Department of Computer Science & Engineering

B.Tech (CSE) 4th Sem – (1 &2) (Batch 2020-2024)

RDBMS
ACCS – 16405

Assignment -2

Total Marks :24

Student Name : Sumit Kumar Giri

Class CSE Sem 4th Univ RollNo. 2000213

Section – A (6 Ques * 2 Marks=12 Marks)

Q1. Explain constraints in SQL along with their types. Give examples also. [CO4]

Ans:- A Constraint can constrain a single column or a group of column in a table. The more constraints you add to a table definition, the less work you have to do in applications to maintain the data integrity.

The different kinds of constraints are:-

- NOT NULL Integrity constraint.
- primary key constraint.
- foreign key constraint.
- Unique key Integrity constraint.
- check Integrity constraint.
- Default constraint.

Example:-

```
create table Student
(
  Studentid number NOT NULL,
  Student_firstname varchar(20),
  Student_lastname varchar(30),
  Student_phoneno. number
);
```

SQL> desc Student;

Name	Null?	TYPE
Studentid	NOT NULL	NUMBER
Student_first name		VARCHAR2(20)
student_lastname		VARCHAR2(20)
Student_Phoneno		NUMBER

Q2. Explain the various operations involved with Transactions. [CO5]

Q3. Explain the "Grant" and "Revoke" commands in SQL. Write syntax and examples.[CO4]

Ans:- Grant:- SQL Grant command is specifically used to provide Privileges to database objects for a user. This command also allow users to grant permissions to other users too.

Syntax:-
grant privilege_name on object_name
to {user_name | public | role_name}

Here privileges_name is which permission has to be granted, object_name is the name of the database object.

example:- Grant create session to Abcd;
Grant succeeded.

SQL> Alter user abcd
default tablespace users quota 5m on users;
user altered.

SQL> connect abcd/abcd;
Connected.

Revoke:- Revoke command withdraw user privileges on database objects if any granted. It does operations opposite the Grant command.

Syntax:- revoke privilege_name on object_name
from {user_name | public | role_name}

Example:- Revoke select on Emp from abcd;
Revoke succeeded.

Q4. Explain aggregate functions in SQL. Give examples [CO1]

Ans:- The Aggregate functions act on group of rows to give a result per group or rows rather on single row. The arrgregate function returns no rows or only rows with Nulls for the agregate to the aggregate function.

There are many type of aggregate functions.

1. COUNT.
2. SUM.
3. AVG.
4. MIN and MAX.
5. VARIANCE(n).

Example:-

```
SQL> Select Count(*), Count(Comm), from Emp;

  Count(*)    Count(Comm)
 ----------  -------------
     14            4
```

```
SQL> select sum(comm), Count(comm), Avg(comm)
     from emp;

  Sum(comm)   Count(comm)   Avg(comm)
 ----------  ------------  ----------
     2200          4           550
```

Q6. Differentiate between 3NF and BCNF. [C03]

Ans:- There are many different between 3NF and BCNF.

	3NF	BCNF
1.	IN 3NF there should be no Transitive dependency that is no non-prime attribute should be transitively should be dependent on the Candidate Key.	IN BCNF for any relation A→B, A should be a super key of relation.
2.	It is less Stronger then BCNF.	It is Comparatively more stronger than 3NF.
3.	IN 3NF the functional dependencies are already in 1NF and 2NF.	IN BCNF the functional dependencies are already in 1 NF, 2NF and 3NF.
4.	The redundancy is high in 3NF.	The redundancy is comparatively low in BCNF.
6.	IN 3NF there is preservation of all functional dependencies.	IN BCNF there may or may not be preservation of all functional depencies.

(3 Ques * 4 ...

Section – B

Q1. Explain ER Diagram along with its various symbols. Take suitable examples. [CO2]

Ans:- The structure of the database employing the E-R model is shown pictorially using entity relationship (E-R) diagram. The various entities and the relationship between them are shown with the help of following conventions.

- Entity is an instance of an entity type so it represented by a rectangle enclosing its name.
- Diamonds represents relationship. The name of the relationship is enclosed in Diamond.
- Ellipses represents attributes.
- Lines links attributs to entity and entity to the relationship.

Symbol	meaning	Symbol	meaning
E	Entity	A	Derived Attribute
E	Weak Entity		Composite Attribute
R	Relationship	R E	Total participation
R	Relationship for weak Entity	A	Primary Key
A	Attribute	E_1 1 R N E_2	Cardinality Ratio 1:1 for E_1:E_2 in R
A	multivalued Attribute	E_1 M R N E_2	Cardinality Ratio M:N for E_1:E_2 in R

A Complete E-R diagram should include:

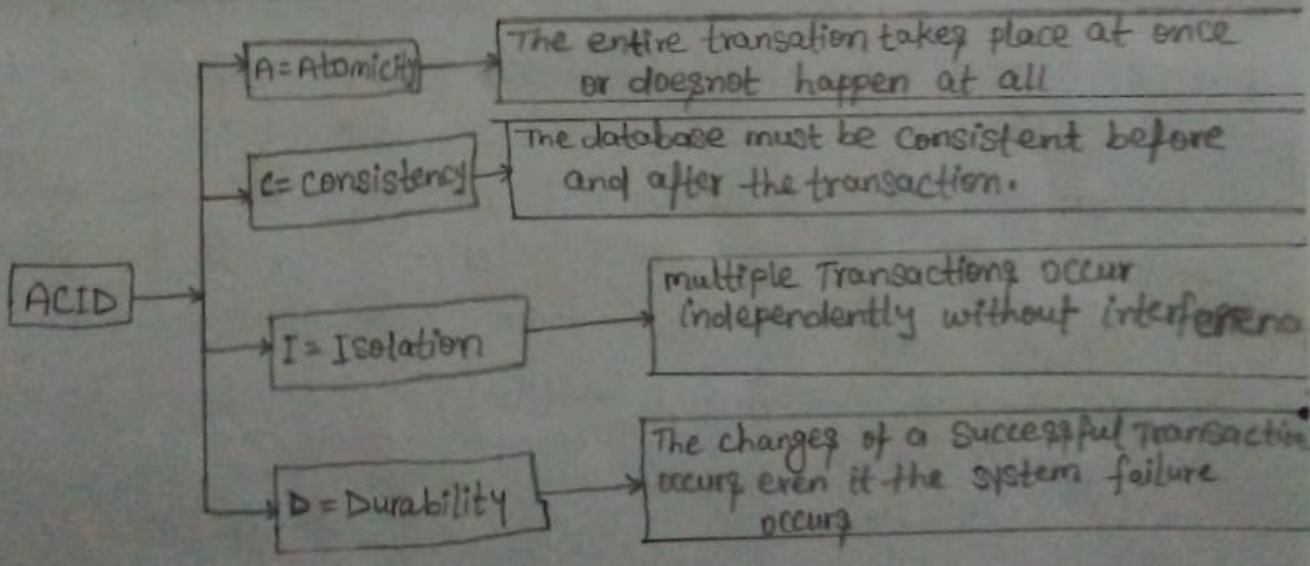
1. one or more entities (entity sets). Each entity only appears one per diagram.
2. one or more relationships. Each relationship only appears once per diagram.

3. Indications for all existing optionally conditions.
4. Cardinalities for all relationships.
5. All composite attributes should be expanded.

Q2. Explain the ACID properties for a transaction. [CO6]

Ans:- A transaction is a single logical unit of work which accessess and possibly modifies the contents of a database. Transaction access data using read and write operations. The ACID properties are meant for the transaction that goes through a different group of tasks, and there we come to see the role of the ACID properties.

ACID properties in DBMS



1. Atomicity:- The term atomicity defines that the data remains atomic. It means if any operation is performed on the data, either it should be performed on the data.
2. Consistency:- The word consistency means that the value should remain preserved always. In DBMS, the Integrity of data should be maintained, which means if a change in the database is made

3. Isolation :- The term 'isolation' means separation. In DBMS, Isolation is the property of a database where no data should affect the other one and may occur concurrently.

4. Durability :- Durability ensures the permanency of something. In DBMS, the term durability ensures that the data after the successful execution of the operation becomes permanent in the database.

Q3. Explain the concept of normalization in detail. Explain the normal forms.[CO3]

Ans:- Normalization is a designing technique that is widely as a guide in designing Relational database. It is process of decomposing (splitting) the Relations into Relations with fewer attributes by minimizing the redundancy of data and minimizing insertion, deletion and updation anomalies. The Relations with fewer attributes possess all desirable properties.

we normalize the Relational database management system because of the following reasons:-

- Minimize data redundancy.
- To make database structure flexible.
- Complex queries required by the user should be easy to handle.
- on decomposition of a Relation into smaller Relations with fewer attributes on normalization.

The normal forms are:-

(1). First normal form (1NF):- A Relation is said to be in first normal form if and only if it follow the rules:-

- All the primary key attributes are defined.
- There are no repeating groups in the table.
- All attributes are dependent on the primary key.

(2). Second Normal form (2NF) :- A Relation is said to be Second Normal form (2NF) if

- The Relation is in First Normal form (1NF).
- Every Non-key attributes should be fully functionally dependent on the primary key.

(3) Third Normal form (3NF) :- A Relations is said to be Third Normal form (3NF) if

- The Relation is in Second Normal form (2NF).
- Non-key attribute of the Relation should not be transitively functionally dependent on the primary key.

(4) Boyce-Code Normal form (BCNF) :- A BCNF is Stronger definition of BNF. Unlike the 3NF where a Relational table consists of only one Candidate key. the BCNF deals with Relational tables that have—

(a) multiple Candidate keys.

(b) Composite Candidate keys.

(c) Candidate keys that are overlapped.