JOINS (INNER/LEFT/RIGHT/FULL OTER)

There are two tables: advisor and student. Assume that an advisor may have zero or more students.
A student however can have zero or one advisor.

advisor_id (pk)	fname	lname	email
1	Morgan	Freeman	f@email.com
12	Natalie	Portman	p@email.com
22	Diane	Keaton	d@email.com

advisor (parent table)

student_id (pk)	first_name	last_name	advisor_id (fk)
10	Sarah	Adam	12
20	John	Stone	22
30	Nick	Look	NULL
40	Uma	Thurman	12

student (child table)

See the following JOIN examples:

INNER JOIN

1. For each advisor, display their students.
   We need to have an INNER JOIN. (We are interested to find marching rows from both tables).

```
select advisor.advisor_id,
       advisor.fname,
       advisor.lname,
       advisor.email,
       student.student_id,
       student.first_name,
       student.last_name,
       student.advisor_id
from advisor
inner join student
on advisor.advisor_id = student.advisor_id;
```

   The result will be the rows from both tables that satisfy the join condition.
   Two rows from the advisor table are matched with three rows in the student tables.

advisor_id	fname	lname	email	student_id	first_name	last_name	advisor_id
12	Natalie	Portman	p@email.com	10	Sarah	Adam	12
12	Natalie	Portman	p@email.com	40	Uma	Thurman	12
22	Diane	Keaton	d@email.com	20	John	Stone	22

LEFT JOIN

2. Display the information of all advisors. Show the advisors' students if they have any students.

advisor_id (pk)	fname	lname	email
1	**Morgan**	**Freeman**	**f@email.com**
12	**Natalie**	**Portman**	**p@email.com**
22	**Diane**	**Keaton**	**d@email.com**

student_id (pk)	first_name	last_name	advisor_id (fk)
10	**Sarah**	**Adam**	**12**
20	**John**	**Stone**	**22**
30	Nick	Look	NULL
40	**Uma**	**Thurman**	**12**

We want to see all advisors. Those who satisfy the join condition and those who do not satisfy the join condition. (It does not matter if the advisor has any students. We will include them all in the report. However, the values for student data will be null, if the advisor does not have any students.)

```
select advisor.advisor_id,
       advisor.fname,
       advisor.lname,
       advisor.email,
       student.student_id,
       student.first_name,
       student.last_name,
       student.advisor_id
from advisor
left join student
on advisor.advisor_id = student.advisor_id;
```

See the following result:

advisor_id	fname	lname	email	student_id	first_name	last_name	advisor_id
1	Morgan	Freeman	f@email.com	NULL	NULL	NULL	NULL
12	Natalie	Portman	p@email.com	10	Sarah	Adam	12
12	Natalie	Portman	p@email.com	40	Uma	Thurman	12
22	Diane	Keaton	d@email.com	20	John	Stone	22

Right JOIN

3. Display the information of all students. Show the students' advisor if they have any.

advisor_id (pk)	fname	lname	email
1	Morgan	Freeman	f@email.com
12	**Natalie**	**Portman**	**p@email.com**
22	**Diane**	**Keaton**	**d@email.com**

student_id (pk)	first_name	last_name	advisor_id (fk)
10	**Sarah**	**Adam**	**12**
20	**John**	**Stone**	**22**
30	**Nick**	**Look**	**NULL**
40	**Uma**	**Thurman**	**12**

We want to show all students even if they do not have any advisor (We include the students that do not satisfy the join condition.)

table1 right join table2 means all the rows from table2 and the rows from table1 that satisfy the condition.
Assume that table1 is the advisor table and table2 is the student table. So ***advisor right join student*** will return all students and only those advisors that satisfy the join condition.

To get the same result, you can also write ***student left join advisor***. It is the same as the ***advisor right join student*** since we want to see all the rows from student.

```
select advisor.advisor_id,
       advisor.fname,
       advisor.lname,
       advisor.email,
       student.student_id,
       student.first_name,
       student.last_name,
       student.advisor_id
from advisor
right join student
on advisor.advisor_id = student.advisor_id;
```

advisor_id	fname	lname	email	student_id	first_name	last_name	advisor_id
12	Natalie	Portman	p@email.com	10	Sarah	Adam	12
12	Natalie	Portman	p@email.com	40	Uma	Thurman	12
22	Diane	Keaton	d@email.com	20	John	Stone	22
NULL	NULL	NULL	NULL	30	Nick	Look	NULL

FULL OUTER JOIN

4. Display the information of all students and their advisors. Also Show the students that do not have any advisor. Include advisor without student to the report.

advisor_id (pk)	fname	lname	email
1	Morgan	Freeman	f@email.com
12	Natalie	Portman	p@email.com
22	Diane	Keaton	d@email.com

student_id (pk)	first_name	last_name	advisor_id (fk)
10	Sarah	Adam	12
20	John	Stone	22
30	Nick	Look	NULL
40	Uma	Thurman	12

We want to include all rows from both tables, those rows that satisfy the join conditions and the ones that do not satisfy the condition from both table will be included in the result.

```
select advisor.advisor_id,
       advisor.fname,
       advisor.lname,
       advisor.email,
       student.student_id,
       student.first_name,
       student.last_name,
       student.advisor_id
from advisor
full outer join student
on advisor.advisor_id = student.advisor_id;
```

advisor_id	fname	lname	email	student_id	first_name	last_name	advisor_id
1	Morfan	Freeman	f@email.com	NULL	NULL	NULL	NULL
12	**Natalie**	**Portman**	**p@email.com**	**10**	**Sarah**	**Adam**	**12**
12	**Natalie**	**Portman**	**p@email.com**	**40**	**Uma**	**Thurman**	**12**
22	**Diane**	**Keaton**	**d@email.com**	**20**	**John**	**Stone**	**22**
NULL	NULL	NULL	NULL	30	Nick	Look	NULL

We have three rows that satisfy the condition. There is a row from advisor table that do not satisfy the condition (It does not have any students.) but it is included in the result. There is also a student in the student table that does not satisfy the join condition (It does not have any advisor.) but it is included in the result.