ADVANCE DATABASE MANAGEMENT

PRACTICAL

1. implement DDL (CREATE, ALTER, DROP, TRUNCATE) and DML (INSERT, UPDATE, DELETE) SQL Command.

SOLUTION:

CREATE:

```
SQL> create table std1(s_name varchar(13),s_id number(12)primary key,s_enrollment
number(14));
Table created.
```

INSERT:

```
SQL> insert into std1 values('kahan',11,60029);
1 row created.

SQL>  insert into std1 values('darshil',12,60029);
1 row created.

SQL>  insert into std1 values('druv',13,60051);
1 row created.
```

OUTPUT:

```
SQL> select*from std;

S_NAME          S_ID S_ENROLLMENT
--------------- ---------- ------------
kahan             11       60029
darshil           12       60029
druv              13       60051
```

UPDATE:

```
SQL> update std1 set s_name='jay'where s_id=12;

1 row updated.
```

OUTPUT:

```
SQL> select * from std1;

S_NAME               S_ID S_ENROLLMENT
------------- ---------- ------------
kahan                  11        60029
jay                    12        60029
druv                   13        60051

SQL> |
```

DROP:

```
SQL> alter table std1 drop column s_enrollment;

Table altered.
```

OUTPUT:

```
SQL> desc std1;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 S_NAME                                             VARCHAR2(13)
 S_ID                                      NOT NULL NUMBER(12)

SQL> |
```

DELETE:

```
SQL> delete from std1 where s_name='jay';

1 row deleted.
```

OUTPUT:

```
SQL> select * from std1;

S_NAME               S_ID
------------- ----------
kahan                 11
druv                  13

SQL> |
```

ALTER:

```
SQL> alter table std1 add s_dept varchar(13);

Table altered.
```

OUTPUT:

```
SQL> desc std1
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 S_NAME                                             VARCHAR2(13)
 S_ID                                      NOT NULL NUMBER(12)
 S_DEPT                                             VARCHAR2(13)

SQL> |
```

TRUNCATE:

```
SQL> truncate table std1;
Table truncated.
```

OUTPUT:

```
SQL> select * from std1;

no rows selected
```

2. Insert sample data into tables using INSERT command.

SOLUTION:

```
SQL> create table emp2946
(name varchar(11),id number(12),dept varchar(13));

Table created.

SQL> insert into emp2946 values ('kahan',11,'it');

1 row created.

SQL>  insert into emp2946 values ('darshil',12,'it');

1 row created.

SQL>  insert into emp2946 values ('dhruv',13,'civil');

1 row created.

SQL>  insert into emp2946 values ('jay',14,'civil');

1 row created.

SQL>  insert into emp2946 values ('het',15,'m_tech');

1 row created.

SQL>  insert into emp2946 values ('mumukshu',16,'it');

1 row created.

SQL>  insert into emp2946 values ('dev',17,'it');

1 row created.

SQL>  insert into emp2946 values ('yug',18,'civil');

1 row created.

SQL>  insert into emp2946 values ('aarav',19,'civil');

1 row created.

SQL>  insert into emp2946 values ('daksh',20,'civil');

1 row created.
```

OUTPUT:

```
SQL> select * from emp2946;

NAME                 ID DEPT
---------- ---------- -------------
kahan                11 it
darshil              12 it
dhruv                13 civil
jay                  14 civil
het                  15 m_tech
mumukshu             16 it
dev                  17 it
yug                  18 civil
aarav                19 civil
daksh                20 civil

10 rows selected.

SQL>
```

3. Apply all database Entity Integrity constraints (i.e. Primary key, Foreign key, NOT NULL, Unique and CHECK).

SOLUTION:

```
create table std1 (s_id number(3) primary key, s_name varchar(20) not null, s_age int
check(age>=18 and age<=30) ,email varchar(50) unique);

insert into std1 values (111,'kahan',25,'kp46@gmail.com');
insert into std1 values (112,'het',30,'het40@gmail.com');
insert into std1 values (113,'mumu',26,'mumu@gmail.com');
insert into std1 values (114,'darshil',20,'deadly50@gmail.com');
insert into std1 values (115,'dhruv',26,'dhruv26@gmail.com');
```

OUTPUT:

```
SQL> select * from std;

      S_ID S_NAME                    S_AGE
---------- -------------------- ----------
EMAIL
--------------------------------------------------
       111 kahan                        25
kp46@gmail.com

       112 het                          30
het40@gmail.com

       113 mumu                         26
mumu@gmail.com


      S_ID S_NAME                    S_AGE
---------- -------------------- ----------
EMAIL
--------------------------------------------------
       114 darshil                      20
deadly50@gmail.com

       115 dhruv                        26
dhruv26@gmail.com


SQL>
```

Foreign key:

```
create table s_course (c_id number(3) primary key, s_id number(3), c_name varchar(10)
not null, foreign key (s_id) references std1(s_id));

insert into s_course values (121,111,'IT');
insert into s_course values (122,112,'IT');
insert into s_course values (123,113,'Comp');
insert into s_course values (124,114,'Civil');
insert into s_course values (125,115,'Comp');

select s_name,c_id,c_name from std1 inner join s_course on std1.s_id=s_course.s_id;
```

OUTPUT:

```
SQL> select * from  s_course;

      C_ID       S_ID C_NAME
---------- ---------- ----------
       121        111 IT
       122        112 IT
       123        113 Comp
       124        114 Civil
       125        115 Comp

SQL>
```

4. **Apply all database Domain constraints.**

SOLUTION:

```
SQL> create table student46 (student_id number(3) primary key, name varchar(20) not
null, age int check(age>=18 and age<=30));

Table created.
```

OUTPUT:

```
SQL>      Insert into student29 values (101,'kahan',20);

1 row created.

SQL> Insert into student29 values (102,'kahan',17);
Insert into student29 values (102,'kahan',17)
*
ERROR at line 1:
ORA-02290: check constraint (HR.SYS_C005534) violated


SQL>
```