Lab 6
(SQL Queries Set 4)

Objective:
To learn queries related to cartesian products, join(natural join, theta join, equi join, left outer join, right outer join, full outer join etc), and set operations(such as union, intersection and difference).

Q1.

We have table given as:

Students

studno	name	course
100	Fred	PH
200	Dave	CM
300	Bob	CM

and

Courses

courseno	name
PH	Pharmacy
CM	Computing

Task1: Create the schema for the tables Students and Courses.

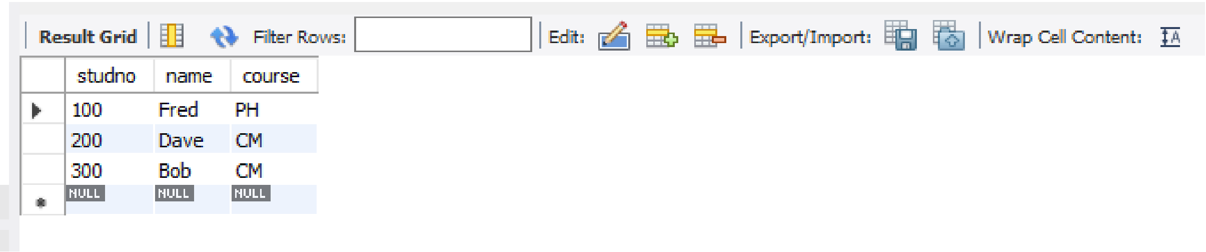
```
use  bct_cd;
```

```
create table Students(
studno integer primary key,
name varchar(50),
course varchar(4)
);
```

```
create table Courses(
courseno varchar(4) primary key,
name varchar(50)
);
```

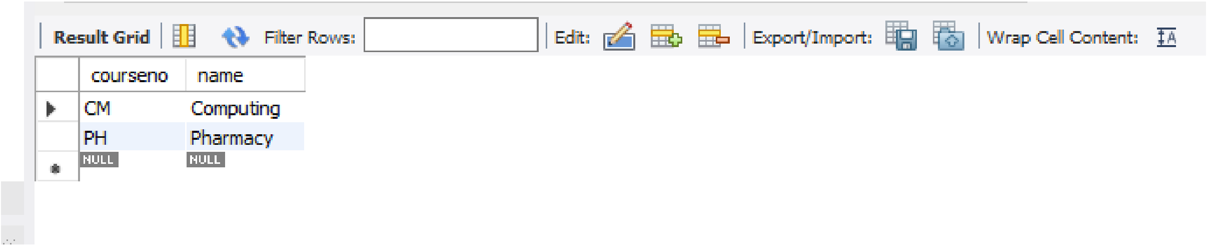
Task 2: Populating the tables with above given values in tables.

```
insert into Students
values
(100,'Fred','PH'),
(200,'Dave','CM'),
(300,'Bob','CM');

select * from Students;
```



studno	name	course
100	Fred	PH
200	Dave	CM
300	Bob	CM
NULL	NULL	NULL

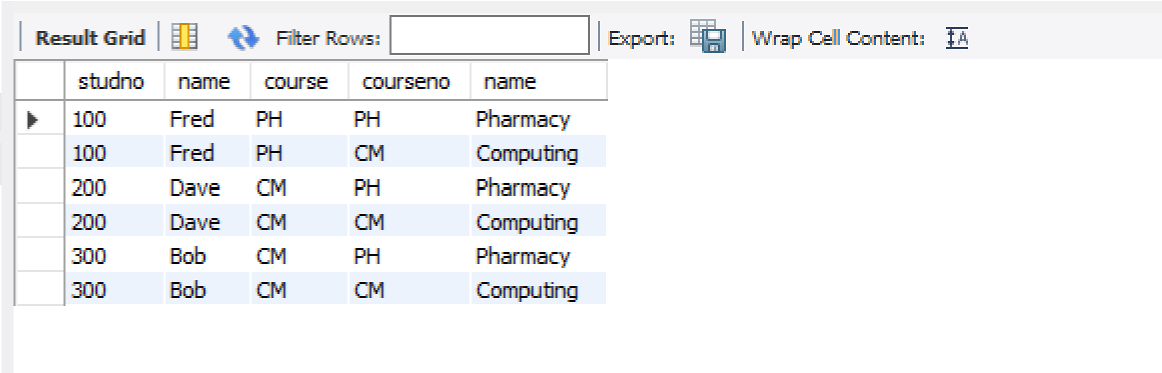
```
insert into Courses
values
('PH','Pharmacy'),
('CM', 'Computing');

select * from Courses;
```



courseno	name
CM	Computing
PH	Pharmacy
NULL	NULL

Task 3: SQL query to display the cartesian product of two tables.

```
select * from Students, Courses;
```

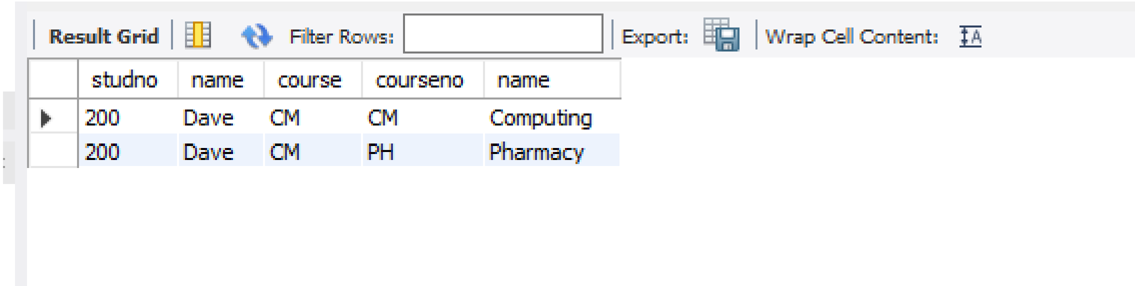


studno	name	course	courseno	name
100	Fred	PH	PH	Pharmacy
100	Fred	PH	CM	Computing
200	Dave	CM	PH	Pharmacy
200	Dave	CM	CM	Computing
300	Bob	CM	PH	Pharmacy
300	Bob	CM	CM	Computing

Task 4: SQL query to display the result of the theta join operation.

Students ⋈(studno=200) **Courses**

select * from Students, Courses
where studno = 200 ;

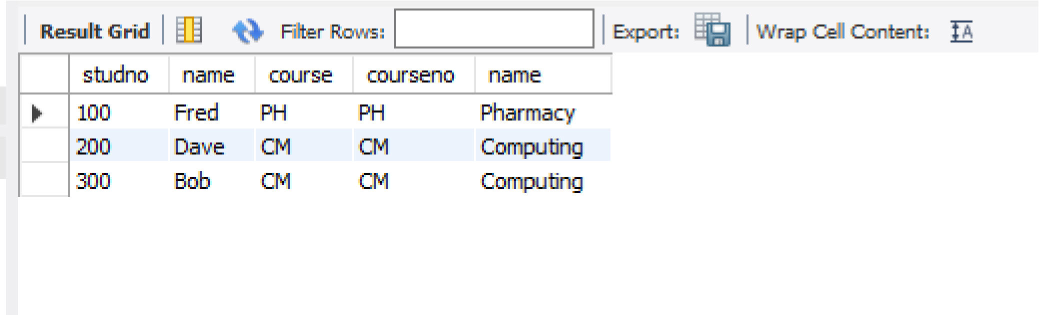


studno	name	course	courseno	name
200	Dave	CM	CM	Computing
200	Dave	CM	PH	Pharmacy

Task 5: SQL query to display the result of the equi join operation.

Students ⋈(course=courseno) **Courses**

select * from Students, Courses
where course = courseno;



studno	name	course	courseno	name
100	Fred	PH	PH	Pharmacy
200	Dave	CM	CM	Computing
300	Bob	CM	CM	Computing

Q2. Given tables are:

r

a	**b**

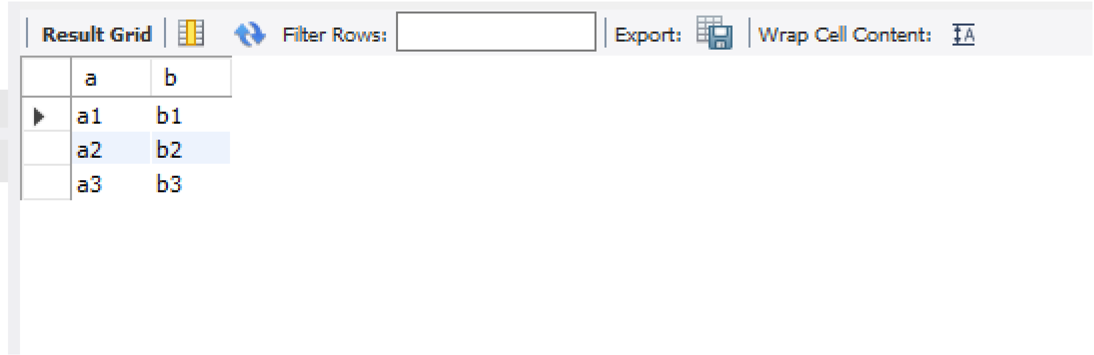
a1	b1
a2	b2
a3	b3

s

b	**c**
b1	c1
b2	c2
b4	c4

Task 1: SQL query to create above schema table r and s and populate them.

```
use  bct_cd;

create table r(
a varchar(4),
b varchar(4)
);

create table s(
b varchar(4),
c varchar(4)
);

insert into r values
('a1','b1'),
('a2','b2'),
('a3', 'b3');

select * from r;
```

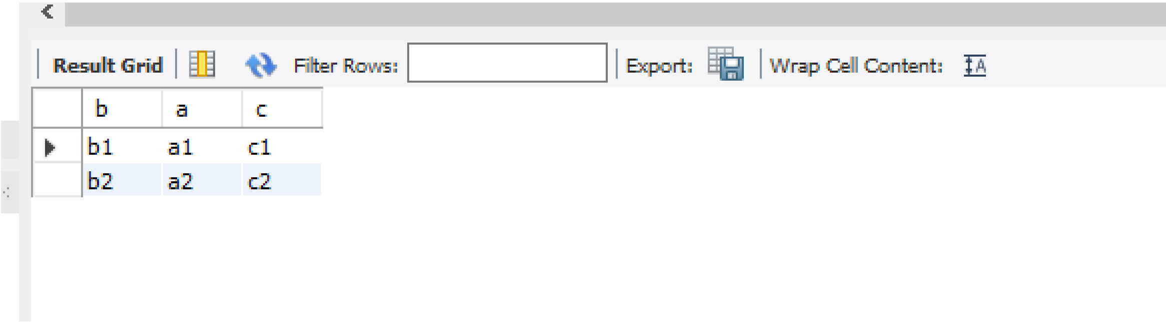


a	b
a1	b1
a2	b2
a3	b3

```
insert into s values
('b1','c1'),
('b2','c2'),
('b4', 'c4');
```

Task 2: SQL query to display the result of the natural join operation r ⋈ s.

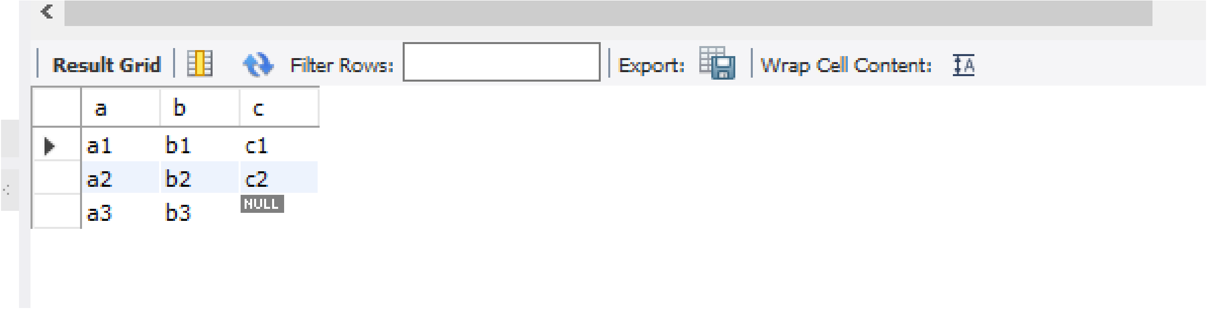
```
select * from r natural join s;
```



b	a	c
b1	a1	c1
b2	a2	c2

Task 3: SQL query to display the result of the left outer join operation r ⟕ s.

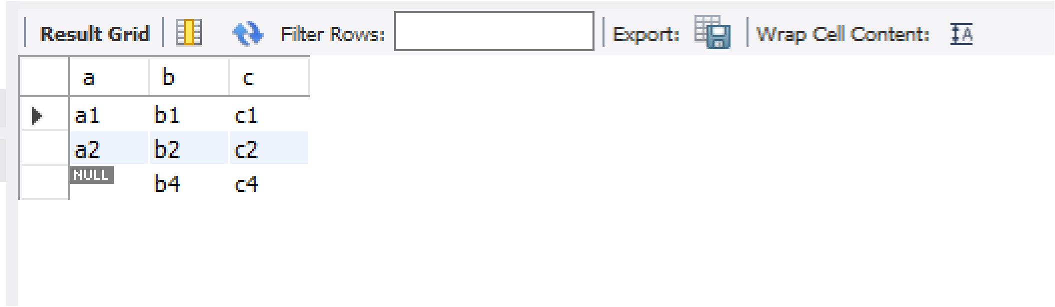
```
select a,r.b,s.c from r left join s on r.b = s.b;
```



a	b	c
a1	b1	c1
a2	b2	c2
a3	b3	NULL

Task 4: SQL query to display the result of the right outer join operation r ⟖ s.

```
select a,s.b,c from r right join s on r.b = s.b;
```

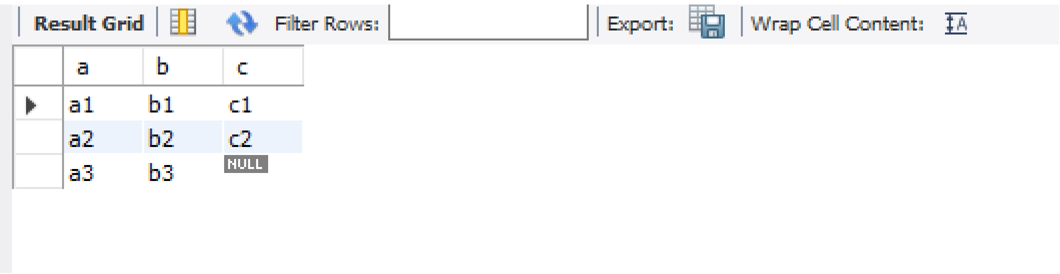


a	b	c
a1	b1	c1
a2	b2	c2
NULL	b4	c4

Task 5: SQL query to display the result of the left outer join operation r ⟕ s.

create view g as
(select a,r.b, s.c from r left join s on r.b = s.b);

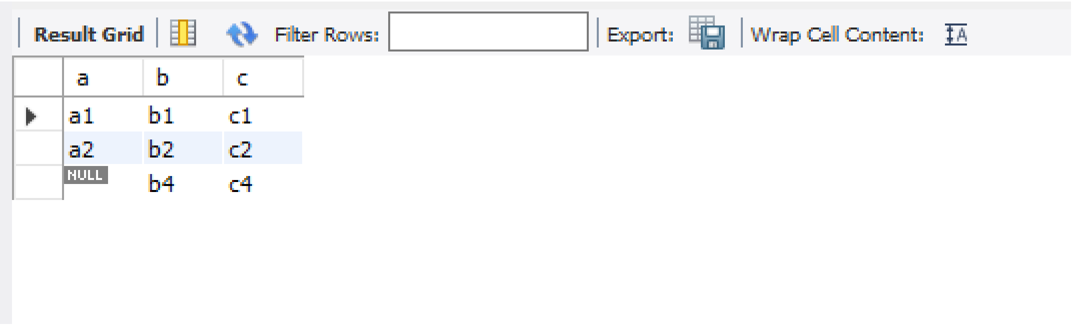
select * from g;



a	b	c
a1	b1	c1
a2	b2	c2
a3	b3	NULL

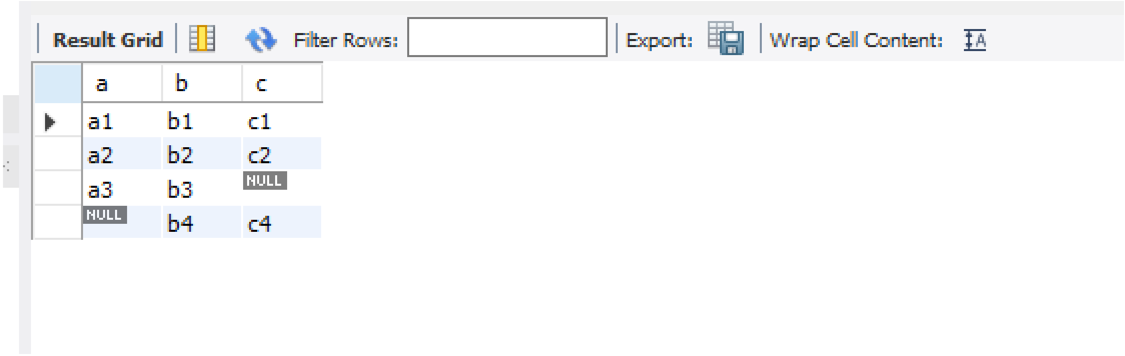
create view h as
(select a,s.b,c from r right join s on r.b =s.b);

select * from h;



a	b	c
a1	b1	c1
a2	b2	c2
NULL	b4	c4

select * from g union select * from h;



a	b	c
a1	b1	c1
a2	b2	c2
a3	b3	NULL
NULL	b4	c4

Q3. Given tables are :

id	**name**
1	A
2	B
3	C
4	D

id	**name**
2	B
3	C
5	E
6	F

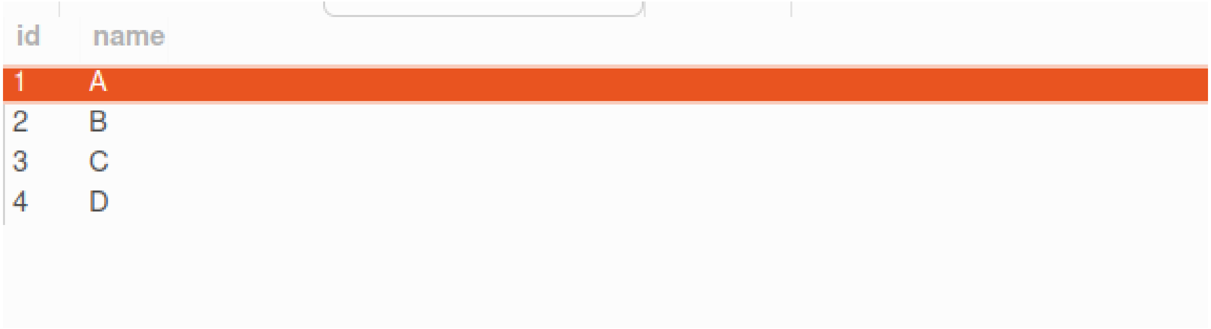
Task 1: SQL query to create schema for the tables First and Second.

```
create table First (
id integer,
name varchar(4)
);

create table Second(
id integer,
name varchar(4)
);
```

Task 2: SQL query to populate the indicated values.

```
insert into First
values
(1,'A'),
(2,'B'),
(3,'C'),
(4,'D');
```

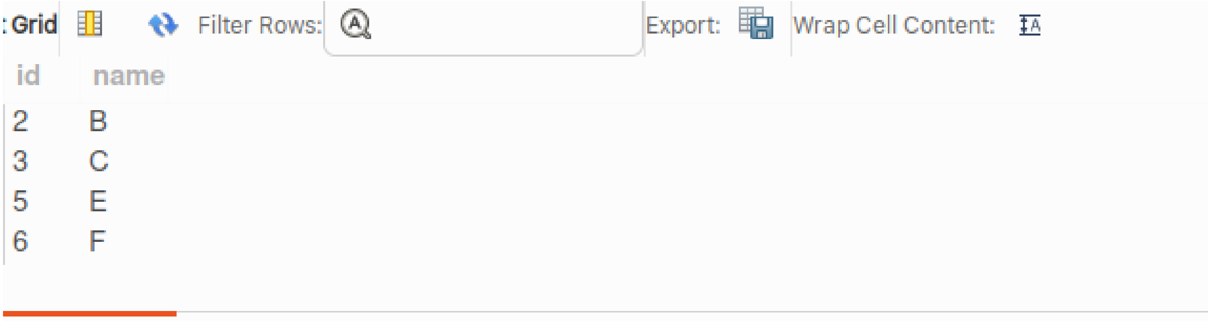
```
select * from First;
```



id	name
1	A
2	B
3	C
4	D

```
insert into Second
values
(2,'B'),
(3,'C'),
(5, 'E'),
(6,'F');
```

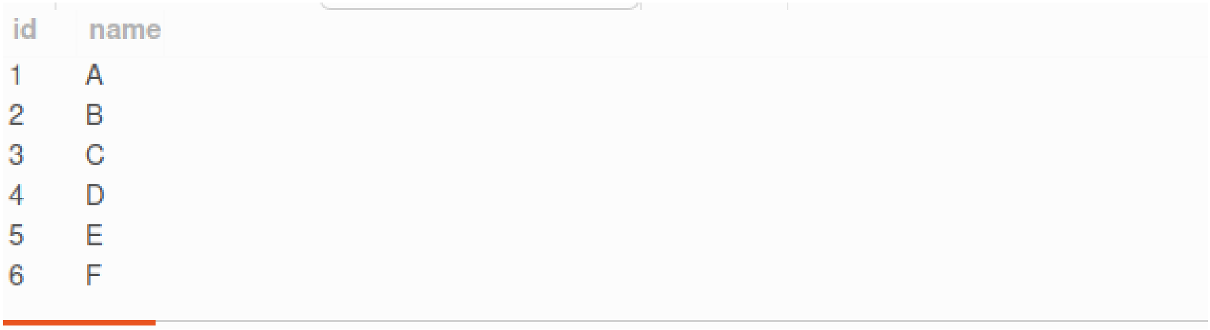
```
Select * from Second;
```



id	name
2	B
3	C
5	E
6	F

Task 3: SQL query to find the union of two tables.

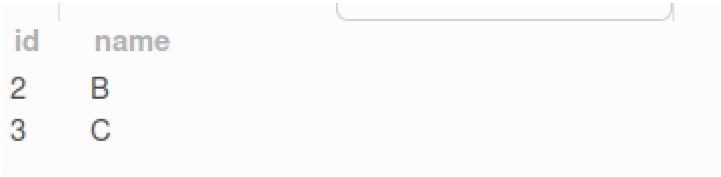
```
select * from First union Select * from Second;
```



id	name
1	A
2	B
3	C
4	D
5	E
6	F

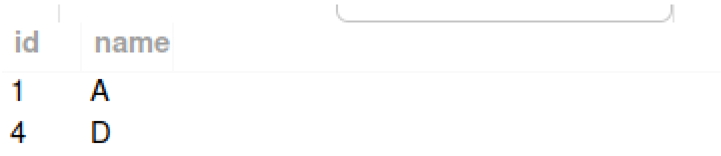
Task 4: SQL query to find the intersection of the two tables.

select * from First where id in(Select id from Second);



id	name
2	B
3	C

select * from First where id not in (Select id from Second);



id	name
1	A
4	D

DISCUSSION

In this lab, we created a table *Students and Courses* in qn1 and performed different operations like Cartesian product, theta join, equi join and output was verified.

In qn2, table called *r* and *s* was created and operations like natural join (r⋈s), left outer join, right outer join, full outer join was performed. We also used (create view g as ..) to view different attributes of table as table name *g.*

In qn3, two tables *First* and *Second* were created and inserted different values for operations like union, intersection, difference. The result was displayed and verified.

CONCLUSION

From this lab, we learned about the fundamentals of relational database operations, including Cartesian product, natural join, theta join, equi join, and set operations like union, intersection, and difference, equips individuals with the essential skills to manipulate and extract information efficiently. With this knowledge, users can perform diverse operations, such as combining all possible pairs of rows, establishing connections based on specified criteria, and conducting powerful set operations for efficient data retrieval and analysis within relational databases.