

Lab-04

Instructor:

Table Create:

```
CREATE TABLE `instructor` ( `Id` varchar(30), `Name` varchar(30), `Dept_Name` varchar(30), `Salary` int(30) );
```

Table data input:

```
INSERT INTO `instructor` (`Id`, `Name`, `Dept_Name`, `Salary`) VALUES
```

```
('10101', 'Srinivasan', 'CSE', 65000),
```

```
('12121', 'Wu', 'FIN', 90000),
```

```
('15151', 'Mozart', 'Music', 40000),
```

```
('22222', 'Einstein', 'Physics', 50000),
```

```
('32343', 'Said', 'History', 60000),
```

```
('33456', 'Gold', 'Physics', 87000),
```

```
('45555', 'Katz', 'CSE', 75000),
```

```
('58583', 'Cali', 'History', 62000),
```

```
('76543', 'Singh', 'FIN', 80000),
```

```
('76766', 'Crick', 'Bio', 72000),
```

```
('83821', 'Brandt', 'CSE', 92000),
```

```
('98345', 'Kin', 'EEE', 80000);
```

Teaches:

Table create:

```
CREATE TABLE `teaches` ( `Id` varchar(30), `C_Id` varchar(30), `Section_Id` int(10));
```

Table data input: INSERT INTO `teaches` (`Id`, `C_Id`, `Section_Id`) VALUES ('10101', 'CSE_101', 1), ('12121', 'CSE_111', 2), ('13131', 'CSE_311', 3);

Q1. Perform Cartesian Product Operation between these two relation.

```
SELECT * FROM instructor, teaches;
```

Id	Name	Dept_Name	Salary	Id	C_Id	Section_Id
10101	Srinivasan	CSE	65000	10101	CSE_101	1
10101	Srinivasan	CSE	65000	12121	CSE_111	2
10101	Srinivasan	CSE	65000	13131	CSE_311	3
12121	Wu	FIN	90000	10101	CSE_101	1
12121	Wu	FIN	90000	12121	CSE_111	2
12121	Wu	FIN	90000	13131	CSE_311	3
15151	Mozart	Music	40000	10101	CSE_101	1
15151	Mozart	Music	40000	12121	CSE_111	2
15151	Mozart	Music	40000	13131	CSE_311	3
22222	Einstein	Physics	50000	10101	CSE_101	1
22222	Einstein	Physics	50000	12121	CSE_111	2
22222	Einstein	Physics	50000	13131	CSE_311	3
32343	Said	History	60000	10101	CSE_101	1
32343	Said	History	60000	12121	CSE_111	2
32343	Said	History	60000	13131	CSE_311	3
33456	Gold	Physics	87000	10101	CSE_101	1
33456	Gold	Physics	87000	12121	CSE_111	2
33456	Gold	Physics	87000	13131	CSE_311	3
45555	Katz	CSE	75000	10101	CSE_101	1
45555	Katz	CSE	75000	12121	CSE_111	2
45555	Katz	CSE	75000	13131	CSE_311	3
58583	Cali	History	62000	10101	CSE_101	1
58583	Cali	History	62000	12121	CSE_111	2
58583	Cali	History	62000	13131	CSE_311	3
76543	Singh	FIN	80000	10101	CSE_101	1

Q2. Find those instructors who teaches any of the courses.

```
SELECT Name FROM instructor,teaches WHERE instructor.Id=teaches.Id;
```

Name
Srinivasan
Wu

Q3. Find only instructor names and course id for instructors in the Computer Science department.

```
Select Name,instructor.Id from instructor INNER JOIN teaches where instructor.Id=teaches.Id AND Dept_Name= 'CSE';
```

Name	Id
Srinivasan	10101

Q4. Find the total no. of tuples in "Instructor" relation.

```
SELECT COUNT(*) FROM instructor;
```

Extra options
COUNT(*)
12

Q5. Answer Q2 using Natural Join.

```
SELECT * FROM instructor NATURAL JOIN teaches;
```

Id	Name	Dept_Name	Salary	C_Id	Section_Id
10101	Srinivasan	CSE	65000	CSE_101	1
12121	Wu	FIN	90000	CSE_111	2

Q6. Perform Left Outer Join.

```
SELECT * FROM instructor NATURAL LEFT OUTER JOIN teaches;
```

Id	Name	Dept_Name	Salary	C_Id	Section_Id
10101	Srinivasan	CSE	65000	CSE_101	1
12121	Wu	FIN	90000	CSE_111	2
15151	Mozart	Music	40000	NULL	NULL
22222	Einstein	Physics	50000	NULL	NULL
32343	Said	History	60000	NULL	NULL
33456	Gold	Physics	87000	NULL	NULL
45555	Katz	CSE	75000	NULL	NULL
58583	Cali	History	62000	NULL	NULL
76543	Singh	FIN	80000	NULL	NULL
76766	Crick	Bio	72000	NULL	NULL
83821	Brandt	CSE	92000	NULL	NULL
98345	Kin	EEE	80000	NULL	NULL

Q7. Perform Right Outer Join.

```
SELECT * FROM instructor NATURAL RIGHT OUTER JOIN teaches;
```

Id	C_Id	Section_Id	Name	Dept_Name	Salary
10101	CSE_101	1	Srinivasan	CSE	65000
12121	CSE_111	2	Wu	FIN	90000
13131	CSE_311	3	NULL	NULL	NULL