OVERVIEW OF JOIN TYPES

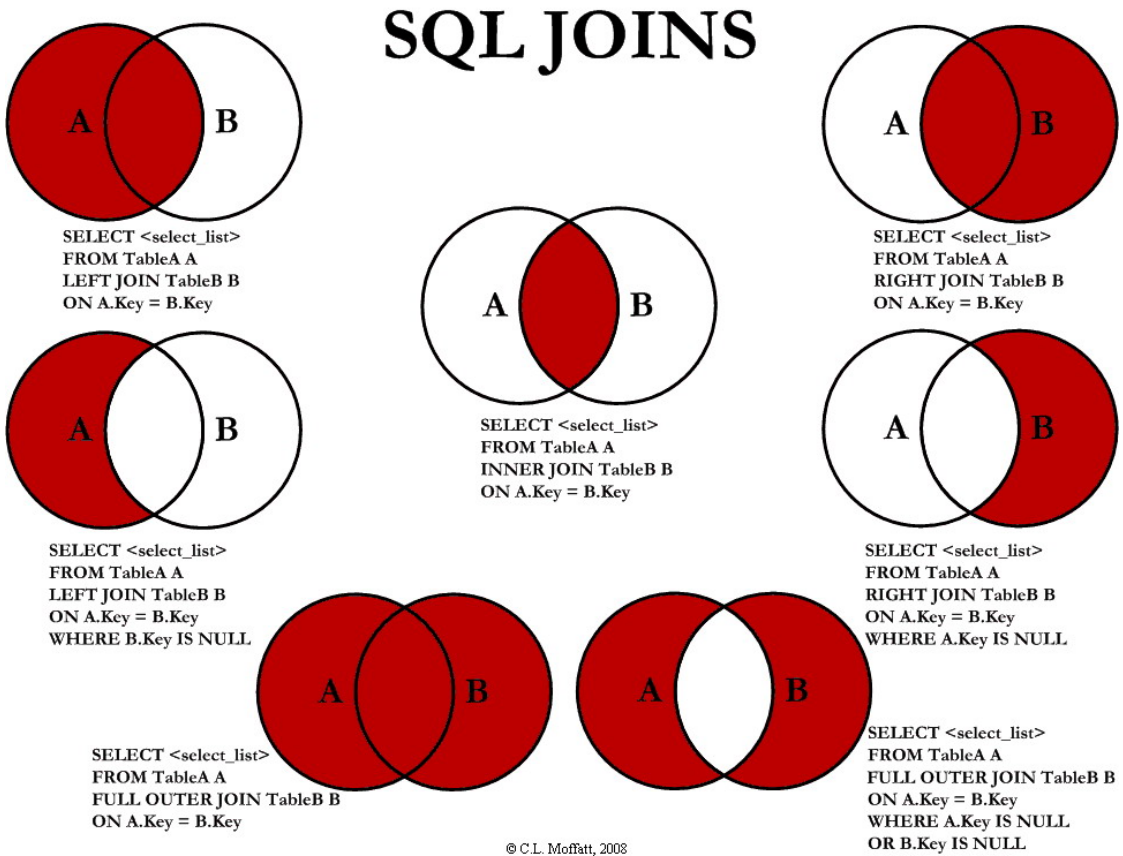
Overview

- This will begin to build our understanding of the various JOIN types.
- This presentation is a resource for you in this lecture!

JOINS Overview

- As we learn about more types of JOINS it will become very useful to reference a JOINS Venn Diagram figure.
- These are very easy to find via a Google Image Search!

JOINS Overview



JOINS Overview

- The example table for our discussion:

A

id	name
1	Pirate
2	Monkey
3	Ninja
4	Spaghetti

B

id	name
1	Rutabaga
2	Pirate
3	Darth Vader
4	Ninja

- Items in red are present in *both* tables.

Original Tables

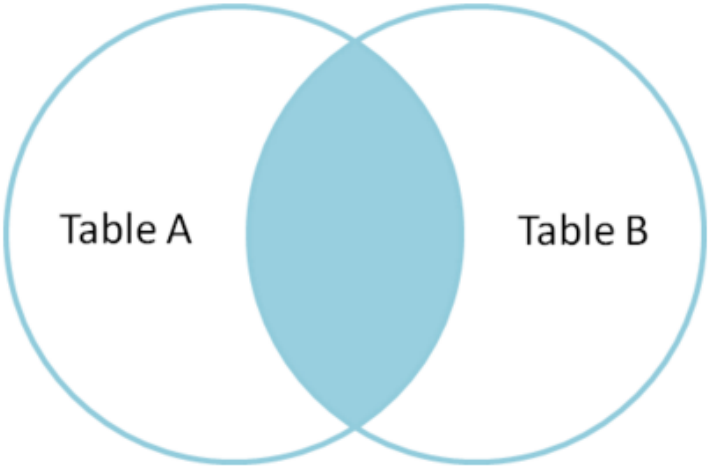
```
id name       id  name
-- ----       --  ----
1  Pirate     1   Rutabaga
2  Monkey     2   Pirate
3  Ninja      3   Darth Vader
4  Spaghetti  4   Ninja
```

INNER JOIN

Inner join produces only the set of records that match in both Table A and Table B.

```
SELECT * FROM TableA
INNER JOIN TableB
ON TableA.name = TableB.name

id  name       id   name
--  ----       --   ----
1   Pirate     2    Pirate
3   Ninja      4    Ninja
```



Original Tables

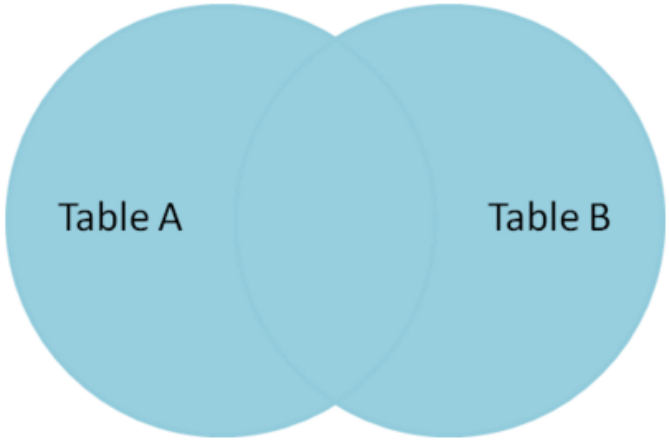
```
id name       id  name
-- ----       --  ----
1  Pirate     1   Rutabaga
2  Monkey     2   Pirate
3  Ninja      3   Darth Vader
4  Spaghetti  4   Ninja
```

FULL OUTER JOIN

Full outer join produces the set of all records in Table A and Table B, with matching records from both sides where available. If there is no match, the missing side will contain null.

```
SELECT * FROM TableA
FULL OUTER JOIN TableB
ON TableA.name = TableB.name

id    name       id    name
--    ----       --    ----
1     Pirate     2     Pirate
2     Monkey     null  null
3     Ninja      4     Ninja
4     Spaghetti  null  null
null  null       1     Rutabaga
null  null       3     Darth Vader
```



Original Tables

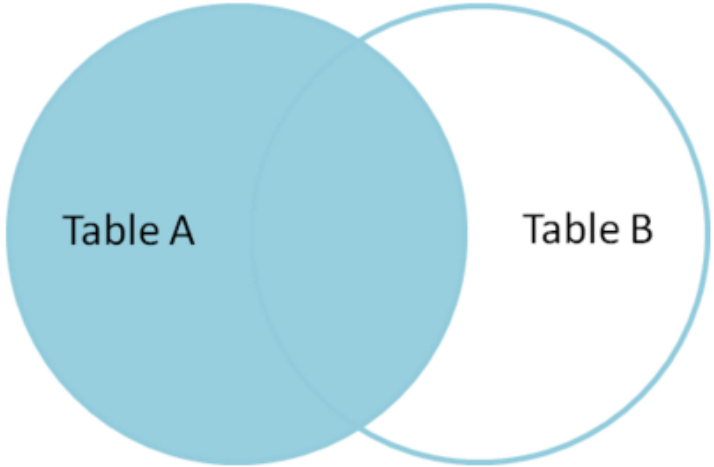
```
id name       id  name
-- ----       --  ----
1  Pirate     1   Rutabaga
2  Monkey     2   Pirate
3  Ninja      3   Darth Vader
4  Spaghetti  4   Ninja
```

LEFT OUTER JOIN

Left outer join produces a complete set of records from Table A, with the matching records (where available) in Table B. If there is no match, the right side will contain null.

```
SELECT * FROM TableA
LEFT OUTER JOIN TableB
ON TableA.name = TableB.name

id  name       id    name
--  ----       --    ----
1   Pirate     2     Pirate
2   Monkey     null  null
3   Ninja      4     Ninja
4   Spaghetti  null  null
```



Original Tables

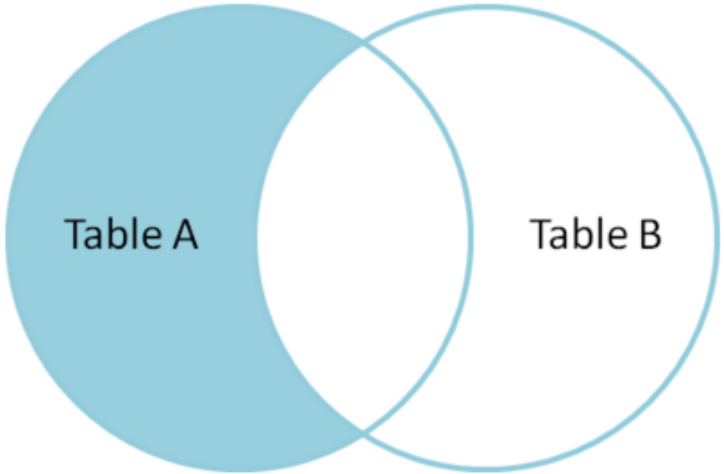
```
id name       id  name
-- ----       --  ----
1  Pirate     1   Rutabaga
2  Monkey     2   Pirate
3  Ninja      3   Darth Vader
4  Spaghetti  4   Ninja
```

LEFT OUTER JOIN with WHERE

To produce the set of records only in Table A, but not in Table B, we perform the same left outer join, then **exclude the records we don't want from the right side via a where clause**.

```
SELECT * FROM TableA
LEFT OUTER JOIN TableB
ON TableA.name = TableB.name
WHERE TableB.id IS null

id  name       id    name
--  ----       --    ----
2   Monkey     null  null
4   Spaghetti  null  null
```



Original Tables

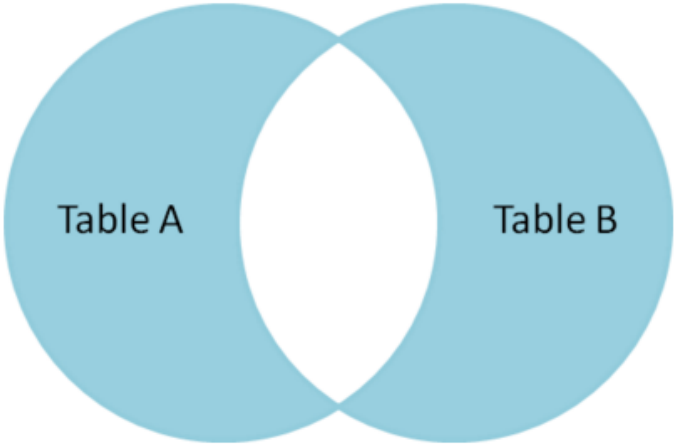
```
id name       id  name
-- ----       --  ----
1  Pirate     1   Rutabaga
2  Monkey     2   Pirate
3  Ninja      3   Darth Vader
4  Spaghetti  4   Ninja
```

```
SELECT * FROM TableA
FULL OUTER JOIN TableB
ON TableA.name = TableB.name
WHERE TableA.id IS null
OR TableB.id IS null
```

```
id    name       id    name
--    ----       --    ----
2     Monkey     null  null
4     Spaghetti  null  null
null  null       1     Rutabaga
null  null       3     Darth Vader
```

FULL OUTER JOIN with WHERE

To produce the set of records unique to Table A and Table B, we perform the same full outer join, then **exclude the records we don't want from both sides via a where clause**.



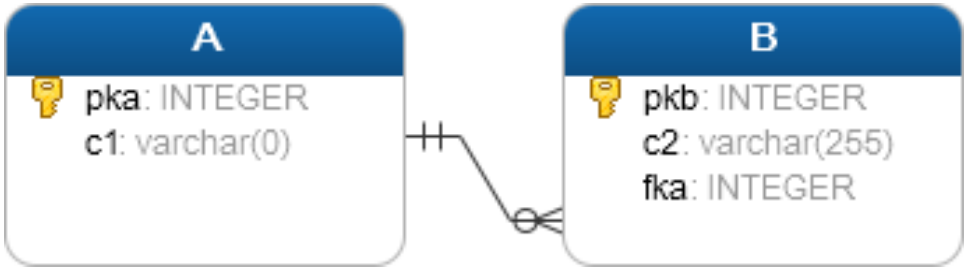
Review

- We’ve learned about the various JOIN types
- The next lectures will focus on showing examples of these various JOIN types.

LEFT JOIN

LEFT JOIN Statement

- Suppose we have two tables: A and B.



LEFT JOIN Statement

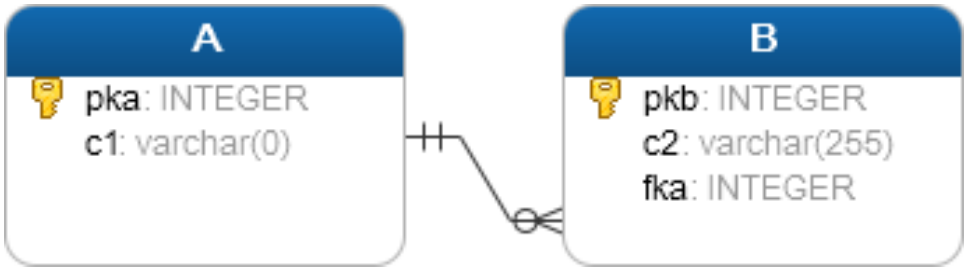
- The data in the B table relates to the data in the A table via the fka field.
- Each row in the A table may have zero or many corresponding rows in the B table.
- Each row in the B table has one and only one corresponding row in the A table.
- If you want to select rows from the A table that have corresponding rows in the B table, you use the INNER JOIN clause.

LEFT JOIN Statement

SELECT A.pka, A.c1,B.pkb,B.c2

FROM A

LEFT JOIN B **ON** A.pka = B.fka;



LEFT JOIN Statement

- To join the A table to the B table, you need to:
  - Specify the columns from which you want to select data in the SELECT clause.
  - Specify the left table i.e., A table where you want to get all rows, in the FROM clause.
  - Specify the right table i.e., B table in the LEFT JOIN clause. In addition, specify the condition for joining two tables.

LEFT JOIN Statement

- The LEFT JOIN clause returns all rows in the left table (A) that are combined with rows in the right table (B) even though there is no corresponding rows in the right table (B).
- The LEFT JOIN is also referred as LEFT OUTER JOIN.

LEFT JOIN Statement

- Let’s see an example of a LEFT JOIN!