CS306 Project

Project Repository: [SleepyCoffy/CS-306-BasedData-Project: CS 306 Database project that will use data from www.ourworldindata.com and MySQL as a main tool. Project Group name: BasedData. (github.com)]

Group Name: **BasedData**

Group Members:

Name-Surname	Student ID	E-mail
Emir Aikan	28614	aikan@sabanciuniv.edu
Safa Abdullah Söğütlügil	29214	safasogutlugil@sabanciuniv.edu
Yunus Emre Şencan	29562	sencanyunus@sabanciuniv.edu
Yusuf Erkut Bayram	28976	yusuferkut@sabanciuniv.edu
Yusuf Kılıç	29431	yusufk@sabanciuniv.edu

Agriculture and Food Supply Database

Description of Step 2:

In Step 2, we have launched a database and have written SQL code to create relation tables. Then via Table Import wizard, we have imported entity csv files, which were prepared in the previous step, into newly created tables. During the step we have fixed errors that have occurred during csv import such as unwanted deletion of rows due to variable type conversion and missing inputs. Next parts of this report will thoroughly explain the details of Step 2.

SQL statements with constraints ():

Database creation code:

```
CREATE DATABASE CS306;
USE CS306;
SET GLOBAL general_log = 'ON';
```

cs306_countries.sql code:

```
create table countries(
    name varchar(50) Not Null,
    area_ha integer unsigned,
    primary key(name)
);
```

cs306_population.sql code:

```
create table population(
    country_name varchar(50) Not Null,
    year integer Not Null,
    size int unsigned,
    primary key(country_name, year),
    foreign key(country_name) references countries(name) on
delete cascade
 );
```

Cs306_animal_feed code:

```
create table animal_feed(
    country_name varchar(50) Not Null,
    year integer Not Null,
    weight_t int unsigned,
    primary key(country_name, year),
    foreign key(country_name) references countries(name) on
delete cascade
);
```

cs306_supply_chain_waste.sql

```
create table supply_chain_waste(
    country_name VARCHAR(50) NOT NULL,
    year INTEGER NOT NULL,
    weight_t INTEGER,
    PRIMARY KEY(country_name, year),
    FOREIGN KEY(country_name) REFERENCES countries(name) ON
DELETE CASCADE
);
```

cs306_food.sql code:

```
create table food(
    country_name varchar(50) Not Null,
    year integer Not Null,
    weight_t integer,
    protein_g_per_capita decimal,
    fat_g_per_capita decimal,
    primary key(country_name, year),
    foreign key(country_name) references countries(name) on
delete cascade
);
```

cs306_crops.sql code:

```
create table crops(
    country_name varchar(50) Not Null,
    year integer Not Null,
    production_t integer,
    export_t integer,
    import_t integer,
    primary key(country_name, year),
    foreign key(country_name) references countries(name) on
delete cascade
);
```

Entity sets and Relation sets conversion explanation:

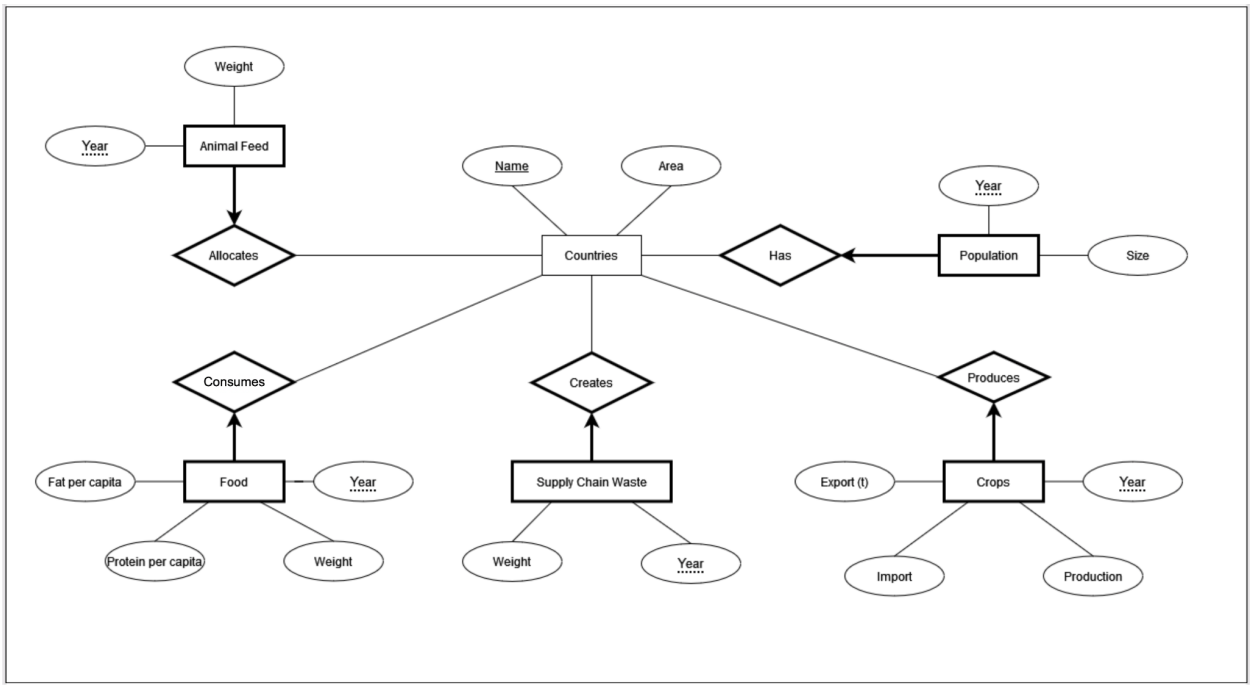

Figure-1.

Countries entity is the main entity in this database. It is a standalone entity and converted into a database table called **'countries'**.

Animal Feed entity is a weak entity with participation constraint and depends on Countries entity, it has many-to-one relation with Countries called 'Allocates'. Therefore, both Animal Feed entity and Relation 'Allocates' are converted into **'animal feed'** table.

Population entity is a weak entity with participation constraint and depends on Countries entity, it has many-to-one relationship with Countries called 'Has'. Thus, both Population entity and Relation 'Has' are converted into **'population'** table.

Food entity is a weak entity with participation constraint and depends on Countries entity, it has many-to-one relationship with Countries called 'Consumes'. Thus, both Food entity and Relation 'Consumes' are stored in **'food'** table.

Crops entity is a weak entity with participation constraint and depends on Countries entity, it has many-to-one relationship with Countries called ‘Produces’. Therefore, both Crops entity and Relation ‘Produces’ are converted into **‘crops’** table.

Supply Chain Waste entity is a weak entity with participation constraint and depends on Countries entity, it has many-to-one relationship with Countries called ‘Creates’. Thus, both Supply Chain Waste entity and Relation ‘Creates’ are converted into **‘supply_chain_waste’** table.

Error Fix and Repository file updates:

- We have reallocated previous Step 1 files into Step 1 folder in Github for integrity purposes.
- We have created a Step 2 folder in Github and uploaded necessary Step 2 files into this folder.
- During csv importing into tables some Rows with Empty Strings in INTEGER containing tables were disappearing due to standard behavior of MySQL, we have fixed it by updating csv files by changing the Empty Strings with NULL values.
- During csv importing of weak entities into tables some Rows were disappearing due to non-existing countries in the set, we have fixed it by adding the missing countries in Countries.csv file.
- All Log files were exported and uploaded into the Step 2 folder of the repository.

File names:

Country_Entity.csv: Includes the names and the maximum land use of the countries.

Food_Entity.csv: Amount of food and food supply in terms of protein and fat g per capita per day are for each country and year.

Population_Entity.csv: Population of each country throughout the years.

Crops_Entity.csv: Amount of food production and the countries import and export in tons per year for each country.

Animal_Feed.csv: Amount of allocated animal feed in tons per year for each country.

Supply_Chain_Waste.csv: Amount of food wasted in the supply chain in terms of tons per year for each country.

Cs306_supply_chain_waste.sql, **supply_chain_waste.log:** Code of the supply chain entity and its log no errors occurred during the process.

Cs306_population.sql, population.log: Code of the population entity and its log no errors occurred during the process.

Cs306_food.sql, food.log: Code of the food entity and its log no errors occurred during the process.

Cs306_crops.sql, crops.log: Code of the crops entity and its log no errors occurred during the process.

Cs306_countries.sql, countries.log: Code of the countries entity and its log no errors occurred during the process.

Cs306_animal_feed.sql, **animal_feed.log:**Code of the animal feed entity and its log no errors occurred during the process.

General.log: Shows the whole process from the beginning of creating tables to the end of importing the tables from csv files.