

Homework 2: E/R and Relational Translation (100 points)

Due Date: Wednesday, Oct 21 (04:00 PM)

Submission

All HW assignments must be submitted online via the HW2 in **Gradescope**. See the table below for the HW 2 submission opportunities. Note that after 4 PM on Thursday the 22nd, no further HW 2 submissions will be accepted. (We will be releasing the solution at that time.) Please strive to get all your work in on time! If possible, try to save the one dropped assignment for the end of the term when you are most likely to want/need it.

Date / Time	Grade Implications
Wednesday, Oct 21 (04:00 PM)	Full credit will be available
Thursday, Oct 22 (04:00 PM)	10 points will be deducted

E-R based Relational Schema Design [100 pts]

You have successfully designed an E-R diagram for CheckedTweets. Now, to set up a real database, you must design a relational schema to represent the objects described in the E-R diagram that you created. **Specifically, you will need to design an appropriate collection of tables and create a SQL DDL statement for each table.** To make sure your design is a good one, you have given your E-R diagram to an external database consulting firm, DBInstructor, Inc., to have their experts check the correctness of your design. DBInstructor will provide your final E-R diagram soon. Since time is of the essence, though, you may want to start designing the required relational schema now based on your original E-R diagram. Your final design must be based on the E-R schema that DBInstructor provides, however. That schema will appear shortly after Thursday's 04:00 PM HW 1 "drop dead" deadline has passed. (You can also get MySQL installed and running in the meantime!)

As you work on your design, if you see opportunities to avoid creating excess relations that stem from relationships, do indeed avoid them so as to make the schema simpler. Clearly list all of your relations and their attributes (including their types), primary keys, foreign keys (including the referenced target relations), and *not null* constraints by writing a SQL DDL statement for each table. The resulting design should capture the information and constraints of the E-R diagram as faithfully as possible. For types, use one of the following types for each column:

Category	Type	Remark
----------	------	--------

NUMBER	INTEGER	A number type for integer values.
	DECIMAL(x,y)	A number type for real values where x is the maximum number of digits and y is the number of digits to the right of the decimal point.
STRING	CHAR(n)	A fixed-length string type where n is the column length. (Use only when strings are all of the same fixed length.)
	VARCHAR(n)	A variable-length string type where n specifies the <i>maximum</i> column length.
DATETIME	DATE	A type used for values with a date part but no time part. The format is '0000-00-00'.
	TIME	A type used for values with a time part. The format is '00:00:00'.
	DATETIME	A type used for values with both a date part and a time part. The format is '0000-00-00 00:00:00'.
ENUM	String	A string type with the value being chosen from a list of permitted values.
JSON	JSON	A JSON type used to store a JSON document as a value.
BOOLEAN	BOOL	A type for true or false values.

Uses the entity, relationship, and attribute names from the final E-R (DBInstructor-provided) diagram when naming your tables and columns (to make it crystal clear how your design corresponds to the E-R diagram). Note that some relationship names are RESERVED words (namely, **FROM** and **USING**). For the **FROM** and **USING** relationship names, use the table names **EvidenceFrom** and **VerifiedUsing**, respectively.

Again, all of the following information should be included in the DDL statements for each table. Be sure to:

- [50pts] List the tables, columns, and column types in your design.
- [20pts] For each of your tables, identify its primary key column(s).
- [30pts] For each of your tables, identify its foreign key column(s) (and indicate which other table each one references) and any *not null* constraints and/or other integrity constraints (such as *unique* constraints).

Notes:

- When an instance of a parent entity type in an IsA hierarchy is deleted, any/all associated child class information should also be removed.
- When a tweet is deleted, the tweets that have replied or quoted the deleted tweet should set the reply/quote id to NULL. As an example, if tweet id = 2 replies to tweet_id = 1 and tweet 1 is deleted, then tweet_id = 2 should have its reply id set to NULL.
- Phone type can be either home, office, or mobile.

Here is an example of some DDL statements for a few tables (which you can try running on your own MySQL instance if you want):

```
DROP DATABASE IF EXISTS cs122a_hw; -- let's start fresh! :-)
CREATE DATABASE cs122a_hw;
USE cs122a_hw;

CREATE TABLE Company (
    name          VARCHAR(40) NOT NULL,
    PRIMARY KEY (name)
);

CREATE TABLE Car (
    VIN           INTEGER NOT NULL,
    make          VARCHAR(40),
    year          DATE NOT NULL,
    type          ENUM('sedan', 'hatchback', 'convertible', 'suv',
        'pickup', 'van')
                NOT NULL,
    PRIMARY KEY (VIN),
    FOREIGN KEY (make) REFERENCES Company(name) ON DELETE CASCADE
);
```

The best way to make sure that your solution is sound is to actually try it out on a real system – so go ahead and install **MySQL** at this time on your favorite HW platform (presumably your laptop) and use it to verify that your PDF submission as a SQL file will actually run properly. (In other words, do **not** turn in your homework with DDL statements that you haven't actually tested!)

Be sure to use the HW#2 Google Docs template as the basis for your submission, as its use is **mandatory**. The deliverable should first list the SQL DDL statements for creating tables for entities, including any supporting tables for *entity-related* information. It should then list **(on a new page please)** the additional SQL DDL statements to create any additional tables for *relationships*. (Please organize your DDL statements in this fashion!) The current template has one page for entities and their supporting tables and one page for the additional relationships' DDLs. Feel free to append more pages at the end of each part for the PDF submission if you need more space.