Introduction to Analytics

LocalTek Healthcare Analytics

NASHVILLE SOFTWARE SCHOOL

What is a Relational Database?

- Relational databases store related data in **tabular form**
- Relationships between tables are defined with **keys** (primary and foreign)
- Most relational databases are created for use by applications and optimized for **CRUD *transactions***
  - Create
  - Read
  - Update
  - Delete
- Data Analysts/Data Scientists mostly focused on **Read** - querying the data. They want to look at the data and perform calculations without making any permanent changes.

NASHVILLE SOFTWARE SCHOOL

What is SQL

SQL stands **S**tructured **Q**uery **L**anguage. It is the language you use to interact with a database. It allows you to write out what you want to search for, goes to a database that you specify, then returns those results to you.

The Structured part of SQL means queries are written in a specific format using specific keywords.

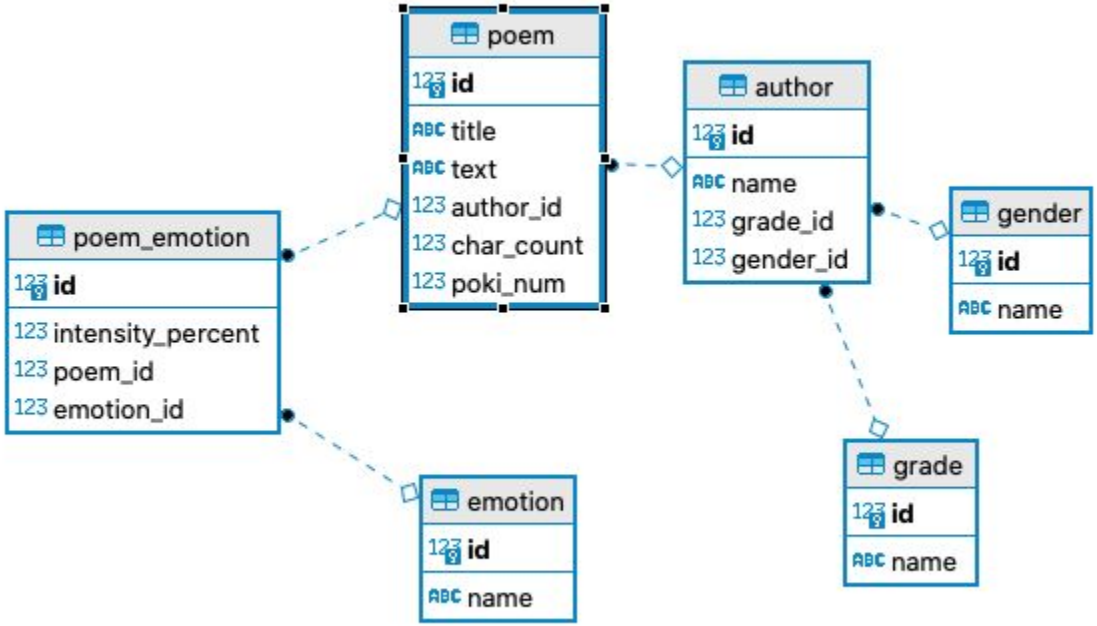
NASHVILLE SOFTWARE SCHOOL

Keywords

- **SELECT**
- **FROM**
- AS
- **LIMIT**
- DISTINCT
- **COUNT**
- **WHERE**
- AND
- OR
- BETWEEN
- IN
- (NOT) NULL
- LIKE
- **AVG(), SUM(), MAX(), MIN(), etc.**
- **ORDER BY**
- **GROUP BY**
- HAVING

NASHVILLE SOFTWARE SCHOOL

For our learning examples, we will be using the PoetryKids database. Below is an Entity Relationship Diagram (ERD) that shows the tables in the database and the relationships between tables.



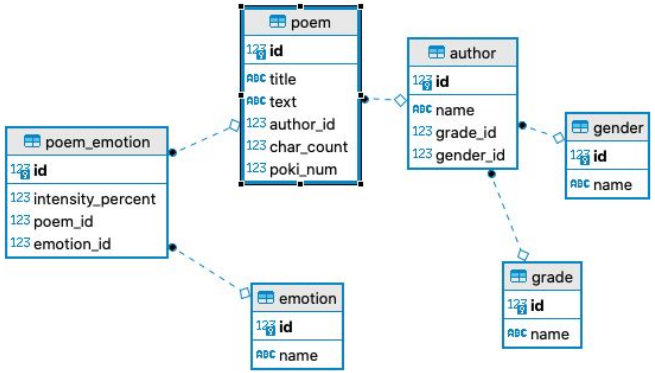
NASHVILLE SOFTWARE SCHOOL

Format of a Query

Keywords all caps

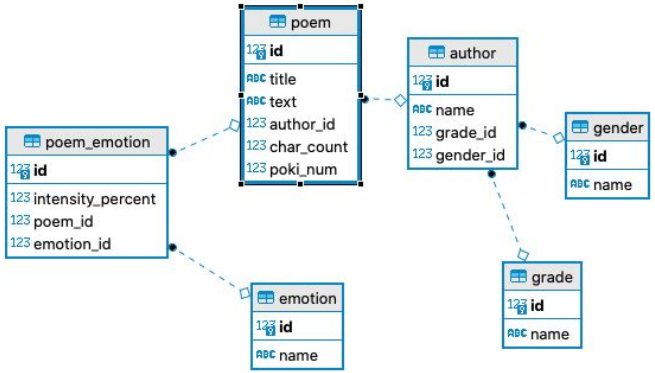
```
SELECT text
FROM poem;
```

End with semicolon



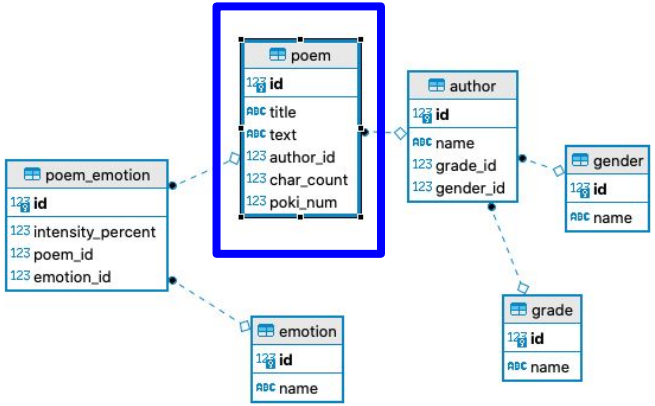
Format of a Query

the column(s)
to retrieve




```
SELECT text
FROM poem;
```

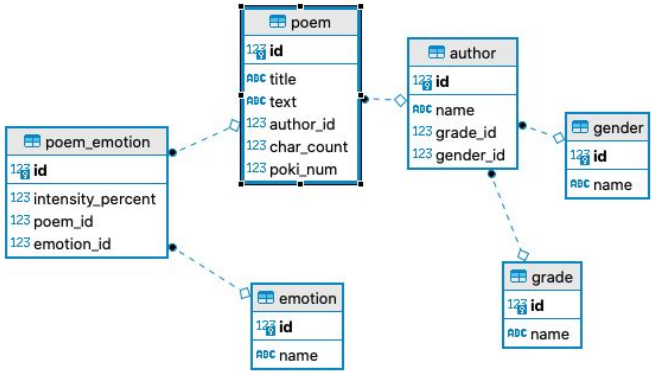
Format of a Query




```
SELECT text
FROM poem;
```

the table to retrieve from

Format of a Query

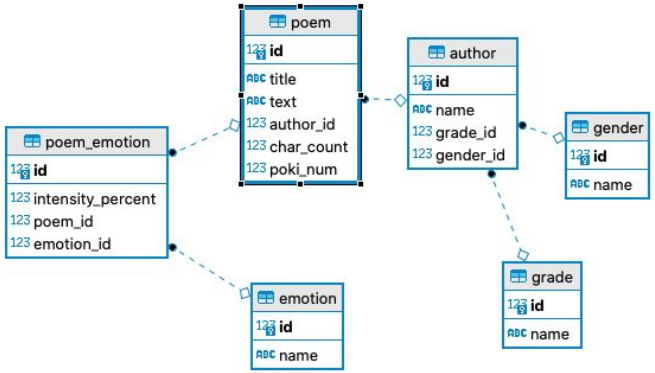



```
SELECT text
FROM poem
WHERE char_count < 20;
```

Only return rows satisfying this condition.

NASHVILLE SOFTWARE SCHOOL

Format of a Query




```
SELECT text
FROM poem
WHERE char_count < 20
LIMIT 5;
```

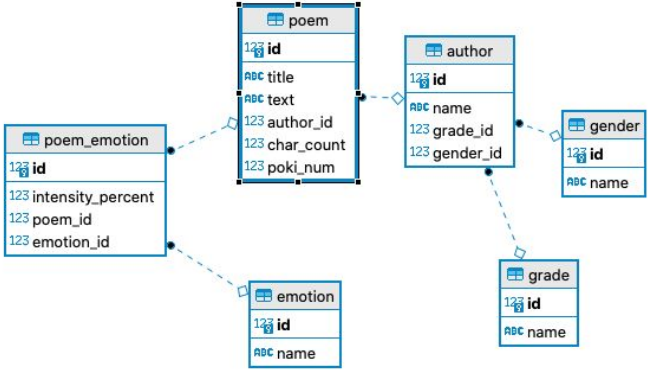
Just return the first
5 results.

Selecting single columns from a table

You can also specify individual columns to return, each separated by a ‘,’:

```
SELECT name, grade_id
FROM author
LIMIT 5;
```

	name character varying	grade_id integer
1	a	1
2	aab	1
3	aadhya	1
4	aaliyah	1
5	aanna	1

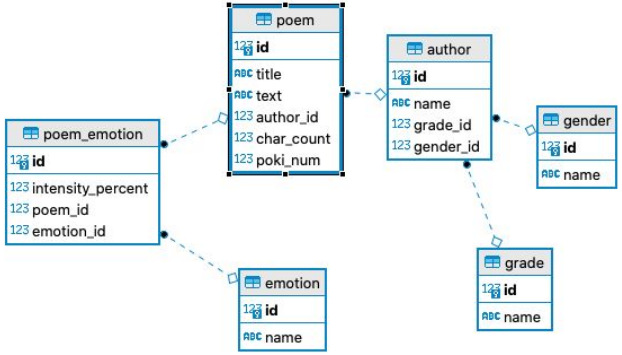


Selecting all columns from a table

A shorthand to **SELECT ALL** is to use a *:

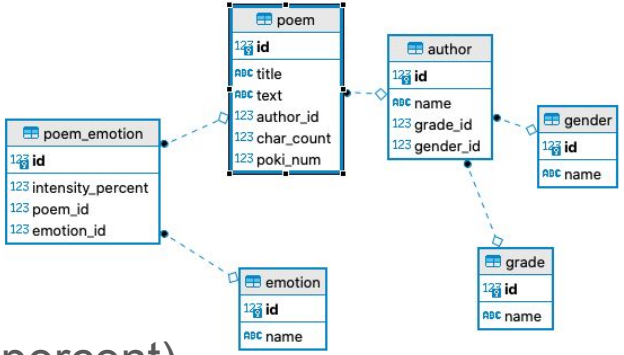
```
SELECT *
FROM emotion;
```

	id [PK] integer	name character varying
1	1	Anger
2	2	Fear
3	3	Sadness
4	4	Joy



Aggregating Data

You may want to summarize your data in different ways. For example you could **COUNT**, **SUM**, **AVERAGE**, or find the **MAX** or **MIN**:




```
SELECT COUNT(id)
FROM author
WHERE grade_id = 4;
```

	count bigint
1	3288

```
SELECT AVG(intensity_percent)
FROM poem_emotion
WHERE emotion_id = 2;
```

	avg numeric
1	45.4740880030086499

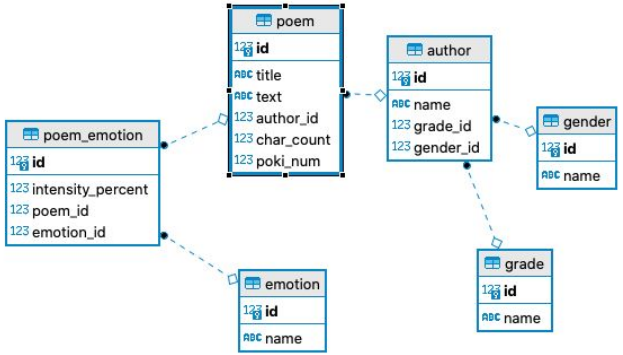
NASHVILLE SOFTWARE SCHOOL

Aggregating Data with GROUP BY

The **GROUP BY** keyword will subdivide the table based on the specified columns. You can then perform aggregations on the subgroups:

```
SELECT COUNT(id)
FROM author
GROUP BY grade_id;
```

	count bigint
1	623
2	2344
3	3464
4	3288
5	1437



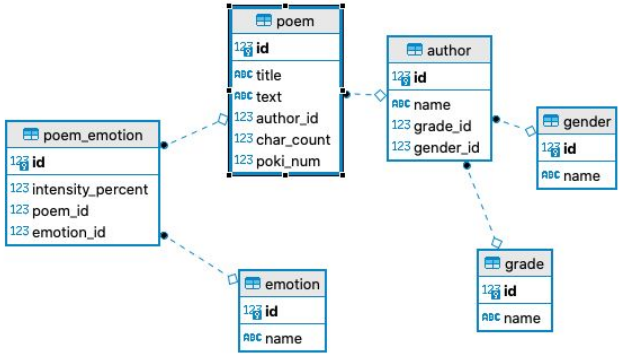
NASHVILLE SOFTWARE SCHOOL

Aggregating Data with GROUP BY

The **GROUP BY** keyword will subdivide the table based on the specified columns. You can then perform aggregations on the subgroups:

```
SELECT grade_id, COUNT(id)
FROM author
GROUP BY grade_id;
```

	grade_id integer	count bigint
1	1	623
2	3	2344
3	5	3464
4	4	3288
5	2	1437



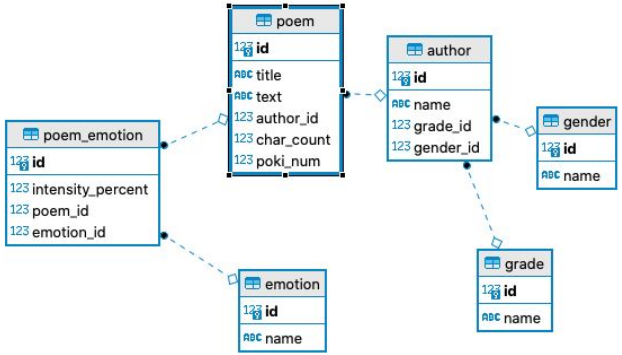
NASHVILLE SOFTWARE SCHOOL

Aggregating Data with GROUP BY

The **GROUP BY** keyword will subdivide the table based on the specified columns. You can then perform aggregations on the subgroups:

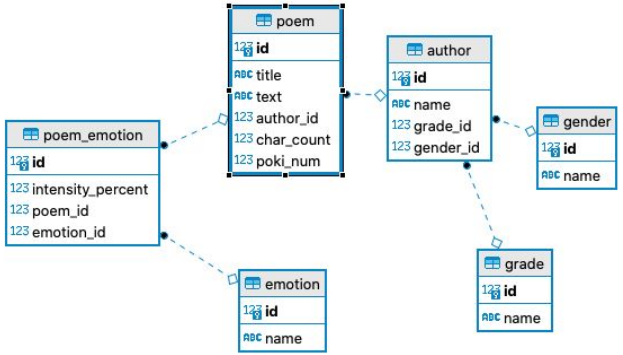
```
SELECT grade_id, COUNT(id)
FROM author
GROUP BY grade_id
ORDER BY grade_id;
```

	grade_id integer	count bigint
1	1	623
2	2	1437
3	3	2344
4	4	3288
5	5	3464



NASHVILLE SOFTWARE SCHOOL

Aggregating Data with GROUP BY



The **GROUP BY** keyword will subdivide the table based on the specified columns. You can then perform aggregations on the subgroups:

```
SELECT grade_id, COUNT(id)
FROM author
GROUP BY grade_id
ORDER BY count DESC;
```

	grade_id integer	count bigint
1	5	3464
2	4	3288
3	3	2344
4	2	1437
5	1	623

By default, ORDER BY sorts in ascending order, but we can change it with DESC.

NASHVILLE SOFTWARE SCHOOL

Joining Data

Often data will be spread across **multiple tables**. In order to perform an analysis you may need to **combine the data from two or more tables**.

Table A		
id	col_1	col_2
1	23-B	12
2	435	45
3	AB145	23
4	BB	56
5	435	123

Table B		
id	lookup_id	col_a
a1	1	a
a2	2	a
b1	3	b
b2	4	b
a3	5	a

NASHVILLE SOFTWARE SCHOOL

Joining Data

Often data will be spread across **multiple tables**. In order to perform an analysis you may need to **combine the data from two or more tables**.

Table A		
id	col_1	col_2
1	23-B	12
2	435	45
3	AB145	23
4	BB	56
5	435	123

Table B		
id	lookup_id	col_a
a1	1	a
a2	2	a
b1	3	b
b2	4	b
a3	5	a

NASHVILLE SOFTWARE SCHOOL

Joining Data

Often data will be spread across **multiple tables**. In order to perform an analysis you may need to **combine the data from two or more tables**.

Table A		
id	col_1	col_2
1	23-B	12
2	435	45
3	AB145	23
4	BB	56
5	435	123

Table B		
id	lookup_id	col_a
a1	1	a
a2	2	a
b1	3	b
b2	4	b
a3	5	a

NASHVILLE SOFTWARE SCHOOL

Joining Data

Often data will be spread across **multiple tables**. In order to perform an analysis you may need to **combine the data from two or more tables**.

Table A		
id	col_1	col_2
1	23-B	12
2	435	45
3	AB145	23
4	BB	56
5	435	123

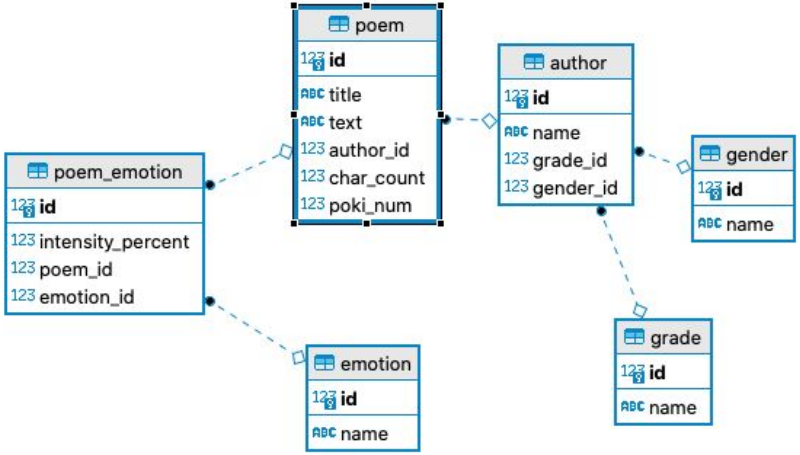
Table B		
id	lookup_id	col_a
a1	1	a
a2	2	a
b1	3	b
b2	4	b
a3	5	a

merge_table			
id	col_1	col_2	col_a
1	23-B	12	a
2	435	45	a
3	AB145	23	b
4	BB	56	b
5	435	123	a

NASHVILLE SOFTWARE SCHOOL

Joining Data - example

Goal: Retrieve the title of each poem along with the author's name.

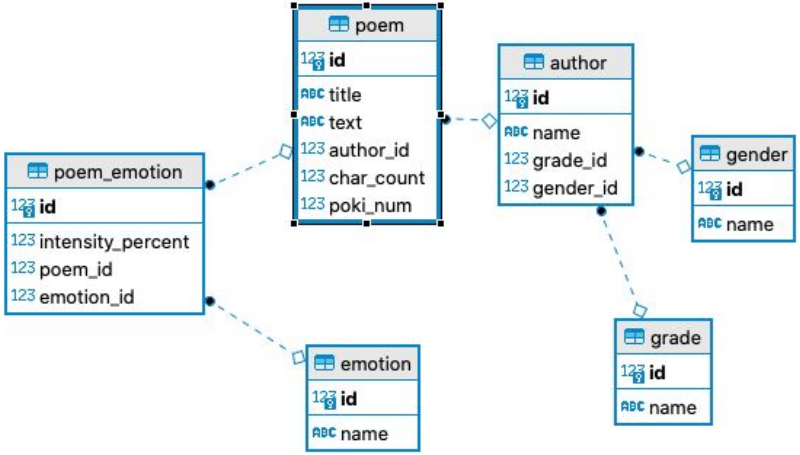


Joining Data - example

Goal: Retrieve the title of each poem along with the author's name.

Process:

a. decide what tables you need
b. decide how to connect your tables
c. decide which columns to select

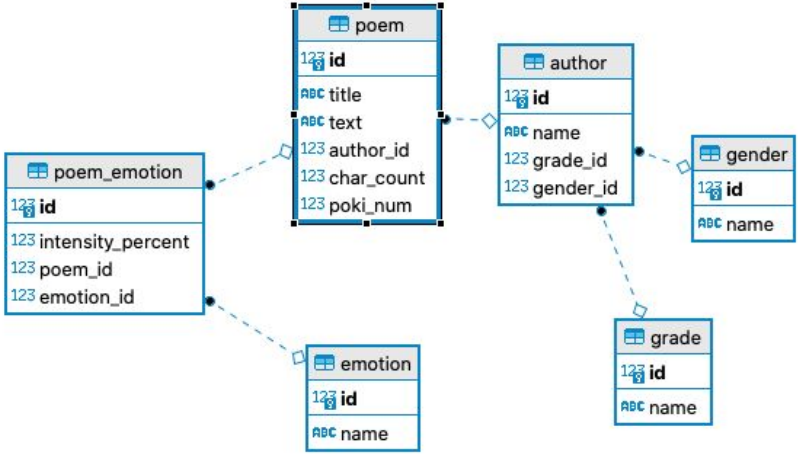


Joining Data - example

Goal: Retrieve the title of each poem along with the author's name.

Process:

a. decide what tables you need
b. decide how to connect your tables
c. decide which columns to select



SELECT
FROM
JOIN
ON

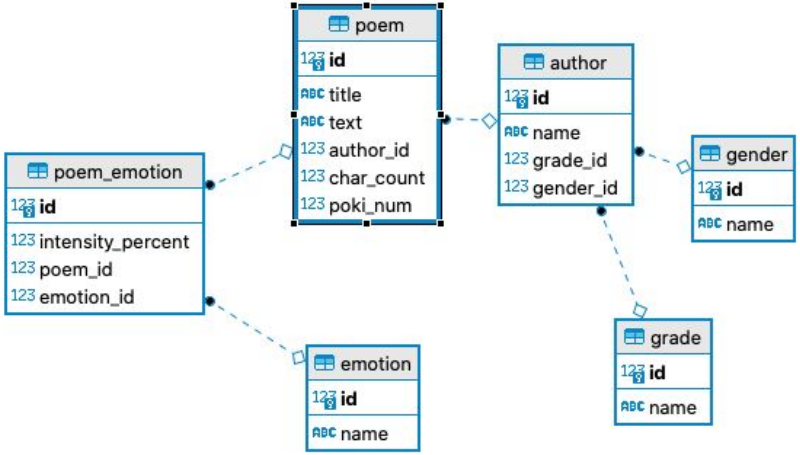
NASHVILLE SOFTWARE SCHOOL

Joining Data - example

Goal: Retrieve the title of each poem along with the author's name.

Process:

a. **decide what tables you need**
b. decide how to connect your tables
c. decide which columns to select



SELECT
FROM
JOIN
ON

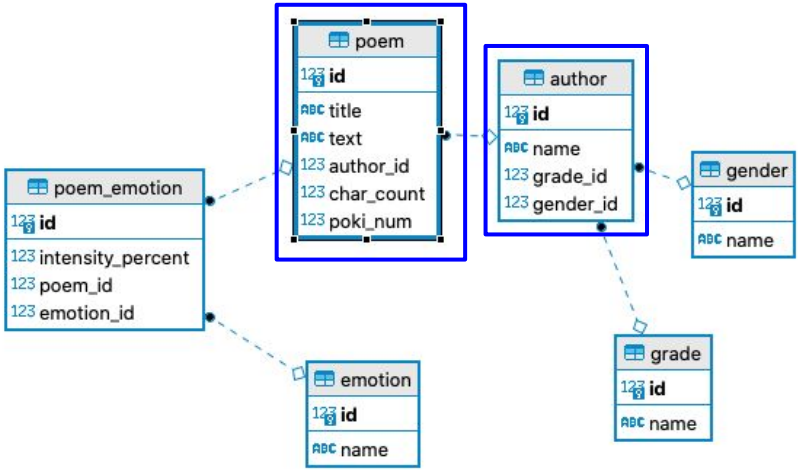
NASHVILLE SOFTWARE SCHOOL

Joining Data - example

Goal: Retrieve the title of each poem along with the author's name.

Process:

a. **decide what tables you need**
b. decide how to connect your tables
c. decide which columns to select



SELECT
FROM
JOIN
ON

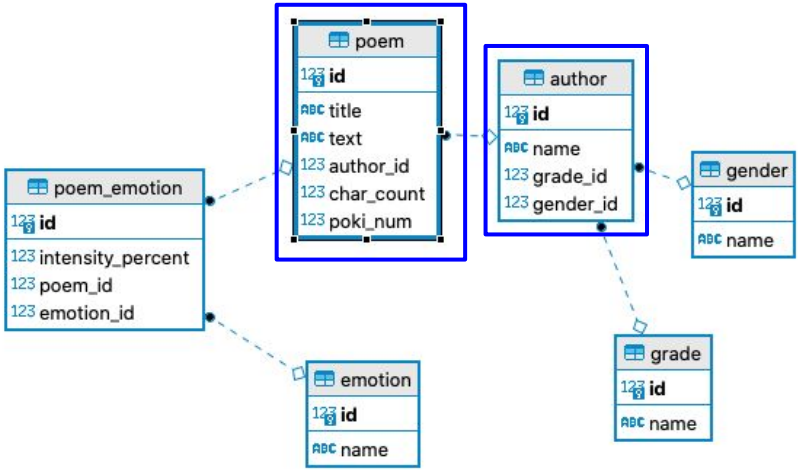
NASHVILLE SOFTWARE SCHOOL

Joining Data - example

Goal: Retrieve the title of each poem along with the author's name.

Process:

a. **decide what tables you need**
b. decide how to connect your tables
c. decide which columns to select



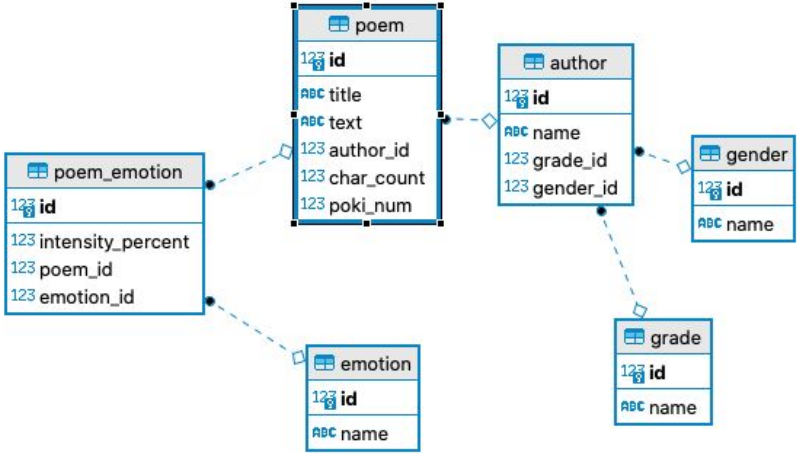

```
SELECT
FROM poem
JOIN author
ON
```

Joining Data - example

Goal: Retrieve the title of each poem along with the author's name.

Process:

a. decide what tables you need
b. decide how to connect your tables
c. decide which columns to select



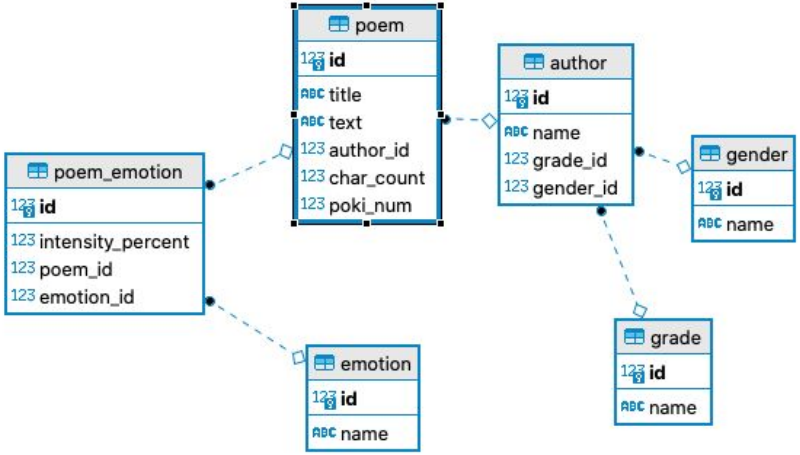

```
SELECT
FROM poem
JOIN author
ON
```

Joining Data - example

Goal: Retrieve the title of each poem along with the author's name.

Process:

a. decide what tables you need
b. decide how to connect your tables
c. decide which columns to select




```
SELECT
FROM poem
JOIN author
ON poem.author_id = author.id;
```

Specify columns using *table.column*

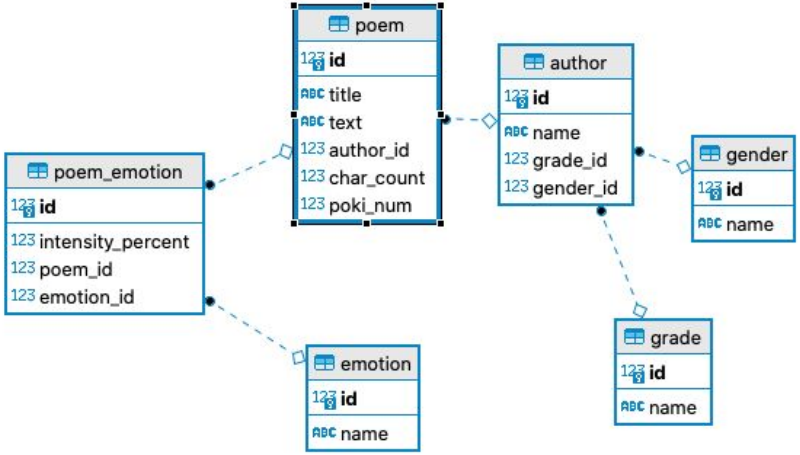
NASHVILLE SOFTWARE SCHOOL

Joining Data - example

Goal: Retrieve the title of each poem along with the author's name.

Process:

a. decide what tables you need
b. decide how to connect your tables
c. decide which columns to select




```
SELECT
FROM poem
JOIN author
ON poem.author_id = author.id;
```

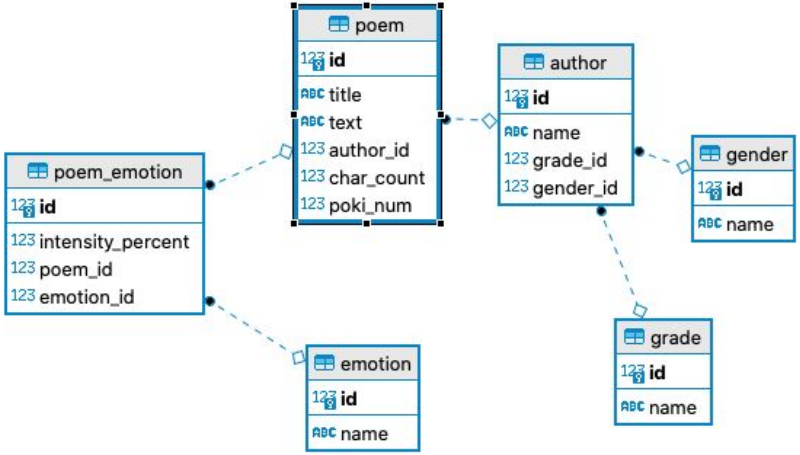
NASHVILLE SOFTWARE SCHOOL

Joining Data - example

Goal: Retrieve the title of each poem along with the author's name.

Process:

a. decide what tables you need
b. decide how to connect your tables
c. decide which columns to select




```
SELECT poem.title, author.name
FROM poem
JOIN author
ON poem.author_id = author.id;
```

Specify columns using *table.column*

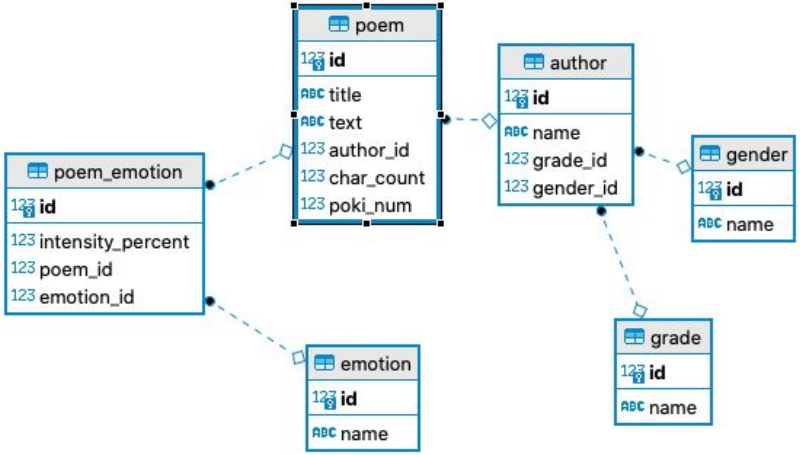
NASHVILLE SOFTWARE SCHOOL

Joining Data - example

Goal: Retrieve the title of each poem along with the author's name.

Process:

a. decide what tables you need
b. decide how to connect your tables
c. decide which columns to select




```
SELECT poem.title, author.name
FROM poem
JOIN author
ON poem.author_id = author.id;
```

	title character varying	name character varying
1	Computer	a
2	Angel	aab
3	Nature Nature and Nature	aadhya
4	Jack	aaliyah
5	When I awoke one morning	aanna
6	My Blue Berries and My Cherries	aarathi

NASHVILLE SOFTWARE SCHOOL

Additional Practice

W3Schools SQL Tutorial - https://www.w3schools.com/sql/default.asp

Khan Academy - https://www.khanacademy.org/computing/computer-programming/sql

DataCamp - https://learn.datacamp.com/skill-tracks/sql-fundamentals

NASHVILLE SOFTWARE SCHOOL