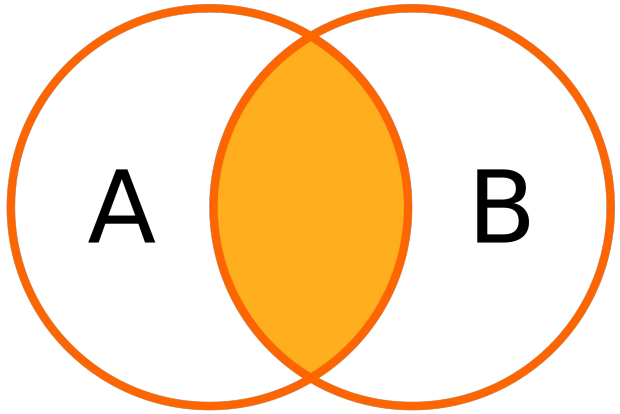
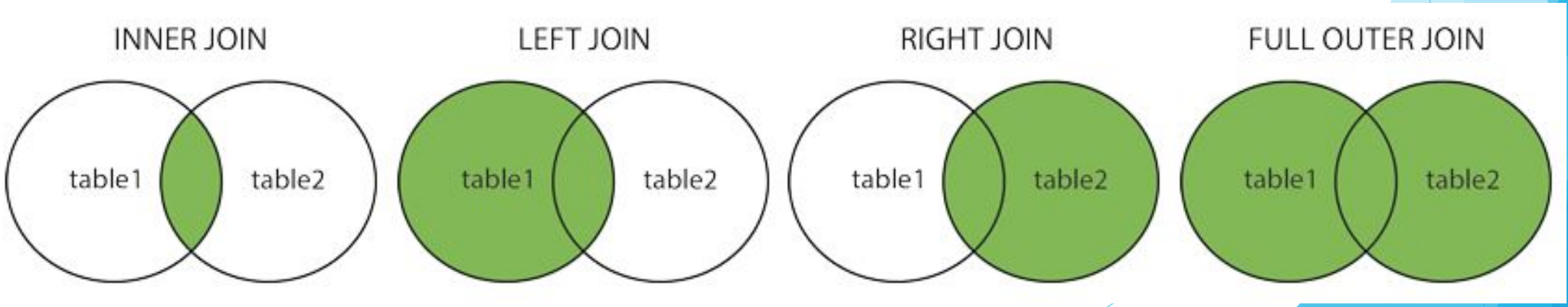
SQL JOINs

- So far we have learned how to retrieve data from a single table. Now we will learn how we can retrieve data from more tables into a single result.
- To combine columns from two or more tables into a single result based on a related column between them, we use a JOIN.
- The most common type of join is an INNER JOIN.



(COMMON) Types of JOINs

- **CROSS JOIN:** Returns all records from both tables, in all possible combinations.
- **(INNER) JOIN**: Returns records that have matching values in both tables
- **LEFT (OUTER) JOIN**: Return all records from the left table, and the matched records from the right table
- **RIGHT (OUTER) JOIN**: Return all records from the right table, and the matched records from the left table
- **FULL (OUTER) JOIN**: Return all records when there is a match in either left or right table

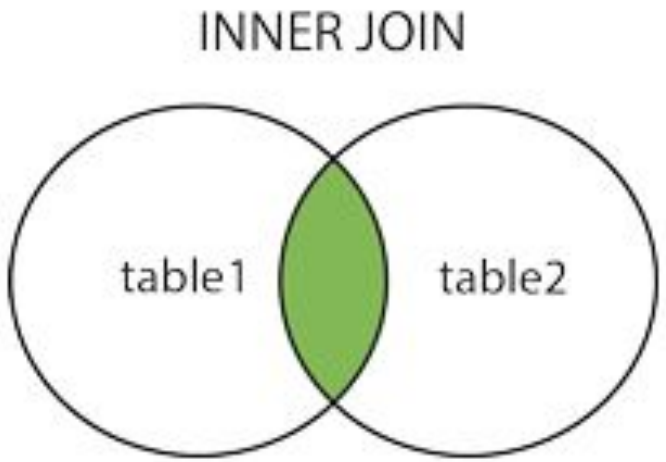


INNER JOIN

- The INNER JOIN keyword selects records that have matching values in both tables.

The explicit syntax for an inner join

```
SELECT select_list
FROM table_1
     [INNER] JOIN table_2
          ON join_condition_1
   [[INNER] JOIN table_3
          ON join_condition_2]..
```



An inner join of the Vendors and Invoices tables

```
SELECT InvoiceNumber, VendorName
FROM Vendors JOIN Invoices
    ON Vendors.VendorID = Invoices.VendorID;
```

	InvoiceNumber	VendorName
1	QP58872	IBM
2	Q545443	IBM
3	547481328	Blue Cross
4	547479217	Blue Cross
5	547480102	Blue Cross
6	P02-88D77S7	Fresno County Tax Collector
7	40318	Data Reproductions Corp

```
(114 rows)
```

- Note: The INNER keyword is optional.

Alias Names/Correlation Names

The syntax for an inner join that uses correlation names

```
SELECT select_list
FROM table_1 [AS] n1
    [INNER] JOIN table_2 [AS] n2
        ON n1.column_name operator n2.column_name
   [[INNER] JOIN table_3 [AS] n3
        ON n2.column_name operator n3.column_name]...
```

- You can give your table names aliases in your from clause. You do it the same way give aliases in the SELECT clause…you use the AS keyword.
- If you use a correlation name, you must use that correlation name anywhere you are referencing the table in your SELECT.

Be careful...

Correlation names that make the query more difficult to read

```
SELECT InvoiceNumber, VendorName, InvoiceDueDate,
    InvoiceTotal - PaymentTotal - CreditTotal AS BalanceDue
FROM Vendors AS v INNER JOIN Invoices AS i
    ON v.VendorID = i.VendorID
WHERE InvoiceTotal - PaymentTotal - CreditTotal > 0
ORDER BY InvoiceDueDate DESC;
```

The result set

	InvoiceNumber	VendorName	InvoiceDueDate	BalanceDue
1	0-2436	Malloy Lithographing Inc	2016-04-30 00:00:00	10976.06
2	547480102	Blue Cross	2016-04-30 00:00:00	224.00
3	9982771	Ford Motor Credit Company	2016-04-23 00:00:00	503.20

(11 rows)

A correlation name that simplifies the query

```
SELECT InvoiceNumber, InvoiceLineItemAmount,
InvoiceLineItemDescription
FROM Invoices INNER JOIN InvoiceLineItems AS LineItems
    ON Invoices.InvoiceID = LineItems.InvoiceID
WHERE AccountNo = 540
ORDER BY InvoiceDate;
```

The result set

	InvoiceNumber	InvoiceLineItemAmount	InvoiceLineItemDescription
1	I77271-O01	478.00	Publishers Marketing
2	972110	207.78	Prospect list
3	133560	175.00	Card deck advertising

(6 rows)

An inner join with two conditions

```
SELECT InvoiceNumber, InvoiceDate,
    InvoiceTotal, InvoiceLineItemAmount
FROM Invoices INNER JOIN InvoiceLineItems AS LineItems
    ON (Invoices.InvoiceID = LineItems.InvoiceID) AND
        (Invoices.InvoiceTotal >
            LineItems.InvoiceLineItemAmount)
ORDER BY InvoiceNumber;
```

The result set

	InvoiceNumber	InvoiceDate	InvoiceTotal	InvoiceLineItemAmount
1	97/522	2016-02-28 00:00:00	1962.13	1197.00
2	97/522	2016-02-28 00:00:00	1962.13	765.13
3	I77271-001	2015-12-26 00:00:00	662.00	50.00
4	I77271-001	2015-12-26 00:00:00	662.00	75.60
5	I77271-001	2015-12-26 00:00:00	662.00	58.40
6	I77271-001	2015-12-26 00:00:00	662.00	478.00

- Note: This is also called a compound join. Best practice is usually to put any filter expressions in the WHERE instead of the ON.

The same join with the second condition coded in a WHERE clause

```
SELECT InvoiceNumber, InvoiceDate,
    InvoiceTotal, InvoiceLineItemAmount
FROM Invoices INNER JOIN InvoiceLineItems AS LineItems
    ON Invoices.InvoiceID = LineItems.InvoiceID
WHERE Invoices.InvoiceTotal >
    LineItems.InvoiceLineItemAmount
ORDER BY InvoiceNumber;
```

The same result set

	InvoiceNumber	InvoiceDate	InvoiceTotal	InvoiceLineItemAmount
1	97/522	2016-02-28 00:00:00	1962.13	1197.00
2	97/522	2016-02-28 00:00:00	1962.13	765.13
3	I77271-001	2015-12-26 00:00:00	662.00	50.00
4	I77271-001	2015-12-26 00:00:00	662.00	75.60
5	I77271-001	2015-12-26 00:00:00	662.00	58.40
6	I77271-001	2015-12-26 00:00:00	662.00	478.00