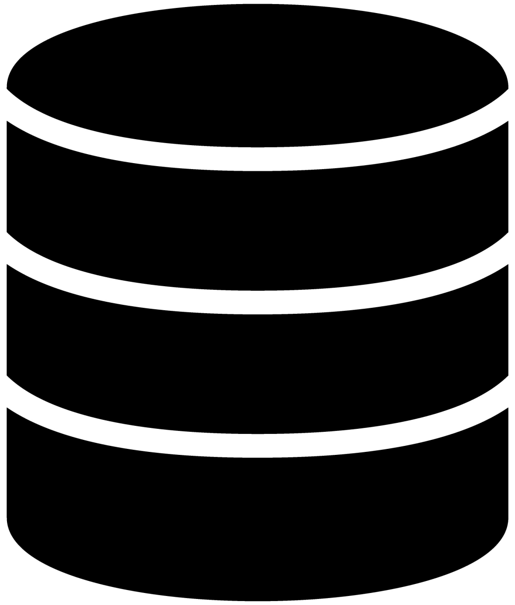
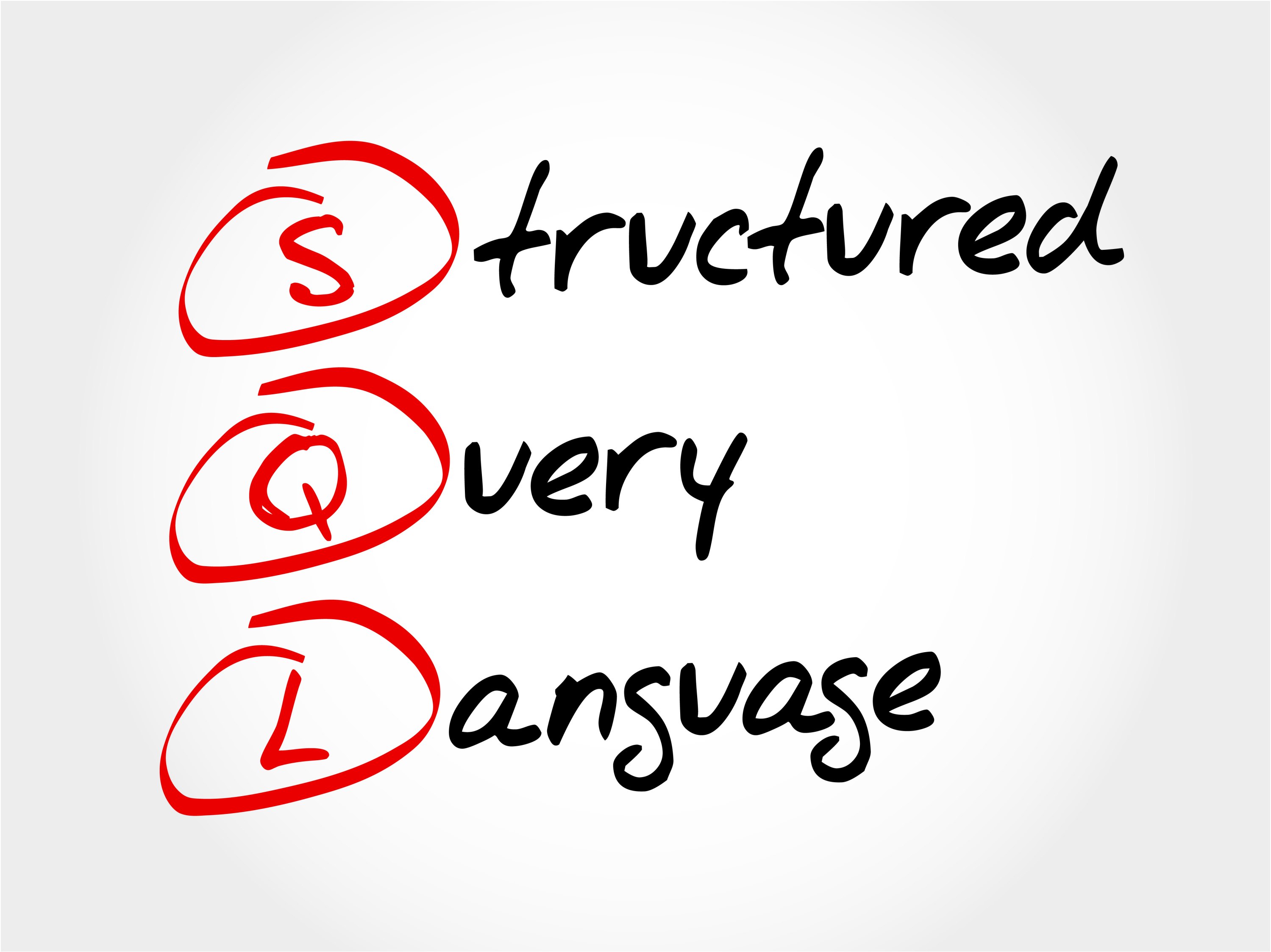
Welcome

INTRODUCTION TO SQL SERVER

John MacKintosh
Instructor




Structured
Query
Language

SQL Server & Transact-SQL

- SQL Server - relational database system developed by Microsoft
- Transact-SQL (T-SQL) - Microsoft's implementation of SQL, with additional functionality
- In this course: Master the fundamentals of T-SQL
- Learn how to write queries

LET'S GET
STARTED

Querying 101

- SQL-Server: the *store* containing databases and tables
- Queries: how we *pick* different items, from different aisles, and load up our cart
- `SELECT` : key term for retrieving data

```sql
SELECT description
FROM grid;
```

```
+-------------------------------------+
| description                         |
|-------------------------------------|
| Severe Weather Thunderstorms        |
| Severe Weather Thunderstorms        |
| Severe Weather Thunderstorms        |
| Fuel Supply Emergency Coal          |
| Physical Attack Vandalism           |
| Physical Attack Vandalism           |
| Physical Attack Vandalism           |
| Severe Weather Thunderstorms        |
| Severe Weather Thunderstorms        |
| Suspected Physical Attack           |
| Physical Attack Vandalism           |
| ...                                 |
+-------------------------------------+
```

Selecting more than one column

```
SELECT
  artist_id,
  artist_name
FROM
  artist;
```

```
+-----------+----------------------+
| artist_id | artist_name          |
|-----------+----------------------|
| 1         | AC/DC                |
| 2         | Accept               |
| 3         | Aerosmith            |
| 4         | Alanis Morissette    |
| 5         | Alice In Chains      |
| 6         | Antônio Carlos Jobim |
| 7         | Apocalyptica         |
| 8         | Audioslave           |
| 9         | BackBeat             |
| 10        | Billy Cobham         |
+-----------+----------------------+
```

Query formatting

```
SELECT description, event_year, event_date
FROM grid;
```

```
SELECT
  description,
  event_year,
  event_date
FROM
  grid;
```

Select TOP ()

```sql
-- Return 5 rows
SELECT TOP(5) artist
FROM artists;
```

```sql
-- Return top 5% of rows
SELECT TOP(5) PERCENT artist
FROM artists;
```

```
+------------------------+
| artist                 |
|------------------------|
| AC/DC                  |
| Accept                 |
| Aerosmith              |
| Alanis Morissette      |
| Alice in Chains        |
+------------------------+
```

Select DISTINCT

```sql
-- Return all rows in the table
SELECT nerc_region
FROM grid;
```

```
+-------------+
| nerc_region |
|-------------|
| RFC         |
| RFC         |
| MRO         |
| MRO         |
| ....        |
+-------------+
```

```sql
-- Return unique rows
SELECT DISTINCT nerc_region
FROM grid;
```

```
+-------------+
| nerc_region |
|-------------|
| NPCC        |
| NPCC RFC    |
| RFC         |
| ERCOT       |
| ...         |
+-------------+
```

Select *

```
-- Return all rows
SELECT *
FROM grid;
```

- **NOT** suitable for large tables

Aliasing column names with AS

```
SELECT demand_loss_mw AS lost_demand
FROM grid;
```

```
+-------------+
| lost_demand |
|-------------|
| 424         |
| 217         |
| 494         |
| 338         |
| 3900        |
| 3300        |
+-------------+
```

```
SELECT description AS cause_of_outage
FROM grid;
```

```
+-------------------------------+
| cause_of_outage               |
|-------------------------------|
| Severe Weather Thunderstorms  |
| Fuel Supply Emergency Coal    |
| Physical Attack Vandalism     |
| Suspected Physical Attack     |
| Electrical System Islanding   |
+-------------------------------+
```

Let's write some T-SQL!

INTRODUCTION TO SQL SERVER

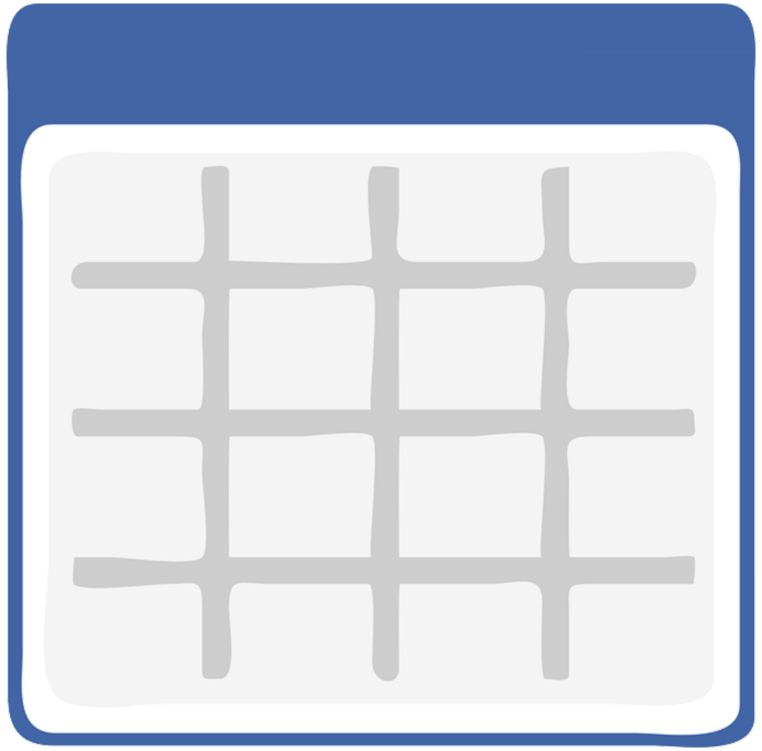
Ordering and Filtering

INTRODUCTION TO SQL SERVER

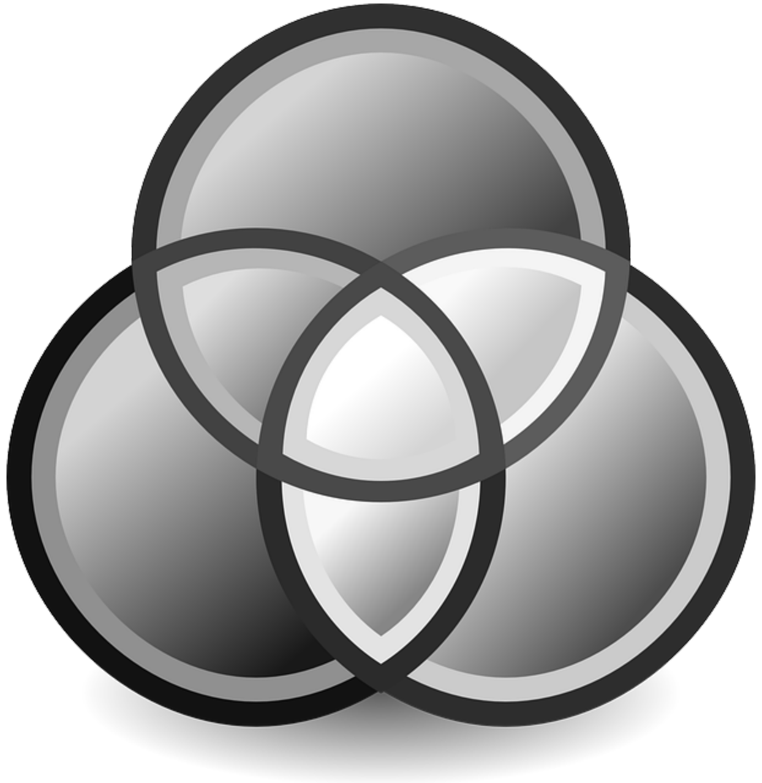
John MacKintosh
Instructor

Order! Order!

- Tables comprise of rows and columns
- Queries return *sets*, or *subsets*



- Sets have no inherent order
- If order is important, use `ORDER BY`



```
SELECT TOP (10) prod_id, year_intro
FROM products
-- Order in ascending order
ORDER BY year_intro, product_id;
```

```
+------------+-------------+
| product_id | year_intro  |
|------------+-------------|
| 36         |  1981       |
| 37         |  1982       |
| 38         |  1983       |
| 39         |  1984       |
| 40         |  1984       |
| 41         |  1984       |
| 52         |  1985       |
| 43         |  1986       |
| 44         |  1987       |
| 54         |  1987       |
+------------+-------------+
SELECT TOP (10) product_id, year_intro
FROM products
-- Order year_intro in descending order
ORDER BY year_intro DESC, product_id;
```

```
+------------+-------------+
| product_id | year_intro  |
|------------+-------------|
| 158        |  2015       |
| 173        |  2015       |
| 170        |  2014       |
| 171        |  2014       |
| 172        |  2014       |
| 144        |  2013       |
| 146        |  2013       |
| 147        |  2013       |
| 148        |  2013       |
| 149        |  2013       |
+------------+-------------+
SELECT
  TOP (10) channels,
  year_intro
FROM products
-- Order in different directions
ORDER BY
  year_intro DESC,
  channels;
```

```
SELECT
  TOP (10) channels,
  year_intro
FROM products
-- Both columns in descending order
ORDER BY
  year_intro DESC,
  channels DESC;
```

```
+-------------+------------+
| channels    | year_intro |
|-------------+------------|
| 35          | 2015       |
| 74          | 2015       |
| 29          | 2014       |
| 45          | 2014       |
| 48          | 2014       |
| 12          | 2013       |
| 13          | 2013       |
| 14          | 2013       |
| 22          | 2013       |
| 24          | 2013       |
+-------------+------------+
```

```
+-------------+------------+
| channels    | year_intro |
|-------------+------------|
| 74          | 2015       |
| 35          | 2015       |
| 48          | 2014       |
| 45          | 2014       |
| 29          | 2014       |
| 837         | 2013       |
| 642         | 2013       |
| 561         | 2013       |
| 491         | 2013       |
| 198         | 2013       |
+-------------+------------+
```

```sql
SELECT city_id, name_alias
FROM invoice
-- Ordering text (Ascending order)
ORDER BY name_alias;
```

```
+-------------+----------------+
| city_id     | name_alias     |
|-------------+----------------|
|  48         | Amsterdam      |
|  59         | Bangalore      |
|  36         | Berlin         |
|  38         | Berlin         |
|  42         | Bordeaux       |
|  23         | Boston         |
|  13         | Brasília       |
|   8         | Brussels       |
|  45         | Budapest       |
|  56         | Buenos Aires   |
+-------------+----------------+
```

```sql
SELECT city_id, name_alias
FROM invoice
-- Ordering text (Descending order)
ORDER BY name_alias DESC;
```

```
+-------------+----------------+
| city_id     | name_alias     |
|-------------+----------------|
|  33         | Yellowknife    |
|  32         | Winnipeg       |
|  49         | Warsaw         |
|   7         | Vienne         |
|  15         | Vancouver      |
|  27         | Tucson         |
|  29         | Toronto        |
|   2         | Stuttgart      |
|  51         | Stockholm      |
|  55         | Sydney         |
+-------------+----------------+
```

What if we only wanted to return rows that met certain criteria?

```sql
SELECT customer_id, total
FROM invoice
WHERE total > 15;
```

First 3 customers with invoice value > 15

```
+-------------+------------+
| customer_id | total      |
|-------------+------------|
| 57          | 17.91      |
|  7          | 18.86      |
| 45          | 21.86      |
+-------------+------------+
-- Rows with points greater than 10
WHERE points > 10

-- Rows with points less than 10
WHERE points  < 10

-- Rows with points greater than or equal to 10
WHERE points >= 10

-- Rows with points less than or equal to 20
WHERE points  <= 20

-- Character data type
WHERE  country =  'Spain'

-- Date data type
WHERE  event_date =  '2012-01-02'
```

```
SELECT customer_id, total
FROM invoice
-- Testing for non-equality
WHERE total <> 10;
```

```
+------------+-------+
| customerid | total |
|------------+-------|
| 2          | 1.98  |
| 4          | 3.96  |
| 8          | 5.94  |
| 14         | 8.91  |
| 23         | 13.86 |
| 37         | 0.99  |
+------------+-------+
```

Between

```sql
SELECT customer_id, total
FROM invoice
WHERE total BETWEEN 20 AND 30;
```

```
+------------+--------+
| customerid | total  |
|------------+--------|
| 45         | 21.86  |
| 46         | 21.86  |
| 26         | 23.86  |
| 6          | 25.86  |
+------------+--------+
```

```sql
SELECT customer_id, total
FROM invoice
WHERE total NOT BETWEEN 20 AND 30;
```

```
+------------+--------+
| customerid | total  |
|------------+--------|
|  2         | 1.98   |
|  4         | 3.96   |
|  8         | 5.94   |
| 14         | 8.91   |
+------------+--------+
```

What is NULL?

- NULL indicates there is no value for that record
- NULLs help highlight gaps in our data

```
SELECT
  TOP (6) total,
  billing_state
FROM invoice
WHERE billing_state IS NULL;
```

```
SELECT
  TOP (6) total,
  billing_state
FROM invoice
WHERE billing_state IS NOT NULL;
```

```
+-------+---------------+
| total | billing_state |
|-------+---------------|
| 1.98  | NULL          |
| 3.96  | NULL          |
| 5.94  | NULL          |
| 0.99  | NULL          |
| 1.98  | NULL          |
| 1.98  | NULL          |
+-------+---------------+
```

```
+--------+---------------+
| total  | billing_state |
|--------+---------------|
|  8.91  | AB            |
| 13.96  | MA            |
|  5.94  | Dublin        |
|  0.99  | CA            |
|  1.98  | WA            |
|  1.98  | CA            |
+-------+----------------+
```

Let's sort it!

INTRODUCTION TO SQL SERVER

WHERE the wild things are

INTRODUCTION TO SQL SERVER

John MacKintosh
Instructor

```
SELECT song, artist
FROM songlist
WHERE
  artist = 'AC/DC';
```

```
+--------------------------+--------+
| song                     | artist |
|--------------------------+--------|
| Baby, Please Don't Go    | AC/DC  |
| Back In Black            | AC/DC  |
| Big Gun                  | AC/DC  |
| CAN'T STOP ROCK'N'ROLL   | AC/DC  |
| Girls Got Rhythm         | AC/DC  |
| Hard As A Rock           | AC/DC  |
| Have a Drink On Me       | AC/DC  |
| Hells Bells              | AC/DC  |
+--------------------------+--------+
```

```
SELECT song, artist
FROM songlist
WHERE
  artist = 'AC/DC'
  AND release_year < 1980;
```

```
+------------------------------+--------+
| song                         | artist |
|------------------------------+--------|
| Dirty Deeds Done Dirt Cheap  | AC/DC  |
| Highway To Hell              | AC/DC  |
| It's A Long Way To The Top   | AC/DC  |
| Let There Be Rock            | AC/DC  |
| Night Prowler                | AC/DC  |
| T.N.T.                       | AC/DC  |
| Touch Too Much               | AC/DC  |
| Whole Lotta Rosie            | AC/DC  |
+------------------------------+--------+
```

AND again

- Returns 3 rows:

```
SELECT *
FROM songlist
WHERE
  release_year = 1994
  AND artist = 'Green Day';
```

- Returns 1 row:

```
SELECT *
FROM songlist
WHERE
  release_year = 1994
  AND artist = 'Green Day'
  AND song = 'Basket Case';
```

```sql
SELECT
  song,
  artist,
  release_year
FROM songlist
WHERE release_year = 1994;
```

```
+----------------------+---------------------+--------------+
| song                 | artist              | release_year |
|----------------------+---------------------+---------------
| Black Hole Sun       | Soundgarden         | 1994         |
| Fell On Black Days   | Soundgarden         | 1994         |
| Spoonman             | Soundgarden         | 1994         |
| Big Empty            | Stone Temple Pilots | 1994         |
| Interstate Love Song | Stone Temple Pilots | 1994         |
| Vasoline             | Stone Temple Pilots | 1994         |
+----------------------+---------------------+--------------+
```

```sql
SELECT
  song,
  artist,
  release_year
FROM songlist
WHERE
  release_year = 1994
  OR release_year > 2000;
```

```
+----------------------+---------------------+--------------+
| song                 | artist              | release_year |
|----------------------+---------------------+---------------
| Doom And Gloom       | Rolling Stones      | 2012         |
| Remedy               | Seether             | 2005         |
| 45                   | Shinedown           | 2003         |
| Black Hole Sun       | Soundgarden         | 1994         |
| Fell On Black Days   | Soundgarden         | 1994         |
| Spoonman             | Soundgarden         | 1994         |
| It's Been Awhile     | Staind              | 2001         |
| Big Empty            | Stone Temple Pilots | 1994         |
| Interstate Love Song | Stone Temple Pilots | 1994         |
| Vasoline             | Stone Temple Pilots | 1994         |
+----------------------+---------------------+--------------+
```

```sql
SELECT song
FROM songlist
WHERE
  artist = 'Green Day'
  AND release_year = 1994;
```

```
+--------------------+
| song               |
|--------------------|
| Basket Case        |
| Longview           |
| When I Come Around |
+--------------------+
```

```sql
SELECT song
FROM songlist
WHERE
  artist = 'Green Day'
  AND release_year > 2000;
```

```
+----------------------------+
| song                       |
|----------------------------|
| Boulevard Of Broken Dreams |
| Holiday (Live)             |
| Holiday                    |
+----------------------------+
```

```sql
SELECT song
FROM songlist
WHERE
  artist = 'Green Day'
  AND release_year = 1994
  OR release_year > 2000;
```

```
+-------------------------------------+
| song                                |
|-------------------------------------|
| Doom And Gloom                      |
| Remedy                              |
| 45                                  |
| It's Been Awhile                    |
| Goodbye Daughters of the Revolution |
| Gold On The Ceiling                 |
| Lonely Boy                          |
| Seven Nation Army                   |
| Get Together                        |
| Vertigo                             |
| When I'm Gone                       |
| ...                                 |
| ...                                 |
+-------------------------------------+
```

What went wrong?

```
SELECT *
FROM songlist
WHERE
  artist = 'Green Day'
  AND release_year = 1994
  OR release_year > 2000;
```

```
SELECT *
FROM songlist
WHERE
  artist = 'Green Day'
  AND release_year = 1994;
```

OR

```
SELECT *
FROM songlist
WHERE
  release_year > 2000;
```

```sql
SELECT song
FROM songlist
WHERE
  artist = 'Green Day'
  AND (
    release_year = 1994
    OR release_year > 2000
  );
```

```
+--------------------------------------+
| song                                 |
|--------------------------------------|
| Basket Case                          |
| Boulevard Of Broken Dreams           |
| Holiday (Live)                       |
| Holiday / Boulevard of Broken Dreams |
| Longview                             |
| When I Come Around                   |
+--------------------------------------+
```

Another way of writing the query:

```sql
SELECT song
FROM songlist
WHERE
  (
    artist = 'Green Day'
    AND release_year = 1994
  )
  OR (
    artist = 'Green Day'
    AND release_year > 2000
  );
```

```sql
SELECT song, artist
FROM songlist
WHERE
  artist IN ('Van Halen', 'ZZ Top')
ORDER BY song;
```

```
+-----------------------------------+-----------+
| song                              | artist    |
|-----------------------------------+-----------|
| (Oh) Pretty Woman                 | Van Halen |
| 1984/jump                         | Van Halen |
| A Fool for Your Stockings         | ZZ Top    |
| Ain't Talkin' 'bout Love          | Van Halen |
| And the Cradle Will Rock...       | Van Halen |
| Arrested For Driving While Blind  | ZZ Top    |
| Atomic Punk                       | Van Halen |
+-----------------------------------+-----------+
```

```sql
SELECT song, release_year
FROM songlist
WHERE
  release_year IN (1985, 1991, 1992);
```

```
+-------------------------+--------------+
| song                    | release_year |
|-------------------------+--------------|
| Addicted to Love        | 1985         |
| Don't You               | 1985         |
| Come As You Are         | 1991         |
| Money for Nothing       | 1985         |
| Walk of Life            | 1985         |
| Man On the Moon         | 1992         |
| Breaking the Girl       | 1992         |
| You Belong to the City  | 1985         |
| Enter Sandman           | 1991         |
| In Bloom                | 1991         |
+-------------------------+--------------+
```

```sql
SELECT song
FROM songlist
WHERE song LIKE 'a%';
```

```
+-------------------------+
| song                    |
|-------------------------|
| Addicted to Love        |
| Ain't Too Proud to Beg  |
+-------------------------+
```

```sql
SELECT artist
FROM songlist
WHERE artist LIKE 'f%';
```

```
+----------------+
| artist         |
|----------------|
| Faces          |
| Faith No More  |
+----------------+
```

Let's practice!

INTRODUCTION TO SQL SERVER