
| 1000 | Kylo | 25-OCT-1999 | 5 | 10 | 10 | New team | NULL |
| 1001 | Rey | 30-NOV-2000 | 15 | 10 | 10 | New team | NULL |

We chose the STD G\_NUM and GRP G\_NUM for the first row and the third row because the values match

* 1. The (inner) equijoin between the two tables on the S\_NUM field. Which rows of the product in question 8 do you select?

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **STD S\_NUM** | **S\_FIRST** | **S\_DOB** | **S\_N** | **STD G\_NUM** | **GRP G\_NUM** | **G\_NAME** | **GRP S\_NUM** |
| 1002 | Luke | 12-DEC-1998 | 1 | NULL | 11 | Old Guys | 1002 |

We chose GRP S\_NUM and STD S\_NUM both have matching values in the last row

1. Match the two results in question 9 to the following queries (which is answering which?):
   1. List all groups with their main contacts.
      1. Result b
   2. List all groups with their members.
      1. Result A
2. What is the result of ‘STD LEFT OUTER JOIN GRP?

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **S\_NUM** | **S\_FIRST** | **S\_DOB** | **S\_N** | **G\_NUM** | **GRP G\_NAME** | **GRP S\_NUM** |
| 1000 | Kylo | 10/25/1999 | 5 | 10 | New team | NULL |
| 1001 | Rey | 11/30/2000 | 15 | 10 | New team | NULL |
| 1002 | Luke | 12/12/1998 | 1 | NULL | NULL | NULL |

1. What is the results of ‘STD RIGHT OUTER JOIN GRP?

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **S\_NUM** | **S\_FIRST** | **S\_DOB** | **S\_N** | **G\_NUM** | **GRP G\_NAME** | **GRP S\_NUM** |
| 1000 | Kylo | 10/25/1999 | 5 | 10 | New team | NULL |
| 1001 | Rey | 11/30/2000 | 15 | 10 | New team | NULL |
| NULL | NULL | NULL | NULL | 11 | Old Guys | 1002 |

1. What is the results of ‘STD FULL OUTER JOIN GRP?

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **S\_NUM** | **S\_FIRST** | **S\_DOB** | **S\_N** | **G\_NUM** | **GRP G\_NAME** | **GRP S\_NUM** |
| 1000 | Kylo | 10/25/1999 | 5 | 10 | New team | NULL |
| 1001 | Rey | 11/30/2000 | 15 | 10 | New team | NULL |
| 1002 | Luke | 12/12/1998 | 1 | NULL | NULL | NULL |
| NULL | NULL | NULL | NULL | 11 | Old Guys | 1002 |