Homework 10

PROBLEM #5.1:

Consider a simplified university database that includes information on courses (name, number, day, time, room number, max enrollment) and on faculty teaching courses and students attending courses. Suggest a relational database for efficiently managing this information.

SOLUTION:

Database is composed of 3 tables, Course table, Faculty table, Student table. The Student and Faculty tables are linked by the course number field in the Course table.

Course Table

Course Name	Course Number	Day	Time	Room Number	Max Enrollment
	Primary Key				

Student Table

Student Name	Course Number
	Foreign Key

Faculty Table

Faculty Name	Course Number
	Foreign Key

PROBLEM #5.2:

The following table below provides information on members of a mountain climbing club.

Climber-ID	Name	Skill Level	Age
123	Edmund	Experienced	80
214	Arnold	Beginner	25
313	Bridget	Experienced	33
212	James	Medium	27

The primary key is *Climber-ID*. Explain whether or not each of the following rows can be added to the table.

Climber-ID	Name	Skill Level	Age
214	Abbot	Medium	40
	John	Experienced	19
15	Jeff	Medium	42

SOLUTION:

The first row in the second table cannot be added because there is already a row in the first table with that primary key. Primary keys need to be unique.

The second row in the second table cannot be added because primary keys cannot be null or empty.

The third row in the second table can be added because there is no other row that matches that primary key.

SOLUTION:
SQL statement to create tables within a database is:

```
CREATE TABLE <table name> (
<column name> <data type>(<size>),
<column name> <data type>(<size>),
...
<column name> <data type>(<size>));
```

Command used to create student table:

```
CREATE TABLE student (
student_ID INTEGER PRIMARY KEY,
firstName VARCHAR(20),
lastName VARCHAR(20),
phoneNumber CHAR(10));
```

Results in table:

```
mysql> describe student;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| student_ID  | int(11)     | NO   | PRI | NULL    |       |
| firstName   | varchar(20) | YES  |     | NULL    |       |
| lastName    | varchar(20) | YES  |     | NULL    |       |
| phoneNumber | char(10)    | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
4 rows in set (0.00 sec)
```

PROBLEM #5.5:

Consider an SQL statement:
```
SELECT
id, forename, surname
FROM
authors
WHERE
forename = 'john' AND surname = 'smith'
```

   a What is this statement intended to do?

   b Assume that the forename and surname fields are being gathered from user supplied input, and suppose the user responds with:
Forename: jo'hn
Surname: smith
What will be the effect?

   c Now suppose the user responds with:
Forename: jo';drop table authors–
Surname: smith
What will be the effect?

SOLUTION:

   a The statement is supposed to give a list of the IDs, forenames, and surnames of all rows in the authors table where the forename is john, and the surname is smith.

   b The apostrophe in the middle of jo'hn will end the query for the forename. Meaning that the user will only search jo in the forenames.

   c Assuming no sanitation of inputs, the apostrophe will end the forename section of the query, the semicolon will end the entire query, and the command `drop table authors--` will remove the authors table from the database, the double hyphens are to comment our the rest of the line after the injection attempt.

PROBLEM #5.8:

Assume that A, B, and C grant certain privileges on the employee table to X, who in turn grants them to Y, as shown in the following table, with the numerical entries indicating the time of granting:

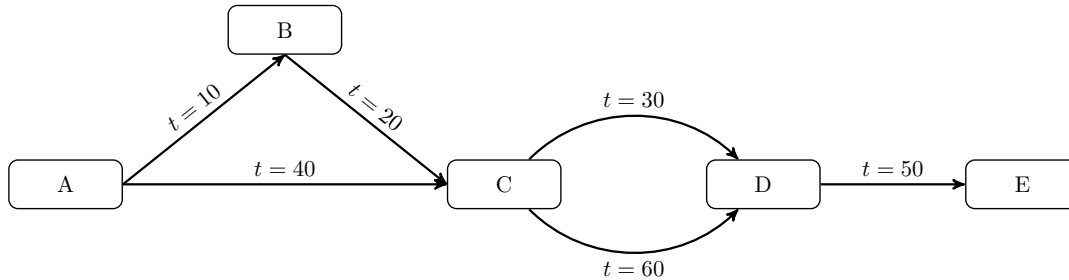| UserID	Table	Grantor	READ	INSERT	DELETE
X	Employee	A	15	15	–
X	Employee	B	20	–	20
Y	Employee	X	25	25	25
X	Employee	C	30	–	30

At time $t = 35$, B issues the command REVOKE ALL RIGHTS ON employee FROM X. Which access rights, if any, of Y must be revoked, using the conventions defined in Section 5.2?

SOLUTION:

Delete access of user Y should be revoked at $t = 35$ because B granted read and delete access to X, but A also gave read and insert access to X. After the revoke, X should have read and insert from A (at $t = 15$) and read and delete from C (at $t = 30$), X has read, insert, and delete. Y gets read, insert and delete from X (at $t = 25$) but X has delete privileges revoked at $t = 35$, Y ends with read and insert privileges from X.
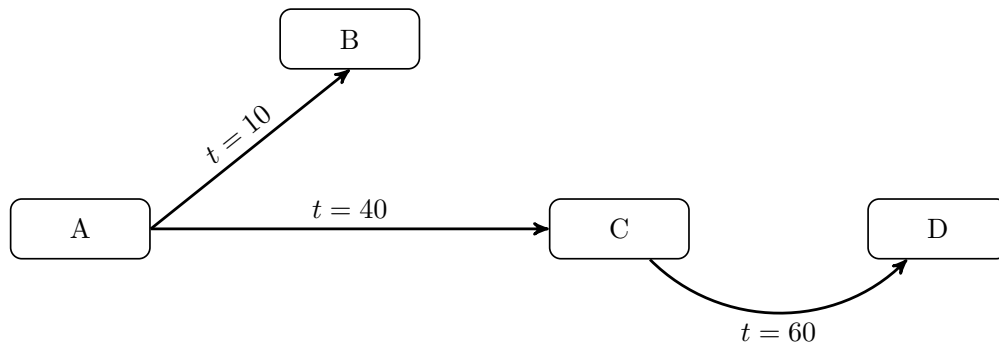
The figure below shows a sequence of grant operations for a specific access right on a table. Assume that at $t = 70$, B revokes the access right from C, using the conventions defined in Section 5.2, show the resulting diagram of access right decencies.

SOLUTION:

- A grants access to B at $t = 10$

- B grants access to C at $t = 20$

- C grants access to D at $t = 30$

- A grants access to C at $t = 40$

- D grants access to E at $t = 50$

- C grants access to D at $t = 60$

- B revokes access from C at $t = 70$

E will be removed from access because E was using privileges given to D from B. When B revoked privileges, it cascaded down and removed E. D stays because D was given access by C at $t = 60$.

PROBLEM #5.11:

Consider the parts department of a plumbing contractor. The department maintains an inventory database that includes parts information (part number, description, color, size, number in stock, etc.) and information on vendors from whom pans are obtained (name, address, pending purchase orders, closed purchase orders, etc.). In an RBAC system, suppose that roles are defined for accounts payable clerk, an installation foreman, and a receiving clerk. For each role, indicate which items should be accessible for read-only and read-write access.

SOLUTION:
Accounts Payable Clerk access rights:

- Read

- Write

Installation Foreman access rights:

- Read

Receiving Clerk access rights:

- Parts – Read

- Parts – Write

- Vendor – Read

PROBLEM #5.12:

Imagine that you are the database administrator for a military transportation system. You have a table named cargo in your database that contains information on the various cargo holds available on each outbound airplane. Each row in the table represents a single shipment and lists the contents of that shipment and the flight identification number. Only one shipment per hold is allowed. The flight identification number may be cross-referenced with other tables to determine the origin, destination, flight time, and similar data. The cargo table appears as follows:

Flight ID	Cargo Hold	Contents	Classification
1254	A	Boots	Unclassified
1254	B	Guns	Unclassified
1254	C	Atomic Bomb	Top Secret
1254	D	Butter	Unclassified

Suppose that two roles are defined: Role 1 has full access rights to the cargo table. Role 2 has access rights only to rows of the table in which the Classification field has the value Unclassified. Describe a scenario in which a user assigned to role 2 uses one or more queries to determine that there is a classified shipment on board the aircraft.

SOLUTION:

If a user with Role 2 tries to insert a item into cargo hold C (because they cannot see the classified shipment) they will get an error, in which case the user can deduce that there is a classified shipment in cargo hold C.

PROBLEM #5.13:

Users hulkhogan and undertaker do not have the SELECT access right to in Inventory table and the Item table. These tables were created by and are owned by user bruno-s. Write the SQL commands that would enable bruno-s to grant SELECT access to these tables to hulkhogan and undertaker.

SOLUTION:
The syntax for the grant command is:

```
GRANT
    priv_type [(column_list)]
      [, priv_type [(column_list)]] ...
    ON [object_type] priv_level
    TO user_or_role [, user_or_role] ...
    [WITH GRANT OPTION]
    [AS user
        [WITH ROLE
            DEFAULT
          | NONE
          | ALL
          | ALL EXCEPT role [, role ] ...
          | role [, role ] ...
        ]
    ]
}
```

So user bruno-s needs to execute the command:

```
GRANT select
ON Inventory
TO hulkhogan, undertaker
```