

### Exercise Number: 1

**Title of the Exercise : DATA DEFINITION LANGUAGE (DDL) COMMANDS**

**Date of the Exercise :**

#### **OBJECTIVE (AIM) OF THE EXPERIMENT**

To practice and implement data definition language commands and constraints.

#### **FACILITIES REQUIRED AND PROCEDURE**

a) Facilities required to do the experiment:

Sl.No.	Facilities required	Quantity
1	System	1
2	Operating System	Windows
3	Front end	
4	Back end	Oracle11g

b) Procedure for doing the experiment:

Step no.	Details of the step
1	<b>DDL COMMAND</b> It is used to communicate with database. DDL is used to: o Create an object o Alter the structure of an object o To drop the object created.
2	The commands used are: Create, Alter, Drop, Truncate
3	<b>INTEGRITY CONSTRAINT</b> An integrity constraint is a mechanism used by oracle to prevent invalid data entry into the table. It has enforcing the rules for the columns in a table. The types of the integrity constraints are: a) Domain Integrity b) Entity Integrity c) Referential Integrity
4	<b>a) Domain Integrity</b> This constraint sets a range and any violations that take place will prevent the user from performing the manipulation that caused the breach. It includes: <b>Not Null constraint:</b> While creating tables, by default the rows can have null value .the enforcement of not null constraint in a table ensure that the table contains values. <b>Principle of null values:</b> o Setting null value is appropriate when the actual value is unknown, or when a value would not be meaningful. o A null value is not equivalent to a value of zero. o A null value will always evaluate to null in any expression. o When a column name is defined as not null, that column becomes a mandatory i.e., the user has to enter data into it. o Not null Integrity constraint cannot be defined using the alter table command when the table contain rows.
5	<b>Check Constraint:</b> Check constraint can be defined to allow only a particular range of values .when the manipulation violates this constraint, the record will be rejected. Check condition cannot contain sub queries.

6	<p><b>b) Entity Integrity</b> Maintains uniqueness in a record. An entity represents a table and each row of a table represents an instance of that entity. To identify each row in a table uniquely we need to use this constraint. There are 2 entity constraints:</p> <p><b>Unique key constraint</b> It is used to ensure that information in the column for each record is unique, as with telephone or drivers license numbers. It prevents the duplication of value with rows of a specified column in a set of column. A column defined with the constraint can allow null value.</p> <p>If unique key constraint is defined in more than one column i.e., combination of column cannot be specified. Maximum combination of columns that a composite unique key can contain is 16.</p> <p><b>Primary Key Constraint</b> A primary key avoids duplication of rows and does not allow null values. It can be defined on one or more columns in a table and is used to uniquely identify each row in a table. These values should never be changed and should never be null. A table should have only one primary key. If a primary key constraint is assigned to more than one column or combination of column is said to be composite primary key, which can contain 16 columns.</p>
7	<p><b>c) Referential Integrity</b> It enforces relationship between tables. To establish parent-child relationship between 2 tables having a common column definition, we make use of this constraint. To implement this, we should define the column in the parent table as primary key and same column in the child table as foreign key referring to the corresponding parent entry.</p> <p><b>Foreign key</b> A column or combination of column included in the definition of referential integrity, which would refer to a referenced key.</p> <p><b>Referenced key</b> It is a unique or primary key upon which is defined on a column belonging to the parent table.</p>

c) **SQL Commands:**

**CREATE TABLE**

It is used to create a table

**Syntax:** Create table tablename (column\_name1 data\_type constraints, column\_name2 data\_type constraints ...)

**Example:**

Create table Emp ( EmpNo number(5), EName VarChar(15), Job Char(10) constraint un unique, DeptNo number(3) **CONSTRAINT FKey2 REFERENCES DEPT(DeptNo)**);

Create table stud (sname varchar2(20) not null, rollno number(10) not null,dob date not null);

**Rules:**

1. Oracle reserved words cannot be used.
3. Underscore, numerals, letters are allowed but not blank space.
3. Maximum length for the table name is 30 characters.
4. 2 different tables should not have same name.
5. We should specify a unique column name.
6. We should specify proper data type along with width.
7. We can include "not null" condition when needed. By default it is 'null'.

**ALTER TABLE**

Alter command is used to:

1. Add a new column.
3. Modify the existing column definition.

3. To include or drop integrity constraint.

**Syntax:** alter table tablename add/modify (attribute datatype(size));

**Example:**

1. Alter table emp add (phone\_no char (20));
2. Alter table emp modify(phone\_no number (10));
3. ALTER TABLE EMP ADD CONSTRAINT Pkey1 PRIMARY KEY (EmpNo);

**DROP TABLE**

It will delete the table structure provided the table should be empty.

**Example:** drop table prog20; Here prog20 is table name

**TRUNCATE TABLE**

If there is no further use of records stored in a table and the structure has to be retained then the records alone can be deleted.

**Syntax:** TRUNCATE TABLE <TABLE NAME>;

Example: Truncate table stud;

**DESC**

This is used to view the structure of the table.

**Example:** desc emp;

Name Null? Type

```
-----  
EmpNo NOT NULL number(5)  
ENAME VarChar(15)  
Job NOT NULL Char(10)  
DeptNo NOT NULL number(3)  
PHONE_NO number (10)
```

**DOMAIN INTEGRITY**

**Example:** Create table cust(custid number(6) not null, name char(10));

Alter table cust modify (name not null);

**CHECK CONSTRAINT**

**Example:** Create table student (regno number (6), mark number (3) constraint b check (mark >=0 and mark <=100));

Alter table student add constraint b2 check (length(regno)<=4));

**ENTITY INTEGRITY**

a) **Unique key constraint**

**Example:** Create table cust(custid number(6) constraint uni unique, name char(10));

Alter table cust add(constraint c unique(custid));

b) **Primary Key Constraint**

**Example:** Create table stud(regno number(6) constraint primary key, name char(20));

d) **Queries:**

**Q1. Create a table called EMP with the following structure.**

Name	Type
EMPNO	NUMBER(6)
ENAME	VARCHAR2(20)
JOB	VARCHAR2(10)
DEPTNO	NUMBER(3)
SAL	NUMBER(7,2)

Allow NULL for all columns except ename and job.

**Solution:**

1. Understand create table syntax.
2. Use the create table syntax to create the said tables.
3. Create primary key constraint for each table as understand from logical table structure.

**Ans:**

```
SQL> create table emp(empno number(6),ename varchar2(20)not null,job varchar2(10) not null,
deptno number(3),sal number(7,2));
Table created.
```

**Q2: Add a column experience to the emp table. experience numeric null allowed.**

Solution:

1. Learn alter table syntax.
2. Define the new column and its data type.
3. Use the alter table syntax.

**Ans:**

```
SQL> alter table emp add(experience number(2));
Table altered.
```

**Q3: Modify the column width of the job field of emp table.**

Solution:

1. Use the alter table syntax.
2. Modify the column width and its data type.

**Ans:**

```
SQL> alter table emp modify(job varchar2(12));
Table altered.
SQL> alter table emp modify(job varchar(13));
Table altered.
```

**Q4: Create dept table with the following structure.**

Name	Type
DEPTNO	NUMBER(2)
DNAME	VARCHAR2(10)
LOC	VARCHAR2(10)

**Deptno as the primarykey**

Solution:

1. Understand create table syntax.
2. Decide the name of the table.
3. Decide the name of each column and its data type.
4. Use the create table syntax to create the said tables.
5. Create primary key constraint for each table as understand from logical table structure.

**Ans:**

```
SQL> create table dept(deptno number(2) primary key,dname varchar2(10),loc
varchar2(10));
Table created.
```

**Q5: create the emp1 table with ename and empno, add constraints to check the empno value while entering (i.e) empno > 100.**

Solution:

1. Learn alter table syntax.
2. Define the new constraint [columns name type]
3. Use the alter table syntax for adding constraints.

**Ans:**

```
SQL> create table emp1(ename varchar2(10),empno number(6) constraint ch
check(empno>100));
Table created.
```

**Q6: drop a column experience to the emp table.**

Solution:

1. Learn alter table syntax. Use the alter table syntax to drop the column.

**Ans:**

```
SQL> alter table emp drop column experience;
Table altered.
```

**Q7: Truncate the emp table and drop the dept table**

Solution:

1. Learn drop, truncate table syntax.

**Ans:**

SQL> truncate table emp;

Table truncated.

SQL> drop table dept;

Table dropped.

**e) Result:**

Thus the data definition language commands was performed and implemented successfully

You may skip copying the questions and answers, as they are for Viva practice.

**QUESTIONS AND ANSWERS****1. Define the terms**

**DDL:**

Data base schema is specified by a set of definitions expressed by a special language called a data definition language.

**2. What are the categories of SQL command?**

SQL commands are divided in to the following categories:

Data Delimitation language

Data manipulation language

Data control language

Transaction Control Language

**3. What is integrity constraint?**

An integrity constraint is a mechanism used by oracle to prevent invalid data entry into the table. It has enforcing the rules for the columns in a table.

**4. List the types of constraint.**

a) Domain Integrity

b) Entity Integrity

c) Referential Integrity

**5. Primary Key Constraint**

A primary key avoids duplication of rows and does not allow null values. It can be defined on one or more columns in a table and is used to uniquely identify each row in a table. These values should never be changed and should never be null.

**6. Referential Integrity**

It enforces relationship between tables. To establish parent-child relationship between 2 tables having a common column definition, we make use of this constraint. To implement this, we should define the column in the parent table as primary key and same column in the child table as foreign key referring to the corresponding parent entry.