

## PRACTICAL 7: STUDY OF VARIOUS TYPES OF SET OPERATORS

Suppose that a Product table contains two attributes, PROD\_CODE and VEND\_CODE. The values for the PROD\_CODE are: ABC, DEF, GHI and JKL. These are matched by the following values for the VEND\_CODE: 125, 124, 124 and 123, respectively (e.g., PROD\_CODE value ABC corresponds to VEND\_CODE value 125). The Vendor table contains a single attribute, VEND\_CODE, with values 123, 124, 125 and 126. (The VEND\_CODE attribute in the Product table is a foreign key to the VEND\_CODE in the Vendor table.)

Given the information, what would be the query output for the following? Show values.

Run SQL Command Line

```
SQL> create table Vendor(VEND_CODE int primary key);
Table created.

SQL> insert into Vendor values(123);
1 row created.

SQL> insert into Vendor values(124);
1 row created.

SQL> insert into Vendor values(125);
1 row created.

SQL> insert into Vendor values(126);
1 row created.

SQL> commit;
Commit complete.

SQL> select * from Vendor;

VEND_CODE
-----
123
124
125
126
```

# IMRAN RIZWAN SHAIKH 109 FYIT PRACTICAL 7 DBMS

```
Run SQL Command Line
SQL> create table Product(PROD_CODE varchar2(10) primary key,VEND_CODE int not null,foreign key(VEND_CODE) references Vendor(VEND_CODE));
Table created.
SQL> Insert into Product values('ABC',125);
1 row created.
SQL> Insert into Product values('DEF',124);
1 row created.
SQL> Insert into Product values('GHI',124);
1 row created.
SQL> Insert into Product values('JKL',123);
1 row created.
SQL> SELECT * FROM Product;
PROD_CODE  VEND_CODE
-----
ABC        125
DEF        124
GHI        124
JKL        123
SQL>
```

a) A UNION query based on these two tables

```
Run SQL Command Line
SQL> SELECT VEND_CODE FROM VENDOR UNION SELECT VEND_CODE FROM PRODUCT;

VEND_CODE
-----
123
124
125
126
SQL>
```

b) A UNION ALL query based on these two tables

```
Run SQL Command Line
SQL> SELECT VEND_CODE FROM VENDOR UNION ALL SELECT VEND_CODE FROM PRODUCT;

VEND_CODE
-----
123
124
125
126
125
124
124
123

8 rows selected.
SQL>
```

c) An INTERSECT query based on these two tables

Run SQL Command Line

```
SQL> SELECT VEND_CODE FROM VENDOR INTERSECT SELECT VEND_CODE FROM PRODUCT;
```

```
VEND_CODE
```

```
-----  
123  
124  
125
```

```
SQL> _
```

d) A MINUS query based on these two tables

Run SQL Command Line

```
SQL> SELECT VEND_CODE FROM VENDOR MINUS SELECT VEND_CODE FROM PRODUCT;
```

```
VEND_CODE
```

```
-----  
126
```

```
SQL>
```