

# Week 2: Conceptual Data Modeling

## Quiz

Q1: A

Q2: D

Q3: A

Q4: A

Q5: B

## Relationships

- **One-to-many (1:m)** – a reference between entities where one entity may have many relationships with another entity (ex. Authors and articles have a one-to-many relationship because an author can write many articles, but an article can only be written by one author)
  - Can also be known as **many-to-one (m:1)**, just be careful how you define entity relationships with these two
- **One-to-one (1:1)** - a reference between entities where one entity may only have one relationship with another entity (ex. Couples in marriage generally have a one-to-one relationship, unless you believe in polygamy)
  - Denoted with a primary key that also serves as a foreign key
- **Many-to-many (m:n)** - a reference between entities where both entities may have many relationships with the other entity (ex. Articles and tags have a many-to-many relationship, because one article may contain several tags, and one tag can be associated with several articles)
  - Denoted using a **junction table**, which is a table whose rows correspond with relationships between the two many-to-many entities.

## Postgres

- **Creating a table** - We can create tables using an SQL command:

```
CREATE TABLE SomeTable (  
    attribute_name TYPE [PRIMARY KEY],  
    notha_attribute_name TYPE,  
    ...  
);
```

- **PRIMARY KEY** - denotes an attribute as a primary key for the table
- **Getting information from Postgres** - if you forget whether or not something is in Postgres, fear not.

```
postgres=> \d [SomeTable]
```

- This command will return a list of all the tables in the current working database.
- If a table is specified, it will return a list of all the attributes in the table, along with their types and modifiers.
- **Running .sql files from psql** - We can run SQL scripts in a file directly with psql:

```
postgres=> \i path/to/sql/file.sql
```

- **Copying .csv files into tables** - We can also copy rows directly from a .csv file to a table:

```
postgres=> \copy SomeTable from path/to/csv/file.csv  
(format csv, header true)
```

- `format` - This option tells Postgres the format of the file from which we'll be copying data. In the example, it is set to `csv` because we will be copying a .csv file.
- `header` - This option tells Postgres whether or not there is a header in the file. In the example, it is set to `true` because our .csv file has a **header**, which is the row in the file that marks the attributes of the table, usually the first row.

### **Lab 2 Information**

Assignment sheet: [http://www.cs.utexas.edu/~scohen/labs/lab2\\_assign.pdf](http://www.cs.utexas.edu/~scohen/labs/lab2_assign.pdf)

Grading rubric: [http://www.cs.utexas.edu/~scohen/labs/lab2\\_rubric.pdf](http://www.cs.utexas.edu/~scohen/labs/lab2_rubric.pdf)