GROUP-11

Ulaş Yıldız - 24931

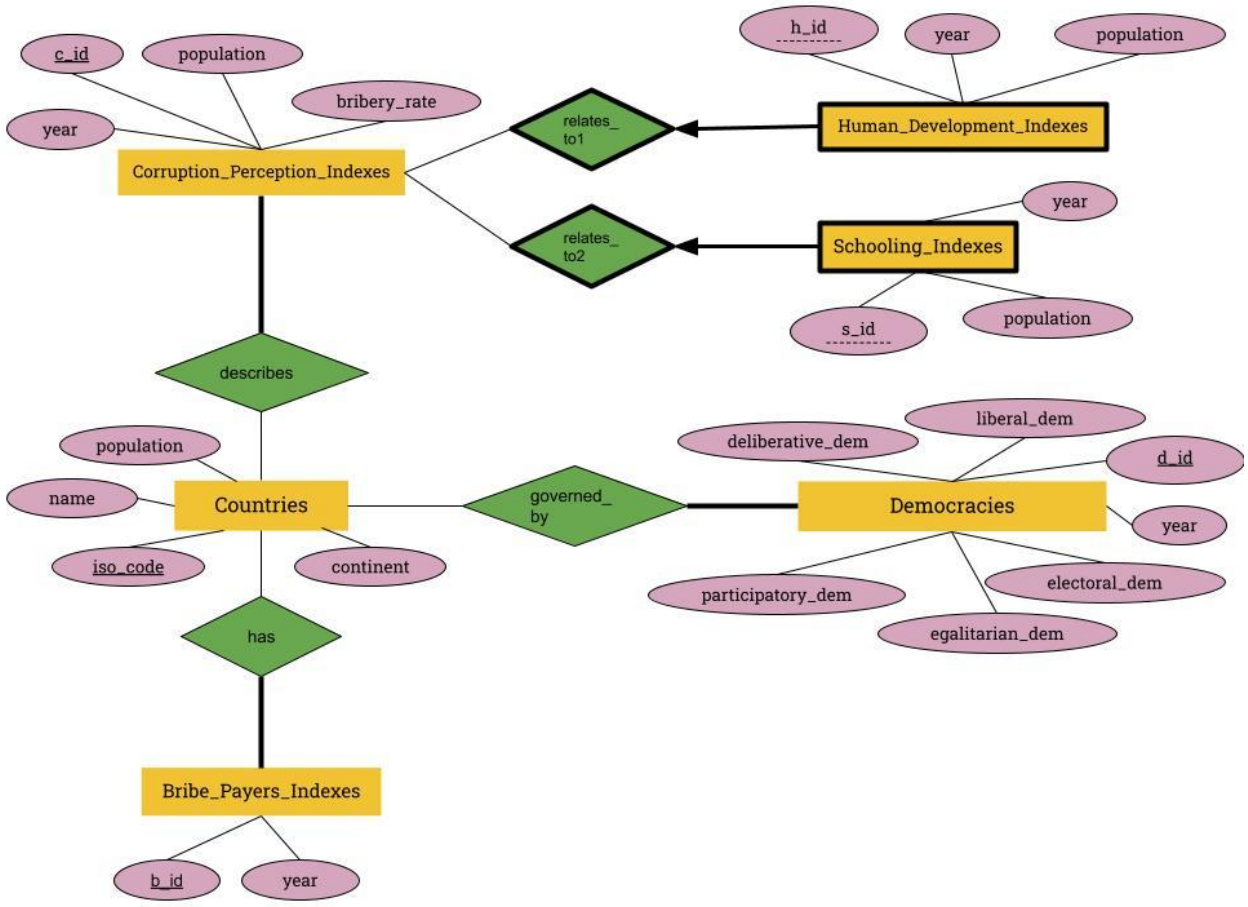
Selin Ceydeli - 28877

Berk Kayabaş - 29485

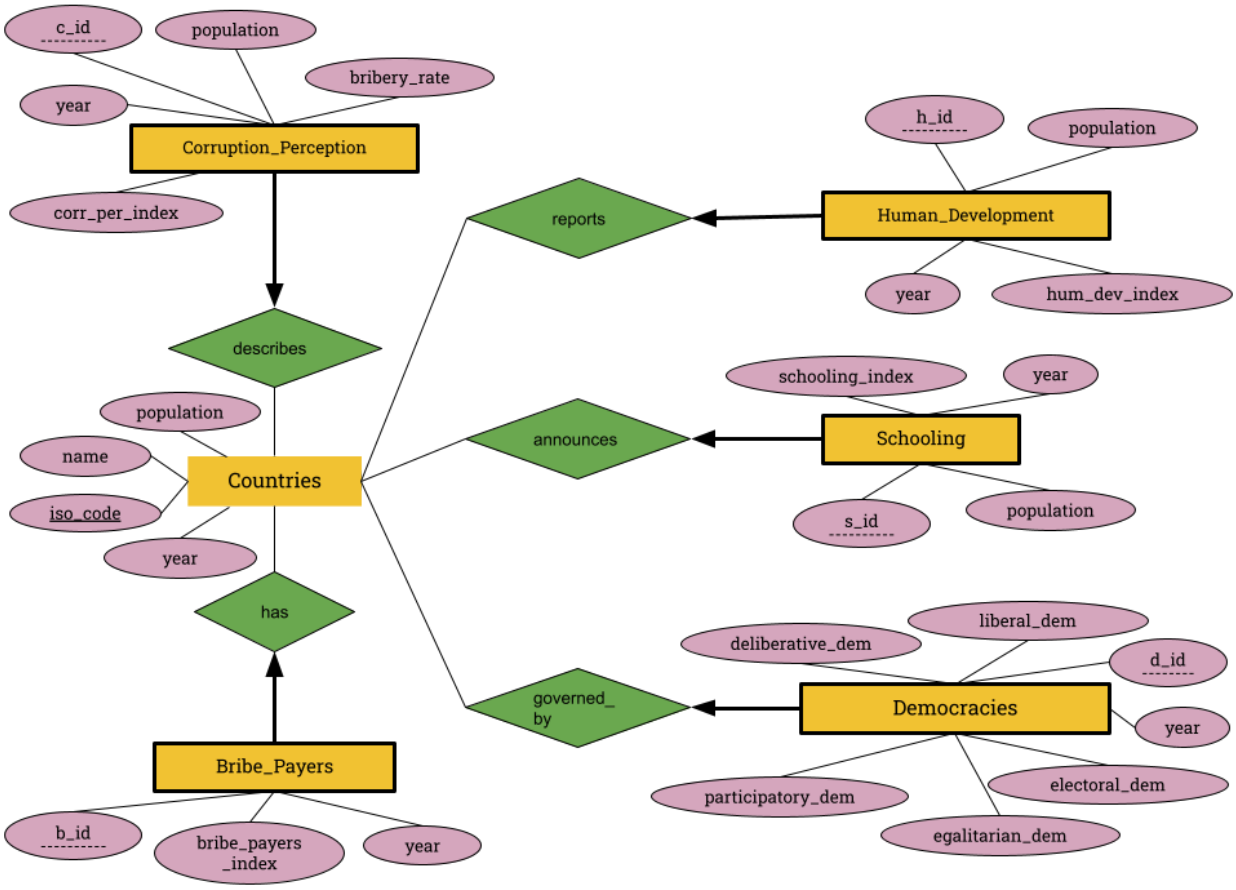
Mehmet Barış Tekdemir - 29068

Canberk Tahıl - 29515

Changes we made in our ER model:



(Old ER Model)



(New ER Model)

In our initial ER model, we represented the Human_Development and Schooling as weak entity sets owned by the Corruption_Perception_Indexes entity set. However, during the implementation of the ER model in MySQL, we realized that it would be more efficient to establish a relationship between these entities and an entity set with a unique and identifiable primary key.

Out of all our entity sets, the only one that satisfied this requirement was the Countries entity set, which has the iso_code attribute. Therefore, we represented the Human_Development and Schooling as weak entity sets owned by the Countries entity set in our new ER model.

After consulting with Elif Hoca during her office hour, we decided to represent the rest of the entity sets as weak entity sets owned by the Countries entity set as well. This decision was made because we wanted to include the iso_code attribute in the rest of the entity sets. As a result, in our new ER model, the owner entity set (Countries) and the weak entity sets participate in a one-to-many relationship set, with the weak entity sets having total participation in their identifying relationship sets.

Furthermore, to make the Bribe_Payers, Corruption_Perception, Human_Development, and Schooling_Index entity sets more representative of the information they hold, we included the index information as attributes.

Changes we made in our CSV files:

In the process of importing CSV files into tables, we encountered some errors related to foreign keys and duplicate primary keys. Upon further investigation, we realized that we needed a more comprehensive Countries dataset to correctly import the remaining entity sets, which were weak entities owned by the Countries entity set.

We found a new CSV dataset from Ourworldindata.org, named "population-past-future.csv", which we used as our new Countries dataset. We removed duplicates from this dataset and kept the latest information for each country corresponding to the year 2021, to avoid primary key conflicts due to different instances of the same country for different years.

Furthermore, we dropped a few rows from the Democracies.csv file that had meaningless country codes, as they did not match with the country codes in the Countries table. After dropping these rows, we were able to import the CSV file successfully into the table, without getting any foreign key errors.

While importing the CSV files, we did not encounter any further errors, which allowed us to proceed with the import process smoothly. As a result, we did not need to make any additional changes to the remaining datasets.

Our create table SQL statements:

```
CREATE TABLE Countries (
        iso_code CHAR(11),
        name VARCHAR(50),
        population INTEGER,
        PRIMARY KEY (iso_code)
);


CREATE TABLE Democracies_Governedby (
        iso_code CHAR(11) NOT NULL,
```

```
	d_id INTEGER AUTO_INCREMENT,
	year_ INTEGER,
	deliberative_dem DECIMAL,
	liberal_dem DECIMAL,
	electoral_dem DECIMAL,
	egalitarian_dem DECIMAL,
	participiary_dem DECIMAL,
	PRIMARY KEY(d_id, iso_code),
	FOREIGN KEY (iso_code) REFERENCES Countries (iso_code) ON DELETE
CASCADE
);


CREATE TABLE BribePayers_Has (
	iso_code CHAR(11) NOT NULL,
	b_id  INTEGER AUTO_INCREMENT,
	year_ INTEGER,
	bribe_payers_index REAL,
	PRIMARY KEY(b_id, iso_code),
	FOREIGN KEY (iso_code) REFERENCES Countries (iso_code) ON DELETE
CASCADE
);


CREATE TABLE Schooling_Announces (
	iso_code CHAR(11) NOT NULL,
	s_id INTEGER AUTO_INCREMENT,
	year_ INTEGER,
	schooling_index REAL,
	population INTEGER,
	PRIMARY KEY(s_id, iso_code),
	FOREIGN KEY (iso_code) REFERENCES Countries (iso_code) ON DELETE
CASCADE
);


CREATE TABLE HumDev_Reports (
	iso_code CHAR(11) NOT NULL,
	h_id INTEGER AUTO_INCREMENT,
	year_ INTEGER,
```

```
    hum_dev_index REAL,
    population INTEGER,
    PRIMARY KEY(h_id, iso_code),
    FOREIGN KEY (iso_code) REFERENCES Countries (iso_code) ON DELETE
CASCADE
);


CREATE TABLE CorrupPercep_Describes (
    iso_code CHAR(11) NOT NULL,
    c_id INTEGER AUTO_INCREMENT,
    year_ INTEGER,
    corr_per_index REAL,
    bribery_rate DECIMAL,
    population INTEGER,
    PRIMARY KEY(c_id, iso_code),
    FOREIGN KEY (iso_code) REFERENCES Countries (iso_code) ON DELETE
CASCADE
);
```

Our remarks about the Create Table SQL Statements:

In our final ER model, we made the decision to represent all relationships as one-to-many by defining all entity sets as weak entity sets that are owned by the Countries entity set.

To achieve this, we created a separate Countries table that serves as the owner entity set. Then, for the remaining weak entity sets, we combined the tables for the entity sets and their identifying relationships into a single table. This allowed us to avoid data redundancy and ensure data integrity by properly defining primary keys and foreign keys.

As a result, we have a total of 6 tables in our final ER model. The Countries table represents the Countries entity set. The other tables are combinations of the weak entity sets with their identifying relationships. Specifically, the Democracies_Governedby table represents the Democracies entity set and Governed_by relationship, the BribePayers_Has table represents the Bribe_Payers entity set and Has relationship, the Schooling_Announces table represents the Schooling entity set and Announces relationship, the HumDev_Reports table represents the Human_Development entity set and Reports relationship, and the CorrupPercep_Describes table represents the Corruption_Perception entity set and Describes relationship.

Our approach of combining the weak entity sets with their identifying relationships into a single table is an effective way to ensure data integrity while minimizing data redundancy.

In order to properly represent the relationships between entities in our MySQL database, we decided to assign numerical primary keys to all tables except for the Countries table. This was necessary because the datasets for these entity sets did not contain a unique and identifiable primary key like iso_code.

To achieve this, we decided to use the AUTO_INCREMENT attribute in our CREATE TABLE SQL statements. This allows MySQL to automatically generate a unique numerical value for each new record inserted into the table, ensuring that each entry has a unique primary key.

Files we have submitted to GitHub:

1. **updated_datasets.zip :** has our updated CSV files
2. **updated_ER-model :** is our updated ER model
3. **306_tables.sql :** has our create table SQL statements
4. **CS306_Project Step-2_Group-11.pdf** : is our PDF file
5. **sql_actions_local_Server.log :** is our log file that demonstrates the successful imports

Our GitHub link: https://github.com/berkkayabas/CS306-PROJECT.git