Practical – 1

(SQL-1)

DDL (Data Definition Language)

- It is set of SQL commands used to create, modify and delete database structure but not data.
- These commands are not used by general user, used by DBA or db designer or application developer.

E.g. CREATE, ALTER, DROP, TRUNCATE

DML (Data Manipulation Language)

- Allows changing data within the database.

E.g. INSERT, UPDATE, DELETE

DCL (Data Control Language)

- SQl statements that control access to data and to the database.

E.g. COMMIT, SAVEPOINT, ROLLBACK, GRANT / REVOKE

DQL (Data Query Language)

- Allows getting data from the database and imposing ordering upon it.

E.g. SELECT

DDL - Data Definition Language:

Command	Description
CREATE	Creates a new table, a view of a table, or other object in database
ALTER	Modifies an existing database object, such as a table.
DROP	Deletes an entire table, a view of a table or other object in the database.

DML - Data Manipulation Language:

Command	Description
INSERT	Creates a record
UPDATE	Modifies records
DELETE	Deletes records

DCL - Data Control Language:

Command	Description
GRANT	Gives a privilege to user
REVOKE	Takes back privileges granted from user

DQL - Data Query Language:

Command	Description
SELECT	Retrieves certain records from one or more tables

BASIC DATATYPES USED IN ORACLE

Datatype	Description	Max Size: Oracle 8	Max Size: PL/SQL
VARCHAR(size)/ Varchar2(size)	Variable length Alphanumeric data having maximum length *size* bytes. You must specify size	**4000** bytes minimum is 1	32767 bytes minimum is 1
NVARCHAR(size)	Variable length national character set string having maximum length *size* bytes. You must specify size	4000 bytes minimum is 1	
CHAR(size)	Fixed length character data of length size bytes. This should be used for fixed length data. Such as codes A100, B102... Varchar is also treat as CHAR(1) *char is faster than varchar	**2000** bytes Default and minimum size is 1 byte.	32767 bytes Default and minimum size is 1 byte.
NUMBER(p,s)	Number having precision p and scale s.	**The precision p can range from 1 to 38.** **The scale s can range from -84 to 127**	
DATE	It is used to represent date and time. Standard format is DD-MON-YY and HH:MM:SS (24 hours format)	from January 1, 4712 BC to December 31, **9999** AD.	
RAW(size)	Raw binary data of length size bytes. You must specify size for a RAW value. e.g. graphics and audio files	Maximum size is 255 **(2000**)bytes	32767 bytes
LONG	Character data of variable length (A bigger version the VARCHAR2 datatype) long data can be used to store arrays of binary data in ASCII	2GB	32760 bytes
BLOB	Is used to store large binary object in the database e.g. graphic images, satellite images, video clips	4GB	
CLOB	Is used to store character large binary object in the database	4GB	

CREATE TABLE

```
CREATE TABLE table_name(
  column1 datatype,
  column2 datatype,
  column3 datatype,
  .....
  columnN datatype,
  PRIMARY KEY( one or more columns )
  );
```

E.g.

```
 SQL> CREATE TABLE CUSTOMERS(
   ID   INT              NOT NULL,
   NAME VARCHAR (20)     NOT NULL,
   AGE  INT              NOT NULL,
   ADDRESS  CHAR (25) ,
   SALARY   DECIMAL (18, 2),
   PRIMARY KEY (ID)
);
```

```
SQL> DESC CUSTOMERS;
+---------+---------------+------+-----+---------+-------+
| Field   | Type          | Null | Key | Default | Extra |
+---------+---------------+------+-----+---------+-------+
| ID      | int(11)       | NO   | PRI |         |       |
| NAME    | varchar(20)   | NO   |     |         |       |
| AGE     | int(11)       | NO   |     |         |       |
| ADDRESS | char(25)      | YES  |     | NULL    |       |
| SALARY  | decimal(18,2) | YES  |     | NULL    |       |
+---------+--------------+------+-----+---------+-------
```

INSERT VALUES INTO TABLE

```
INSERT INTO TABLE_NAME (column1, column2, column3,...columnN)]
VALUES (value1, value2, value3,...valueN);
```

E.g. SQL>INSERT INTO CUSTOMERS (ID,NAME,AGE,ADDRESS,SALARY)
VALUES (1, 'Ramesh', 32, 'Ahmedabad', 2000.00);

OR

```
//ORDER SHOULD REMAIN SAME
INSERT INTO TABLE_NAME VALUES (value1,value2,value3,...valueN);
E.g.  SQL> INSERT INTO CUSTOMERS VALUES (7, 'Muffy', 24, 'Indore', 10000.00 );
```

OR

//Populate one table using another table

```
INSERT INTO first_table_name [(column1, column2, ... columnN)]
   SELECT column1, column2, ...columnN
   FROM second_table_name
   [WHERE condition];
```

FILTERING TABLE DATA (VIEWING DATA IN THE TABLE)

```
SELECT column1, column2, columnN FROM table_name;
E.g. SQL> SELECT ID, NAME, SALARY FROM CUSTOMERS;

//ALL ROWS AND ALL COLUMNS

SELECT * FROM table_name;
```

E.g. SQL> SELECT * FROM CUSTOMERS;

SELECTED ROWS AND ALL COLUMNS

```
SELECT * FROM table_name WHERE [condition]
E.g. SQL> SELECT * FROM CUSTOMERS WHERE SALARY > 2000;
```

SELECTED ROWS AND SELECTED COLUMNS

```
SELECT column1, column2, columnN
FROM table_name
WHERE [condition]
```

E.g.

```
SQL> SELECT ID, NAME, SALARY
       FROM CUSTOMERS
       WHERE SALARY > 2000;

     SQL> SELECT ID, NAME, SALARY
       FROM CUSTOMERS
       WHERE NAME = 'Hardik';
```

And operator

```
SELECT column1, column2, columnN
FROM table_name
WHERE [condition1] AND [condition2]...AND [conditionN];

E.g.SQL> SELECT ID, NAME, SALARY
FROM CUSTOMERS
WHERE SALARY > 2000 AND age < 25;
```

Or operator

```
SELECT column1, column2, columnN
FROM table_name
WHERE [condition1] OR [condition2]...OR [conditionN]


SQL> SELECT ID, NAME, SALARY
FROM CUSTOMERS
WHERE SALARY > 2000 OR age < 25;
```

ELIMINATING DUPLICATE ROWS WHEN USING A SELECT STATEMENT

```
SELECT DISTINCT column1, column2,.....columnN
FROM table_name
WHERE [condition]

E.g. SQL> SELECT DISTINCT SALARY FROM CUSTOMERS;
```

SORTING DATA IN A TABLE

```
SELECT column-list
FROM table_name
[WHERE condition]
[ORDER BY column1, column2, .. columnN] [ASC | DESC];
```

E.g. SQL> SELECT * FROM CUSTOMERS
ORDER BY NAME DESC;

CREATING A TABLE FROM A TABLE

```
CREATE TABLE table_name [(column1, column2, ... columnN)]
  AS SELECT column1, column2, ...columnN
   FROM table_name
```

DELETE OPERATION

```
DELETE FROM table_name
WHERE [condition];
```

E.g. SQL> DELETE FROM CUSTOMERS; //Removal of all rows

```
SQL> DELETE FROM CUSTOMERS WHERE ID = 6;
```

E.g. SQL> DELETE FROM CUSTOMERS WHERE ID = 6; //Removal of specific rows

UPDATING THE CONTENTS OF A TABLE

```
UPDATE table_name
SET column1 = value1, column2 = value2...., columnN = valueN
WHERE [condition];
```

E.g. SQL> UPDATE CUSTOMERS
SET ADDRESS = 'Pune' //updating records conditionally
WHERE ID = 6;

SQL> UPDATE CUSTOMERS
SET ADDRESS = 'Pune', SALARY = 1000.00; //updating all rows

MODIFYING THE STUCTURE OF TABLES

```
ALTER TABLE table_name ADD column_name datatype;
```

E.g. ALTER TABLE CUSTOMERS ADD SEX char(1);

ALTER TABLE table_name DROP COLUMN column_name;

E.g. ALTER TABLE CUSTOMERS DROP SEX;

```
ALTER TABLE table_name ALTER COLUMN column_name datatype;// TO CHANGE THE DATATYPE

ALTER TABLE table_name MODIFY column_name datatype NOT NULL;
//TO ADD CONSTRAINT NOT NULL
```

RENAMING TABLES

RENAME table to newtable;

E.g. SQL > RENAME customer to customer1;

TRUNCATING TABLES

TRUNCATE TABLE tablename;

DESTROYING TABLES

DROP TABLE tablename;

Truncate table differs from delete:

1) Truncate operations drop and re-create the table, which is much faster than deleting rows one by one.

Exercise

Create the tables for the following:

Table Name: CLIENT_MASTER

Description: Used to store client information

Column Name	Data Type	Size	Default	Attributes
CLIENTNO	Varchar2	6		
NAME	Varchar2	20		
ADDRESS1	Varchar2	30		
ADDRESS2	Varchar2	30		
CITY	Varchar2	15		
PINCODE	Number	8		
STATE	Varchar2	15		
BALDUE	Number	10,2		

Table Name: PRODUCT_MASTER

Description: used to store product information

Column Name	Data Type	Size	Default	Attributes
PRODUCTNO	Varchar2	6		
DESCRIPTION	Varchar2	15		
PROFITPERCENT	Number	4,2		
UNITMEASURE	Varchar2	10		
QTYONHAND	Number	8		
REORDERVL	Number	8		
SELLPRICE	Number	8,2		
COSTPRICE	Number	8,2		

Table Name: SALESMAN_MASTER

Description: Used to store salesman information working for the company.

Column Name	Data Type	Size	Default	Attributes
SALESMANNO	Varchar2	6		
SALESMANNAME	Varchar2	20		
ADDRESS1	Varchar2	30		
ADDRESS2	Varchar2	30		
CITY	Varchar2	20		
PINCODE	Number	8		
STATE	Varchar2	20		
SALAMT	Number	8,2		
TGTTOGET	Number	6,2		
YTDSALES	Number	6,2		
REMARKS	Varchar2	60		

INSERT THE DATA INTO THEIR RESPECTIVE TABLES

CLIENT_MASTER Table

ClientNo	Name	City	Pincode	State	Baldue
C00001	Korth sudarshan	Mumbai	400054	Maharashtra	15000
C00002	Mamta Muzumdar	Madras	780001	Tamil Nadu	0
C00003	Chhaya Bankar	Mumbai	400057	Maharashtra	5000
C00004	Ashwini Joshi	Bangalore	560001	Karnataka	0
C00005	Hansel Colaco	Mumbai	400060	Maharashtra	2000
C00006	Deepak Sharma	Mangalore	560050	Karnataka	0

PRODUCT_MASTER

ProductNo	Description	Profit Percent	Unit Measure	QtyOn Hand	ReorderLvl	SellPrice	CostPrice
P00001	T-Shirts	5	Piece	200	50	350	250
P0345	Shirts	6	Piece	150	50	500	350
P06734	Cotton Jeans	5	Piece	100	20	600	450
P07865	Jeans	5	Piece	100	20	750	500
P07868	Trousers	2	Piece	150	50	850	550
P07885	Pull Overs	2.5	Piece	80	30	700	450
P07965	Denim Shirts	4	Piece	100	4	350	250
P07975	Lycra Tops	5	Piece	70	30	300	175
P08865	Skirts	5	Piece	75	30	450	300

SALESMAN_MASTER

SalesmanNo	Name	Address1	Address2	City	PinCode	State	SalAmt	TgtToGet	YtdSales	Remarks
S00001	Aman	A/14	Worli	Mumbai	400002	Maharashtra	3000	100	50	Good
S00002	Omkar	65	Nariman	Mumbai	400001	Maharashtra	3000	200	100	Good
S00003	Ray	P-7	Bandra	Mumbai	400032	Maharashtra	3000	200	100	Good
S00004	Ashish	A/5	Juhu	Mumbai	400044	Maharashtra	3500	200	150	Good

1) Find out the names of all the clients.
2) Retrieve the entire contents of the Client_Master table.
3) Retrieve the list of names, city and the state of all clients.
4) List the various products available from the Product_Master table.
5) List all the clients who are located in Mumbai.
6) Find the names of salesmen who have a salary equal to Rs. 3000.
7) Change the city of ClientNO ‘C00005’ to ‘Bangalore’.

8) Change the BalDue of ClientNo ‘C00001’ to Rs. 1000.
9) Change the cost price of ‘Trousers’ to Rs. 950.00.
10) Change the city of the salesman to Pune.
11) Delete all salesmen from the Salesman_Master whose salaries are equal to Rs. 3500.
12) Delete all products from Product_Master where the quantity on hand is equal to 100.
13) Delete from Client_Master where the column state holds the value ‘Tamil Nadu’.
14) Add a column called ‘Telephone’ of data type ‘number’ and size = ‘10’ to the Client_Master table.
15) Change the size of SellPrice column in Product_Master to 10,2.
16) Destroy the table Client_Master along with its data.
17) Change the name of the Salesman_Master table to sman_mast.