***Q=1: List out TCL and DCL commands.***

***(2 Marks)***

* ***DCL***

1. *GRANT*
2. *REVOKE*

* ***TCL***

1. *COMMIT*
2. *ROLLBACK*
3. *SAVEPOINT*
4. *SET TRANSACTION*

***Q=2: Differentiate TCL and DCL. (3 Marks)***

|  |  |
| --- | --- |
| ***DCL*** | ***TCL*** |
| * *DCL means data control language command.* | * *TCL means transactional control language command.* |
| * *We will use two commands which is describe below:*  1. *GRANT* 2. *REVOKE* | * *We will use three command which is describe below:*  1. *COMMIT* 2. *ROLLBACK* 3. *SAVEPOINT TRANSECTION* |
| * *It is used to create roles, permissions, and referential integrity as well it is used to control access to database by securing it.* | * *It is used to manage different transactions occurring within a database.* |
| * *A Data Control Language is a syntax similar to a computer programming language used to control access to data stored in a database* | * *Its is used to manage transactions in the database. These are used to manage the changes made by DML-statements. It also allows statements to be grouped together into logical transactions.* |

**Q=3: Explain GRANT command by example.**

***(3 Marks)***

* ***Definition:*** 
  + *SQL Grant command is used to provide access or privileges on the database objects to the users.*
* ***Syntax:***
* *grant <privilege record>  
  on <relation title or view title>  
  to <user/role record>;*

### ***Allow a User to create session***

* *When we create a user in SQL, it is not even allowed to login and create a session until and unless proper permissions/privileges are granted to the user.*
* *Following command can be used to grant the session creating privileges.*

GRANT CREATE SESSION TO username;

### ***Allow a User to create table***

* *To allow a user to create tables in the database, we can use the below command,*

GRANT CREATE TABLE TO username;

* ***Example:***
* *As a user2, grant select and insert privileges to user3.*
* ***Input:***

*GRANT SELECT, INSERT*

*ON user1.Customer*

*TO user3;*

* ***Output:***

*Grant Successed.*

***Q=4: Explain REVOKE Command by Example.***

***(3 Marks)***

* *Use the REVOKE statement to remove privileges from a specific user or role, or from all users, to perform actions on database objects. You can also use the REVOKE statement to revoke a role from a user, from PUBLIC, or from another role.*

### ***To take back Permissions***

* *And, if you want to take back the privileges from any user, use the REVOKE command.*

**REVOKE CREATE TABLE FROM username;**

* ***For Example:***

1. *REVOKE SELECT ON employee FROM user1 This command will revoke a SELECT privilege on employee table from user1.*
2. *As a user2, revoke the select and insert privileges from user3.*

***Input:***

*REVOKE SELECT, INSERT*

*ON user1.Customer*

*FROM user3;*

***Output:***

*Revoke Successed.*

***Q=5: Explain COMMIT Command.***

***(3 Marks)***

* *The COMMIT command is the transactional command used to save changes invoked by a transaction to the database.*
* *The COMMIT command is the transactional command used to save changes invoked by a transaction to the database. The COMMIT command saves all the transactions to the database since the last COMMIT or ROLLBACK command.*
* *The syntax for the COMMIT command is as follows.*

***COMMIT;***

* ***Example:***
* *Consider the CUSTOMERS table having the following records –*

***+----+----------+-------+-----------+----------+-----------+***

***| ID | NAME | AGE | ADDRESS | SALARY |***

***+----+----------+--------+-----------+----------+----------+***

***| 1 | Ramesh | 32 | Ahmedabad | 2000.00 |***

***| 2 | Khilan | 25 | Delhi | 1500.00 |***

***| 3 | kaushik | 23 | Kota | 2000.00 |***

***| 4 | Chaitali | 25 | Mumbai | 6500.00 |***

***| 5 | Hardik | 27 | Bhopal | 8500.00 |***

***| 6 | Komal | 22 | MP | 4500.00 |***

***| 7 | Muffy | 24 | Indore | 10000.00 |***

***+-------+----------+--------+-----------+----------+-------+***

* *Following is an example which would delete those records from the table which have age = 25 and then COMMIT the changes in the database.*

***Input:***

*SQL> DELETE FROM CUSTOMERS*

*WHERE AGE = 25;*

*SQL> COMMIT;*

* *Thus, two rows from the table would be deleted and the SELECT statement would produce the following result.*
* ***Output:***

***+----+----------+-----+-----------+----------+***

***| ID | NAME | AGE | ADDRESS | SALARY |***

***+----+----------+-----+-----------+----------+***

***| 1 | Ramesh | 32 | Ahmedabad | 2000.00 |***

***| 3 | Kaushik | 23 | Kota | 2000.00 |***

***| 5 | Hardik | 27 | Bhopal | 8500.00 |***

***| 6 | Komal | 22 | MP | 4500.00 |***

***| 7 | Muffy | 24 | Indore | 10000.00 |***

***+----+----------+-----+-----------+----------+***

***Q=6: Explain Rollback and Save point Command.***

***(3 Marks)***

* *The ROLLBACK command is the transactional command used to undo transactions that have not already been saved to the database. This command can only be used to undo transactions since the last COMMIT or ROLLBACK command was issued.*
* *The syntax for a ROLLBACK command is as follows −*

***ROLLBACK;***

* ***Example:***
* *Consider the CUSTOMERS table having the following records –*

***+----+----------+-----+-----------+----------+***

***| ID | NAME | AGE | ADDRESS | SALARY |***

***+----+----------+-----+-----------+----------+***

***| 1 | Ramesh | 32 | Ahmedabad | 2000.00 |***

***| 2 | Khilan | 25 | Delhi | 1500.00 |***

***| 3 | Kaushik | 23 | Kota | 2000.00 |***

***| 4 | Chaitali | 25 | Mumbai | 6500.00 |***

***| 5 | Hardik | 27 | Bhopal | 8500.00 |***

***| 6 | Komal | 22 | MP | 4500.00 |***

***| 7 | Muffy | 24 | Indore | 10000.00 |***

***+----+----------+-----+-----------+----------+***

* *Following is an example, which would delete those records from the table which have the age = 25 and then ROLLBACK the changes in the database.*

***Input:***

*SQL> DELETE FROM CUSTOMERS*

*WHERE AGE = 25;*

*SQL> ROLLBACK;*

***+----+----------+-----+-----------+----------+***

***| ID | NAME | AGE | ADDRESS | SALARY |***

***+----+----------+-----+-----------+----------+***

***| 1 | Ramesh | 32 | Ahmedabad | 2000.00 |***

***| 2 | Khilan | 25 | Delhi | 1500.00 |***

***| 3 | Kaushik | 23 | Kota | 2000.00 |***

***| 4 | Chaitali | 25 | Mumbai | 6500.00 |***

***| 5 | Hardik | 27 | Bhopal | 8500.00 |***

***| 6 | Komal | 22 | MP | 4500.00 |***

***| 7 | Muffy | 24 | Indore | 10000.00 |***

***+----+----------+-----+-----------+----------+***

### ***The SAVEPOINT Command***

* *A SAVEPOINT is a point in a transaction when you can roll the transaction back to a certain point without rolling back the entire transaction.*
* *The syntax for a SAVEPOINT command is as shown below.*

***SAVEPOINT SAVEPOINT\_NAME;***

* *This command serves only in the creation of a SAVEPOINT among all the transactional statements. The ROLLBACK command is used to undo a group of transactions.*
* *The syntax for rolling back to a SAVEPOINT is as shown below.*

***ROLLBACK TO SAVEPOINT\_NAME;***

* *Following is an example where you plan to delete the three different records from the CUSTOMERS table. You want to create a SAVEPOINT before each delete, so that you can ROLLBACK to any SAVEPOINT at any time to return the appropriate data to its original state.*
* ***Example:***

1. *SAVEPOINT Demo;*
2. *ROLLBACK TO Demo;*

### ***The RELEASE SAVEPOINT Command.***

* *Once a SAVEPOINT has been released, you can no longer use the ROLLBACK command to undo transactions performed since the last SAVEPOINT.*
* *The syntax for a RELEASE SAVEPOINT command is as follows.*

***RELEASE SAVEPOINT SAVEPOINT\_NAME;***

* ***Example:***

1. *RELEASE SAVEPOINT Demo;*

***Q=7: Consider two users named “user1” and “user2”. “user1” is the owner of “employee” table which have following structure.***

***Employee (empno , ename , salary, deptno)***

***Write SQL queries for following:***

***1) user1 gives permissions for insert, update and select privileges to user2 with further grant option.***

***2) user1 take away the permissions for insert and select privileges from user2.***

*Grant insert, update, select user1.Employee ON user2.*

*Grant insert, select user1.Employee ON user2.*

***Q=14: Explain deadlock detection and prevention. (3 Marks)***

* ***Deadlock detection:***
* *It allows deadlock to occur, but then it detects it and solve it.*
* *Here, database is periodically checked for deadlock.*
* *If deadlock is detected one of the transaction in cycle is aborted,*
* *other transaction continues.*
* *Deadlock prevention:*
* *It prevents a deadlock to occur.*
* *It requires that all transactions lock all data item they need in*
* *advance.*
* *It transaction acquires all required locks, a transaction can run to*
* *completion to commit.*
* *But, if any of locks cannot be acquired, a transaction will be aborted,*
* *locks are released, transaction rescheduled.*
* *It is used in Two-phase locking.*

***Q=15: What is deadlock? how to control deadlock. (4 Marks)***

* ***Definition:***
* *A set of transaction is deadlock, if each transaction in the set is*
* *waiting for a lock held by some other transaction in the set.*
* *A process with two or more threads can deadlock when the*
* *following conditions hold:*
  + - * *Transactions that are already holding locks request new locks.*
      * *The requests for new locks are mode concurrently.*
      * *Two or more transaction form a circular chain in which is*
* *held by the next transaction in the chain.*
* *How to control deadlock:*
* *In a database, a deadlock is an unwanted situation in which*
* *two or more transactions are waiting indefinitely for one*
* *another to give up locks.*
* *As a consequence, all activity comes to a halt and remains at a*
* *standstill forever unless the DBMS detects the deadlock and*
* *aborts one of the transactions.*
* *Tips on avoiding deadlocks:*
* *Ensure the database design is properly normalized.*
* *Develop applications to access server objects in the same order*
* *each time.*
* *Do not allow any user input during transactions.*
* *Avoid cursor.*
* *Keep transactions as short as possible.*

***Q=16: what is synonyms? Write syntax for creating &amp; destroying synonyms. (3 Marks)***

* *Synonym is an alternative name for database objects like tables, views, indexes, sequences.*
* *It can be used to hide the actual name of entity from granted user means do not sharing the original name to another user.*
* *It can also be used to make a nick name or a short name of any entity.*
* *Syntax to create synonyms: -*

*CREATE SYNONYM synonym\_name FOR object\_name;*

* ***Syntax to destroy synonyms: -***

*DROP SYNONYM synonym\_name;*

* ***Example: -***

*CREATE SYNONYM stu FOR system. student100;*

* ***OUTPUT: -***

*synonym created.*

***Q=17: what is sequence? Give significance of it. (2 Marks)***

* *Sequence is an automatic counter which generates sequential numbers whenever required.*
* *It can be used to insert and update value.*
* *It can be in ascending or in descending order at defined intervals and can be also restart.*

***Q.18 How sequence can be created explain by example. (4 Marks)***

* *Syntax to create synonym: -*

*CREATE SEQUENCE sequence\_name [STARRT WITH n INCREMENT BY n MINVALUE n / NOMINVALUE MAXVALUE n / NOMAXVALUE CYCLE n / NOCYCLE CACHE n / NOCACHE ORDER n / NO ORDER];*

* *we can also create sequence simply by CREATE SEQUENCE sequence\_name; it will start with 1 and increment also by 1.*
* ***Example:***
* ***To create sequence:***

*Create SEQUENCE seq START WITH 1 INCREMENT BY 1 MINVALUE 1 MAXVALUE 99 NOCYCLE;*

* ***OUTPUT:***

*sequence created.*

* ***Example:***

*To generate values using sequence : SELECT (&#39;A &#39; || LTRIM( TO\_CHAR(seq.NEXTVAL,&#39; 00&#39; ),&#39; &#39; )) ANO FROM DUAL;*

* ***OUTPUT:***

|  |
| --- |
| *ANO*  *A01* |

***Q=19: Consider following table: student (Rollno, name, address) Write SQL command that insert records in above table using sequence for Rollno field. (4 Marks)***

* ***Create sequence: -***

*Create sequence seq;*

* ***Output:***

*sequence created.*

* ***Insert record in table using sequence: -***

*Insert into student values (seq.nextval,&#39;abc&#39;,&#39;pbr&#39;);*

* ***Output:***

*record inserted.*