Marmara University – Faculty of Engineering – Department of Computer Engineering

Fall 2012 – CSE355 Database Systems

Quiz #3

(29.11.2012 | Duration: 50 min)

1) [30 pts] Fill in the blanks with the terms given. Use each term only once.
   atomic | attribute | bigint | cascade | cast | char | check | composite key | consistency | cursor | datetime | decimal | default | foreign key | identity | index | int | integrity | multi-valued | nchar | normalization | ntext | nullability | nvarchar | primary key | redundancy | referential | single-valued | text | unique | varchar

   a) **nullability** determines whether or not a value must be present in a column.
   b) **foreign key** is used to enforce relationships between tables.
   c) **identity** is a property typically associated with integer or bigint columns that provide automated generation of values during insert operations.
   d) While **varchar** represents variable length single-byte character data, **nvarchar** represents variable length unicode character data.
   e) **cast** is used to explicitly convert a value in one data type to another data type.
   f) **check** constraint limits the values that are accepted into a column.
   g) A table can have multiple **unique** constraints but only a single **primary key** constraint.
   h) **cascade** makes the required changes to the referencing tables when a user tries to update or delete a key column (or columns) to which a foreign key constraint references.

2) [25 pts] CREATE TABLE Pet {
   PetID int IDENTITY(1,1) PRIMARY KEY,
   PetName nvarchar(30) NOT NULL,
   DateOfBirth date NOT NULL,
   YearOfBirth AS DATEPART(year,DateOfBirth) PERSISTED);

A petstore database has a table “Pet” which is created by the statement given above. What is the result of the following statements? Give Pet table's final content. (Assume the table is initially empty)

insert into Pet (PetName, DateOfBirth) values ('Lassie', '20080101')
insert into Pet (PetName, DateOfBirth) values ('Boncuk', '20070707')
select * from Pet

<table>
<thead>
<tr>
<th>PetID</th>
<th>PetName</th>
<th>DateOfBirth</th>
<th>YearOfBirth</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Lassie</td>
<td>2008-01-01</td>
<td>2008</td>
</tr>
<tr>
<td>2</td>
<td>Boncuk</td>
<td>2007-07-07</td>
<td>2007</td>
</tr>
</tbody>
</table>

Next Page →
3) [45 pts] You have a relational database with the following tables and data. Assume you created your tables by the following statements:

CREATE TABLE Employee (  
    SSN smallint PRIMARY KEY,  
    DNO tinyint);  

CREATE TABLE Department (  
    DNO tinyint PRIMARY KEY,  
    MGRSSN smallint);  

CREATE TABLE Project (  
    PNO smallint PRIMARY KEY,  
    DNO tinyint);  

Then you modified your tables by executing the “ALTER TABLE” statements, and tried to delete some data from the Department table. (Consider each case a, b, and c separately) What is the result of the following statements? Give each table’s final contents?

a) ALTER table Employee add FOREIGN KEY (DNO) REFERENCES Department(DNO);  
   ALTER table Department add FOREIGN KEY (MGRSSN) REFERENCES Employee (SSN);  
   ALTER table Project add FOREIGN KEY (DNO) REFERENCES Department(DNO);  
   delete from Department where DNO=3

b) ALTER table Employee add FOREIGN KEY (DNO) REFERENCES Department(DNO) ON DELETE CASCADE;  
   ALTER table Department add FOREIGN KEY (MGRSSN) REFERENCES Employee (SSN);  
   ALTER table Project add FOREIGN KEY (DNO) REFERENCES Department(DNO) ON DELETE CASCADE;  
   delete from Department where DNO=1

c) ALTER table Employee add FOREIGN KEY (DNO) REFERENCES Department(DNO) ON DELETE CASCADE;  
   ALTER table Department add FOREIGN KEY (MGRSSN) REFERENCES Employee (SSN);  
   ALTER table Project add FOREIGN KEY (DNO) REFERENCES Department(DNO);  
   delete from Department where DNO=2  
   raises an error, no change in the tables