1. Convert bookstore.xml into json(file is located in Presentation->XML folder)

```
<bookstore>
  <book>
    <title>Harry Potter</title>
    <author>J.K. Rowling</author>
    <price>29.99</price>
    <available>true</available>
  </book>
  <book>
    <title>The Hobbit</title>
    <author>J.R.R. Tolkien</author>
    <price>19.99</price>
    <available>false</available>
  </book>
</bookstore>
```

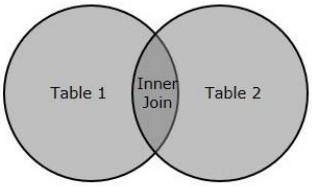
```
JSON:{
Bookstore:
{ BOOK : [ {
“title”:Harry Potter ,
“author” : J.K.Rowling ,
“price”: 29.99
“available”: true
} ,
{
“title” : The Hobbit ,
“author” : J.R.R Tolkien ,
“price” : 19.99 ,
“available” : false } ] } }
```

2. Write a query to give inner join,left outer join,right outer join and full outer join

Inner join:

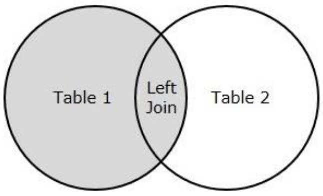
The SQL Inner Join is a type of join that combines multiple tables by retrieving records that have matching values in both tables (in the common column).

It compares each row of the first table with each row of the second table, to find all pairs of rows that satisfy the join-predicate. When the join-predicate is satisfied, the column values from both tables are combined into a new table.



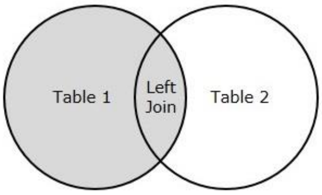
The SQL Left Join

Left Join or Left Outer Join in SQL combines two or more tables, where the first table is returned wholly; but, only the matching record(s) are retrieved from the consequent tables. If zero (0) records are matched in the consequent tables, the join will still return a row in the result, but with NULL in each column from the right table.



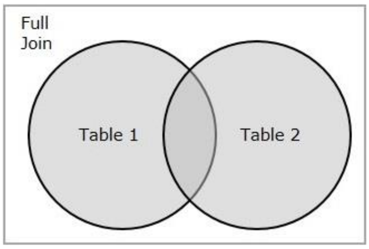
The SQL Left Join

Left Join or Left Outer Join in SQL combines two or more tables, where the first table is returned wholly; but, only the matching record(s) are retrieved from the consequent tables. If zero (0) records are matched in the consequent tables, the join will still return a row in the result, but with NULL in each column from the right table.



The SQL Full Join

SQL Full Join creates a new table by joining two tables as a whole. The joined table contains all records from both the tables and fills NULL values for missing matches on either side. In short, full join is a type of outer join that combines the result-sets of both left and right joins.



You can also achieve the equivalent result-set of FULL JOIN by performing the UNION operation on result-sets of the LEFT JOIN and RIGHT JOIN.

```
mysql> create database dep;

Query OK, 1 row affected (0.03 sec)


mysql> use dep;

Database changed

mysql> create table department(dep_id int primary key,dep_name char(40));

Query OK, 0 rows affected (0.03 sec)


mysql> insert into department values(10,'HR');

Query OK, 1 row affected (0.01 sec)


mysql> insert into department values(20,'Sales');

Query OK, 1 row affected (0.00 sec)


mysql> insert into department values(30,'IT');

Query OK, 1 row affected (0.00 sec)
```

```
mysql> insert into department values(40,'Marketing');
Query OK, 1 row affected (0.02 sec)

mysql> select * from department;
+--------+-----------+
| dep_id | dep_name  |
+--------+-----------+
|     10 | HR        |
|     20 | Sales     |
|     30 | IT        |
|     40 | Marketing |
+--------+-----------+
4 rows in set (0.00 sec)

mysql> create table employee(emp_id int primary key,first_name char(40),last_name char(40),department_id int,foreign key(department_id) references department (dep_id));
Query OK, 0 rows affected (0.06 sec)

mysql> insert into employee values(1,'Anirudh','Ravichandar',10);
Query OK, 1 row affected (0.02 sec)

mysql> insert into employee values(2,'Geeth','Sai',20);
Query OK, 1 row affected (0.00 sec)

mysql> insert into employee values(3,'george','clint',30);
Query OK, 1 row affected (0.01 sec)

mysql> insert into employee values(4,'SS','Thaman',10);
Query OK, 1 row affected (0.02 sec)
```

```
mysql> select * from employee;

+--------+------------+-------------+---------------+

| emp_id | first_name | last_name   | department_id |

+--------+------------+-------------+---------------+

|      1 | Anirudh    | Ravichandar |            10 |

|      2 | Geeth      | Sai         |            20 |

|      3 | george     | clint       |            30 |

|      4 | SS         | Thaman      |            10 |

+--------+------------+-------------+---------------+

4 rows in set (0.00 sec)


mysql> select * from employee e inner join department d on e.department_id=d.dep_id;

+--------+------------+-------------+---------------+--------+----------+

| emp_id | first_name | last_name   | department_id | dep_id | dep_name |

+--------+------------+-------------+---------------+--------+----------+

|      1 | Anirudh    | Ravichandar |            10 |     10 | HR       |

|      2 | Geeth      | Sai         |            20 |     20 | Sales    |

|      3 | george     | clint       |            30 |     30 | IT       |

|      4 | SS         | Thaman      |            10 |     10 | HR       |

+--------+------------+-------------+---------------+--------+----------+

4 rows in set (0.00 sec)


mysql> select * from employee e right join department d on e.department_id=d.dep_id;

+--------+------------+-------------+---------------+--------+-----------+

| emp_id | first_name | last_name   | department_id | dep_id | dep_name  |

+--------+------------+-------------+---------------+--------+-----------+

|      1 | Anirudh    | Ravichandar |            10 |     10 | HR        |

|      4 | SS         | Thaman      |            10 |     10 | HR        |

|      2 | Geeth      | Sai         |            20 |     20 | Sales     |

|      3 | george     | clint       |            30 |     30 | IT        |

|   NULL | NULL       | NULL        |          NULL |     40 | Marketing |
+--------+------------+-------------+---------------+--------+-----------+

5 rows in set (0.00 sec)


mysql> select *from employee e full join department d on e.department_id=d.dep_id

    -> ;

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'full join department d on e.department_id=d.dep_id' at line 1

mysql> select * from employee e left join department d on e.department_id=d.dep_id;

+--------+------------+-------------+---------------+--------+----------+

| emp_id | first_name | last_name   | department_id | dep_id | dep_name |

+--------+------------+-------------+---------------+--------+----------+

|      1 | Anirudh    | Ravichandar |            10 |     10 | HR       |

|      2 | Geeth      | Sai         |            20 |     20 | Sales    |

|      3 | george     | clint       |            30 |     30 | IT       |

|      4 | SS         | Thaman      |            10 |     10 | HR       |

+--------+------------+-------------+---------------+--------+----------+

4 rows in set (0.00 sec)


mysql> select *from employee e full outer join department d on e.department_id=d.dep_id;

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'full outer join department d on e.department_id=d.dep_id' at line 1

mysql> select * from employee e

    -> union department d on e.department_id=d.dep_id;

ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'department d on e.department_id=d.dep_id' at line 2

mysql> select * from employee e left join department d on e.department_id=d.dep_id

    -> union

    -> select * from department d right join employee e on e.department_id=d.dep_id;

+--------+------------+-------------+---------------+-------------+----------+

| emp_id | first_name | last_name   | department_id | dep_id      | dep_name |
+--------+------------+-------------+---------------+-------------+----------+
|      1 | Anirudh    | Ravichandar | 10            | 10          | HR       |
|      2 | Geeth      | Sai         | 20            | 20          | Sales    |
|      3 | george     | clint       | 30            | 30          | IT       |
|      4 | SS         | Thaman      | 10            | 10          | HR       |
|     10 | HR         | 1           | Anirudh       | Ravichandar | 10       |
|     20 | Sales      | 2           | Geeth         | Sai         | 20       |
|     30 | IT         | 3           | george        | clint       | 30       |
|     10 | HR         | 4           | SS            | Thaman      | 10       |
+--------+------------+-------------+---------------+-------------+----------+
```

3. Write a query to find duplicate records(refer SQL_Assignments in Presentation folder)

```
mysql> create table employee(emp_id int primary key,first_name char(40),last_name char(40),email varchar(50));
Query OK, 0 rows affected (0.04 sec)

mysql> insert into employee values(1,'jhon','Doe','jhon.doe@example.com');
Query OK, 1 row affected (0.02 sec)

mysql> insert into employee values(2,'jane','Smith','jane.smith@example.com');
Query OK, 1 row affected (0.02 sec)

mysql> update employee set first_name='John' where emp_id=1;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> insert into employee values(3,'john','Doe','john.doe@example.com');
Query OK, 1 row affected (0.01 sec)

mysql> insert into employee values(4,'Emily','Davis','Emily.davis@example.com');
```

```
Query OK, 1 row affected (0.02 sec)

mysql> select * from employee;
+--------+------------+-----------+-------------------------+
| emp_id | first_name | last_name | email                   |
+--------+------------+-----------+-------------------------+
|      1 | John       | Doe       | jhon.doe@example.com    |
|      2 | jane       | Smith     | jane.smith@example.com  |
|      3 | john       | Doe       | john.doe@example.com    |
|      4 | Emily      | Davis     | Emily.davis@example.com |
+--------+------------+-----------+-------------------------+
4 rows in set (0.00 sec)

mysql> update employee set first_name='John' where emp_id=3;
Query OK, 1 row affected (0.02 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from employee;
+--------+------------+-----------+-------------------------+
| emp_id | first_name | last_name | email                   |
+--------+------------+-----------+-------------------------+
|      1 | John       | Doe       | jhon.doe@example.com    |
|      2 | jane       | Smith     | jane.smith@example.com  |
|      3 | John       | Doe       | john.doe@example.com    |
|      4 | Emily      | Davis     | Emily.davis@example.com |
+--------+------------+-----------+-------------------------+
4 rows in set (0.00 sec)

mysql> select first_name,count(first_name) from employee group by first_name having
count(first_name)>1;
+------------+-------------------+
| first_name | count(first_name) |
+------------+-------------------+
| John       |                 2 |
+------------+-------------------+
1 row in set (0.01 sec)

mysql> select first_name,last_name,count(*) as count from employee group by
first_name,last_name having count(first_name
,last_name)>1;
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to
your MySQL server version for the right syntax to use near ',last_name)>1' at line 1
mysql> select first_name,last_name,count(*) as count from employee group by
first_name,last_name having count(*)>1;
+------------+-----------+-------+
| first_name | last_name | count |
+------------+-----------+-------+
| John       | Doe       |     2 |
+------------+-----------+-------+
1 row in set (0.00 sec)

mysql> select email,count(email) as count from employee group by email having count(email)>1;
Empty set (0.00 sec)

mysql> update employee set email='john.doe@example.com' where emp_id=1;
Query OK, 1 row affected (0.02 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select email,count(email) as count from employee group by email having count(email)>1;
+----------------------+-------+
| email                | count |
+----------------------+-------+
| john.doe@example.com |     2 |
+----------------------+-------+
1 row in set (0.00 sec)
```

```
mysql> select first_name,email,count(*) as count from employee group by first_name,email having count(*)>1;
+------------+----------------------+-------+
| first_name | email                | count |
+------------+----------------------+-------+
| John       | john.doe@example.com |     2 |
+------------+----------------------+-------+
1 row in set (0.00 sec)
```