MAKE SCHOOL

Data Modeling & Relationships

BEW 1.2

MAKE SCHOOL

Welcome!

Warm-Up

MAKE SCHOOL

In a group of 3, answer:

- What is one thing that brought you joy over spring break?
- What is one thing you're excited to learn about this term?

MAKE SCHOOL

About This Course

Course Syllabus

MAKE SCHOOL

Go over the Course Syllabus and answer any questions.

Reminder that in order to pass this course, you must:

- Earn at least a score of **2 (Basic)** for **all but one homework assignment**.
- Pass the **final project** by earning a rubric score of at least **2 (Basic)**.
- Pass the final assessment by earning a score of at least 70%, **OR** earn at least 70% on all 3 quizzes.
- Abide by the Attendance Policy.

Agenda

MAKE SCHOOL

- What is Data Modeling?
- One-to-Many Relationships
- **Activity**: Worksheet Part 1
- Many-to-Many Relationships
- **Activity**: Worksheet Part 2
- **BREAK**
- Entity Relationship Diagrams
- Primary Keys & Foreign Keys
- **Activity**: Create an ERD
- Wrap-Up

Learning Outcomes

MAKE SCHOOL

By the end of today, you should be able to...

1. **Identify** the relevant entities for a given website, as well as the relationships between them.
2. **Create** an Entity Relationship Diagram for a given scenario.
3. **Identify** the Primary & Foreign Keys for a given entity.

What is Data Modeling?

What is Data Modeling?

MAKE SCHOOL

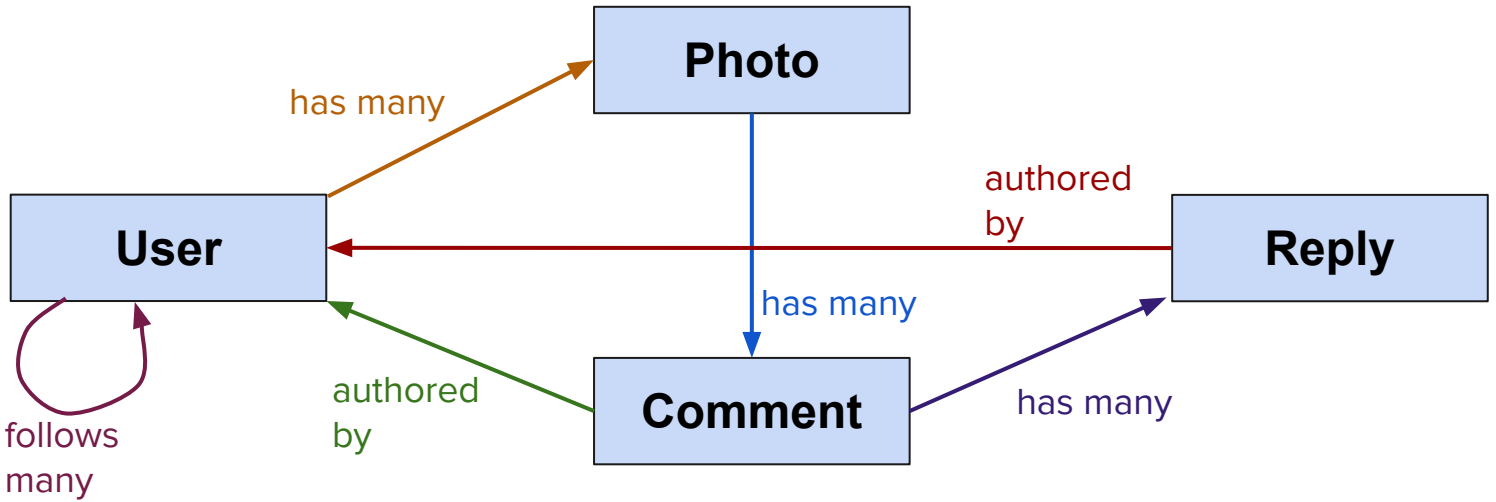
Let's say you were commissioned to build a web site.

On this website, **Users** can upload **Photos**, leave **Comments** on photos, leave **Replies** to comments, and **Follow** other users.

Where would you start building this? (What would the first steps be?)

What is Data Modeling?

I would start by figuring out the **data model**. That is, what **entities** do we need for the site, and how are they **related** to each other?



What is an Entity?

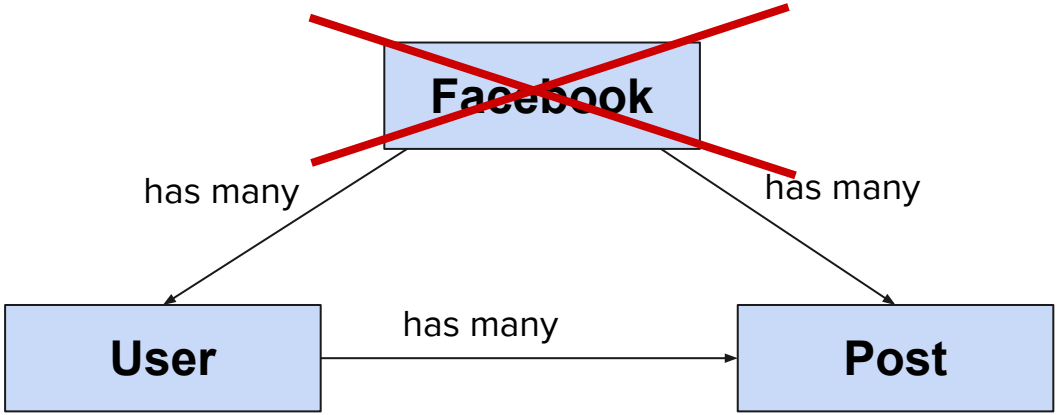
An **entity** (also called a **resource**) is a category of objects that have fields in common. You can think of it like a class.

BlogPost (class)	
title	`string`
content	`string`
author_id	`integer`

BlogPost (instance)	
title	'Top 5 Programming Languages'
content	'In my opinion, the top 5 programming languages include…'
author_id	5

What is NOT an entity?

A common misconception is to draw a diagram like this:



Why is this diagram incorrect?

Answer: "Facebook" is not an entity, because there is only one of it!

What is Data Modeling?

MAKE SCHOOL

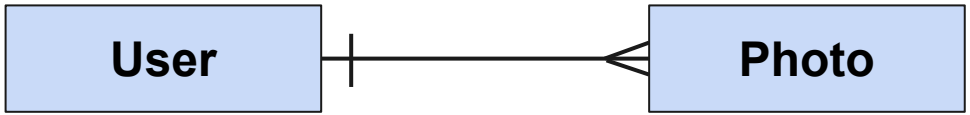
Data modeling is the process of determining what these entities and relationships are.

It is a necessary part of storing our data in a database, and is one of the first steps to creating a web application.

One-to-Many Relationships

One-to-Many

The most common type of relationship in a database is **one-to-many**.

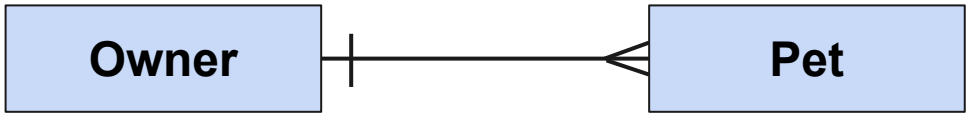


"One User has many Photos"

In diagrams, we use a cross and a **crow's foot** to symbolize the relationship.

One-to-Many

Is this a one-to-many relationship?



*"One Owner **can have** many Pets"*

A misconception is to say, "Well, an owner can have a pet, and a pet can have an owner! So it must be one-to-one!"

It is one-to-many if it is **possible** for an Owner to have many Pets, regardless of if they actually do.

Activity - Worksheet (10 minutes)

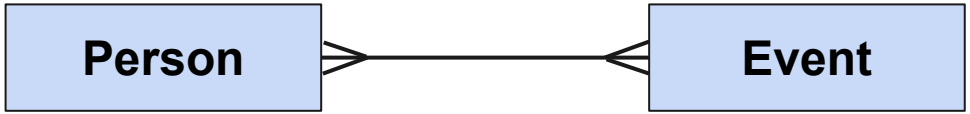
MAKE SCHOOL

With a partner, fill out the One-to-Many Relationships worksheet by drawing the appropriate relationships for each scenario.

Many-to-Many Relationships

Many-to-Many

Another type of relationship is **many-to-many**.

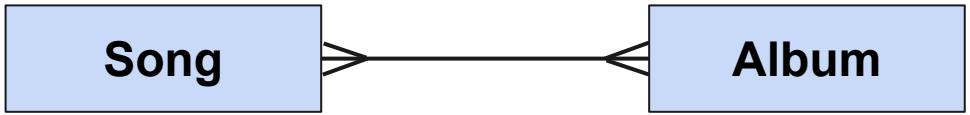


"One Person can attend many Events, and an Event can be attended by many People"

In diagrams, we use a **double crow's foot**  to symbolize the relationship.

Many-to-Many

Is this a many-to-many relationship?

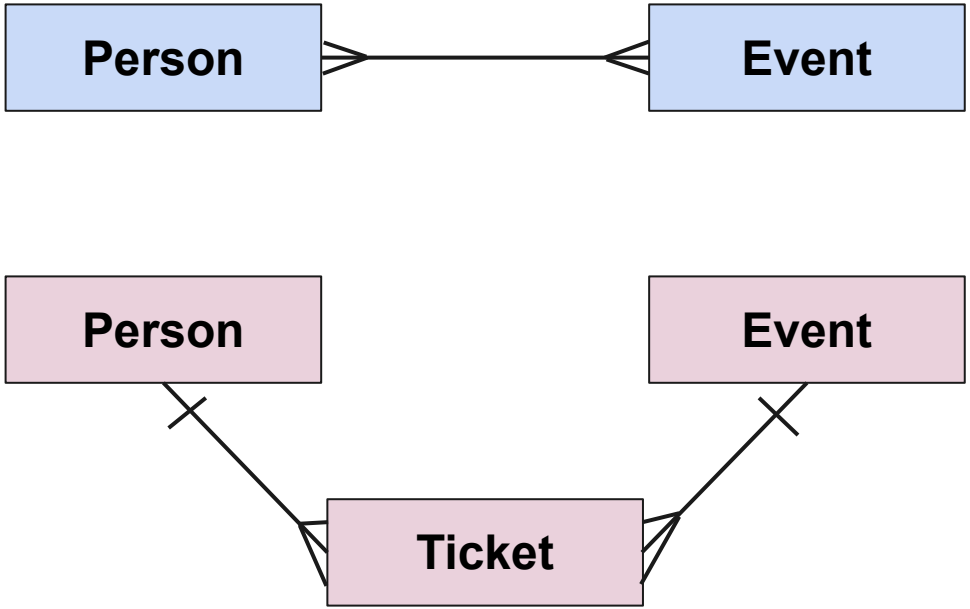


Answer: It depends!

You, as the developer, will need to decide whether it's okay for one song to be on multiple albums.

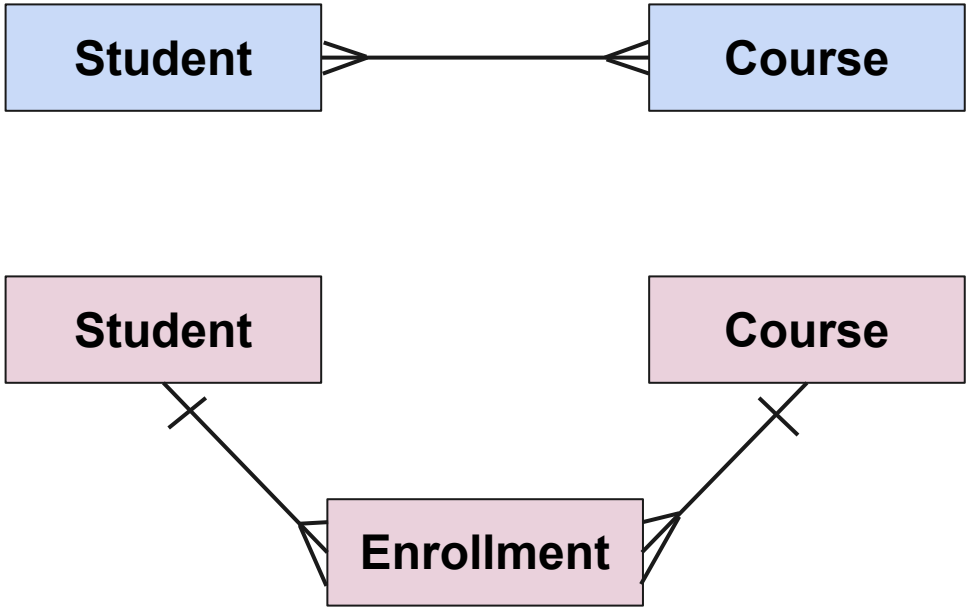
Many-to-Many

Every many-to-many relationship can also be written as **two** one-to-many relationships via a "bridge" table.



Many-to-Many

Let's try one more together. How would we re-write this relationship with a middle entity?



Many-to-Many

This means that if we have **3 students** each taking **3 classes**, we'd have **9 enrollments**.

Students
Sally
Quinn
Joe

Courses
CS
Writing
SPD

Enrollments	
Sally - CS	Joe - CS
Sally - Writing	Joe - Writing
Sally - SPD	Joe - SPD
Quinn - CS	
Quinn - Writing	
Quinn - SPD	

Activity - Worksheet

With a partner, fill out the Many-to-Many Relationships Worksheet to add a bridge table to each relationship.

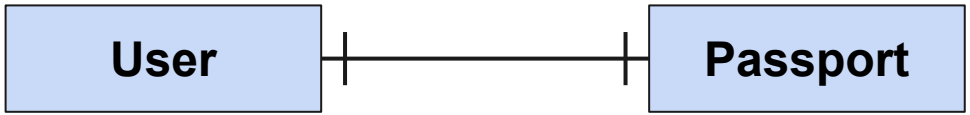
One-to-One Relationships

One-to-One

One-to-one relationships are relatively uncommon in database design, but we do need them sometimes.

Example: Marriage. **One** person can only have **one** spouse (at least in the US).

Another example: Passports. **One** user can only have **one** passport.



MAKE SCHOOL

Break - 10 min

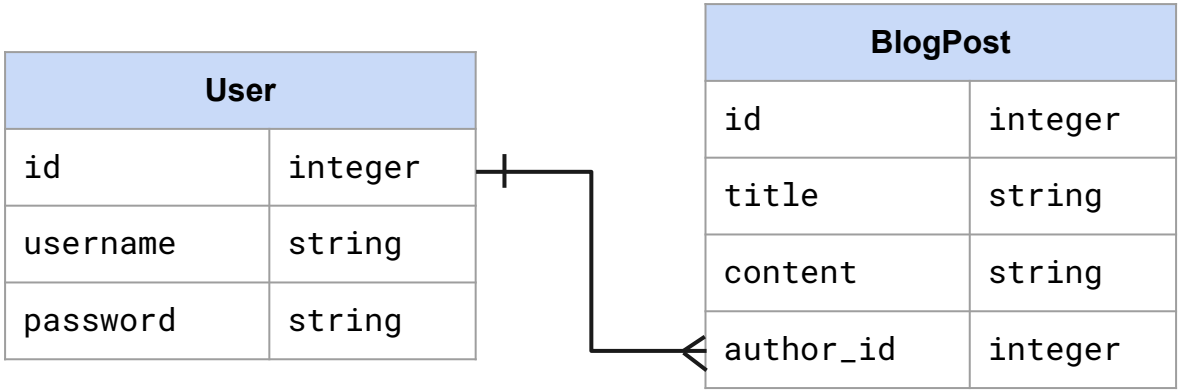
MAKE SCHOOL

Entity Relationship Diagrams

ERDs

An **entity relationship diagram**, or **ERD**, is a formal way of describing the relationships in a database.

It contains a table describing each entity, and lines between them to describe the relationship.



Lucidchart

MAKE SCHOOL

Lucidchart is online software that makes it easier to create professional-quality ERDs.

You can use your student email to get a free student Lucidchart account!

Watch as your instructor demonstrates how to create a diagram.

MAKE SCHOOL

Primary Keys

Primary Keys

A **primary key** is a field in an entity that can be used to uniquely identify each instance of the entity. Example: **username**.

A primary key must be:

- Unique (no two users can have the same username)
- Not null (every user must have a username)
- Never changing (a user cannot modify their username)

Primary Keys

Usually, an **id** field is used as a primary key.

Let's consider some more. Would each of these make a good primary key for a **User** entity?

- First and last name ❌ Not unique
- Phone number ❌ Could change
- Email address ❌ Could change
- Social security number ❌ Not everyone has one

Primary Keys

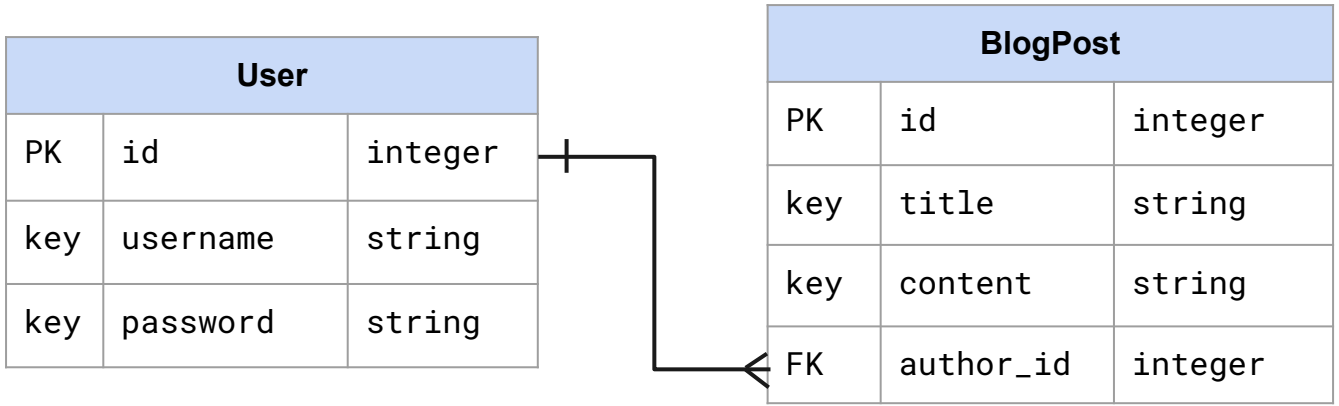
We can label a primary field with **PK** in the entity relationship diagram.

User		
PK	id	integer
Key	username	string
Key	password	string

Foreign Keys

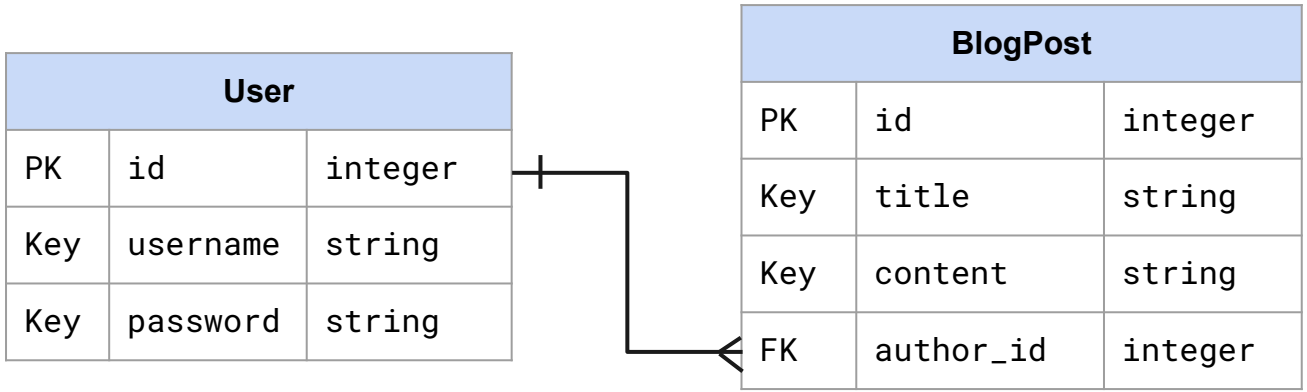
Foreign Keys

A **foreign key** is just a primary key that's in a foreign place - that is, stored within another object. For example, the **User**'s **id** field is a primary key, but when stored in a **BlogPost** as **author_id**, it is a foreign key.



Foreign Keys

We can label foreign keys as **FK** in the entity relationship diagram.



Activity - Create an ERD

With a partner, create a Lucidchart diagram for the following scenario:

On the website Pinterest, a User can create many Pins and many Boards. A Board can contain many Pins, and a Pin can belong to many Boards.

If you are having trouble getting started, watch this video on creating an ERD in Lucidchart.

MAKE SCHOOL

Wrap-Up

Wrap-Up

MAKE SCHOOL

- Homework: ERDs - on Gradescope