Milestone 4 due Friday, 02/22.

Perform the following modeling tasks to improve the data quality of `dataset1`. Keep track of all SQL statements run for sections 1-3 in a file `transforms.sql`.

1. Create new modeled tables by applying the design principles learned in class:
   - split any raw tables that contain more than one entity into separate tables.
   - join any raw tables that store different attributes belonging to the same entity.
   - union any raw tables that store different records belonging to the same entity.
   - identify a primary key (PK) for each modeled table.
   - check for the presence of duplicate records.
   - remove any duplicate records found.

2. Identify relationships between the modeled table:
   - connect the tables in the diagram using the appropriate relationship type
   - checking for any referential integrity violations
   - remove any records that violate referential integrity

3. For each field in the modeled tables, choose a primitive data type that is most precise for its domain of values:
   - if the field is of type STRING and it stores INTEGER, NUMERIC, DATE or TIMESTAMP values, cast its type to the most fitting type.
   - if the field is of type INTEGER and it stores a DATE or TIMESTAMP value, cast its type to the most fitting type.
   - If the field is of type TIMESTAMP and the values it stores are of type DATE (i.e. the time component is not being used), cast its type to DATE.
   - Use BQ's **`CAST`** function to convert from one data type to another.
   - If the **`CAST`** function returned an error, make note of the field which could not be converted in a `TRANSFORMS.txt` file.

4. Create a file `ERD_v2.pdf` that denotes:
   - current state of your modeled tables (as opposed to future state)
   - field names, data types, and keys (PK, FK) for each entity.
   - relationships between entities.

5. Verify queries:
   - re-run join queries developed for Milestone 3.
   - fix any broken queries and update SQL files with code fixes.

Commit and push `transforms.sql`, `TRANSFORMS.txt`, and `ERD_v2.pdf` to your Github repo. Note that `TRANSFORMS.txt` is only required if you encountered errors during type casting.

CS 327E Milestone 4 Rubric
Due Date: 02/22/19

For `dataset1`, all tables should have an identified primary key. Values in the primary key should have no duplicates. String fields, if able to be casted to a more fitting type, should be. In addition, identify all entity types in your tables, split additional entity types into their own tables, join tables belonging to the same entity type, and union all tables that share the same fields. **-40** `transforms.sql` not found in repository **-20** no primary keys identified from ERD **-10** marked primary keys contain duplicates **-10** each string field containing only `INTEGER, NUMERIC, DATE, or TIMESTAMP` not cast, up to **-40** partial credit is awarded for explanations in `TRANSFORMS.txt` **-10** each non-merged entity type, table with multiple entity types, or un-unioned tables containing the same data (i.e tables representing the same data across different years).	40
For `dataset1`, all child tables should have an identified foreign key. **-30** no foreign keys identified on child tables in ERD **-20** relation is incorrect **-15** orphaned rows contained in child table	30
An ERD should be pushed that contains all detailed information for the fields in `dataset1`. Note that credit from other parts of the assignment may rely on this part. **-30** `./ERD_v2.pdf` not found in repository **-10** missing field types **-10** missing field names **-10** missing field keys **-5** incorrect keys marked	30
Fix all broken SQL statements from previous milestones 3-5. Make sure each statement runs properly. Save them into the same files, replacing the broken statements with their fixed counterparts. **-5** each erroneous SQL query, up to **-20**	

`submission.json` submitted into Canvas. Your project **will not** be graded without this submission. The file should have the following schema:
 	
`{`
`"commit-id": "your most recent commit ID from Github",`
`"project-id": "your project ID from GCP"`
`}`
 	
Example:
 	
`{`
	Required

<pre>"commit-id": "dab96492ac7d906368ac9c7a17cb0dbd670923d9",
"project-id": "some-project-id"
}</pre>	

Total Credit:	**100**