ETL Final Report

Background

Based on a historical dataset on the modern Olympic Games, we will find out which country won most medals through Extract, Transform, Load process. The dataset consists of 3 csv files: Summer, Winter, and Dictionary; It shows game information including year, city, sport and information of medal winners such as country, gender, name, medal color between 1896 and 2014. We wanted to use this dataset for ETL in order to answer the following questions:

I. Which country won most Gold/Silver/Bronze medal for summer olympic?
II. Which country won most Gold/Silver/Bronze medal for winter olympic?
III. Does GDP correlate with medals won?
IV. Does population correlate with the amount of medals a country has?
V. Who joined olympic both summer and winter for same year?

Extract:

This dataset shows the data from the olympics from 1896 to 2014. It separates it into two CSV files, summer and winter, as well as a dictionary CSV file. The files were organized into columns such as gender, name, medal, country, and the year of the game. This caused the dataset to contain multiple of the same name but different years and medals. We loaded the CSV files into jupyter notebook in order to prepare them for data cleaning and transforming.

Load CSV into DataFrame

In [ ]:
```
# Load Summer CSV into DataFrame
summer_file = "Resources/summer.csv"
summer_df = pd.read_csv(summer_file)
```

In [ ]:
```
# Load Winter CSV into DataFrame
winter_file = "Resources/winter.csv"
winter_df = pd.read_csv(winter_file)
```

In [ ]:
```
# Load Dictionary CSV into DataFrame
country_file = "Resources/dictionary.csv"
country_df = pd.read_csv(country_file)
```

Dataset Website: https://www.kaggle.com/the-guardian/olympic-games

Transform:

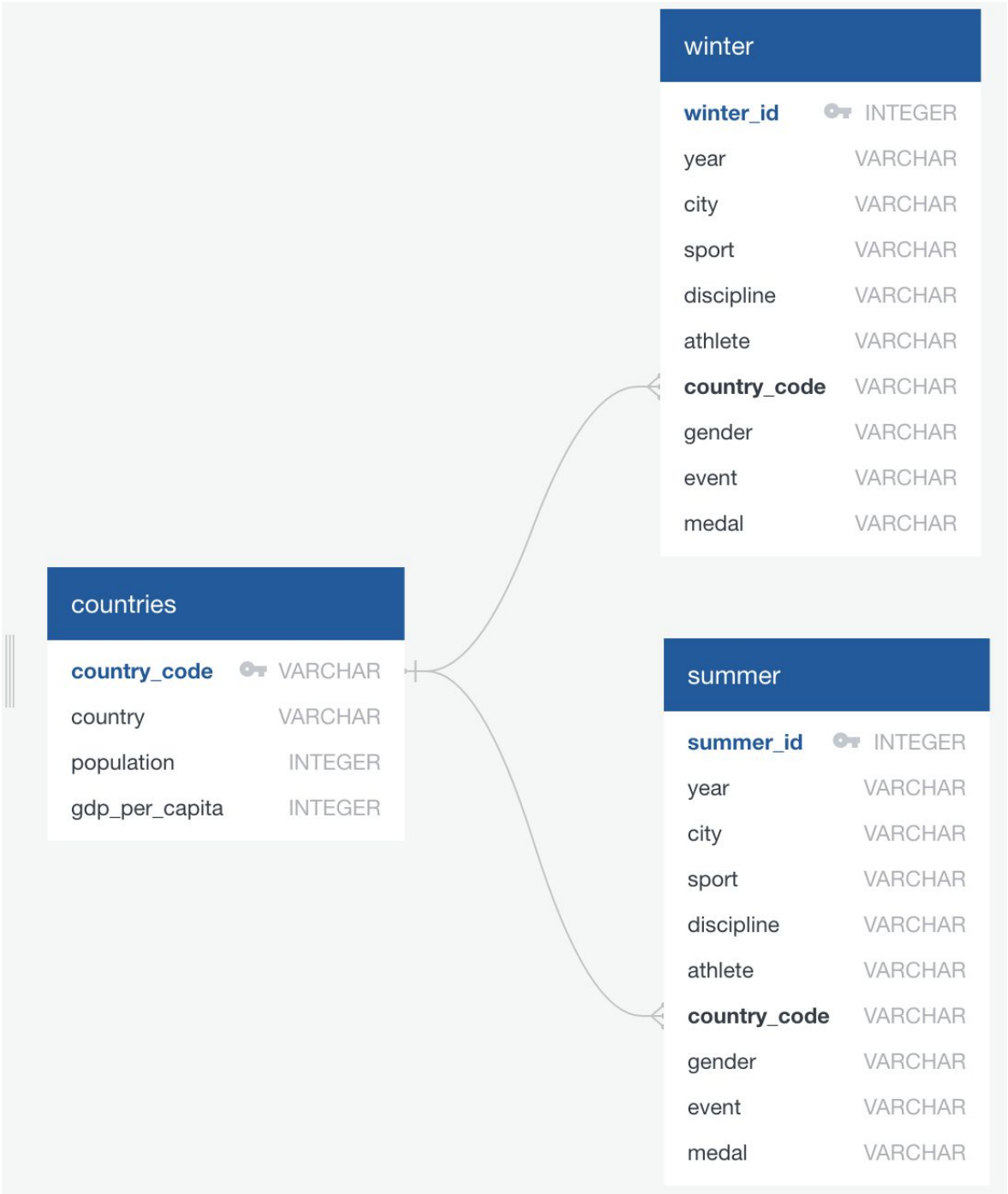
The dataset we had was really clean already. For our analysis and purposes, we did not have to drop any duplicates or N/A values. The duplicates were needed in order to answer some of our questions. The only transformation we needed to do was to rename the columns in each CSV file to make them easier to type as well as set an index name for each CSV to identify which dataset we were looking at. We then created an ERD to organize our data to see how we would join the data with primary keys and foreign keys in order to do the sql queries during the load phase.

Transform DataFrame

```
# Rename Columns
summer_df = summer_df.rename(columns={"Year": "year",
                                      "City": "city",
                                      "Sport": "sport",
                                      "Discipline": "discipline",
                                      "Athlete": "athlete",
                                      "Country": "country_code",
                                      "Gender": "gender",
                                      "Event": "event",
                                      "Medal": "medal"})

# Set index
summer_df.index.name = 'summer_id'

summer_df.head()
```

```
# Rename Columns
winter_df = winter_df.rename(columns={"Year": "year",
                                      "City": "city",
                                      "Sport": "sport",
                                      "Discipline": "discipline",
                                      "Athlete": "athlete",
                                      "Country": "country_code",
                                      "Gender": "gender",
                                      "Event": "event",
                                      "Medal": "medal"})

# Set index
winter_df.index.name = 'winter_id'

winter_df.head()
```

```
# Rename Columns
country_df = country_df.rename(columns={"Country": "country",
                                        "Code": "country_code",
                                        "Population": "population",
                                        "GDP per Capita": "gdp_per_capita"})

# Set index
country_df.set_index("country_code", inplace=True)

country_df.head()
```



Load:

In PGadmin4, we created a database and tables in it to begin the load process. The summer and winter tables had the same columns in order to be able to load the extracted data

from the csv files and answer our questions. The country table was to connect the summer and winter datasets together to make the join process easier.

olympics_db/postgres@PostgreSQL 12

Query Editor Query History

```sql
CREATE TABLE winter (
    winter_id INT PRIMARY KEY,
    year INT,
    city VARCHAR,
    sport VARCHAR,
    discipline VARCHAR,
    athlete VARCHAR,
    country VARCHAR,
    gender VARCHAR,
    event VARCHAR,
    medal VARCHAR
);

CREATE TABLE summer (
    summer_id INT PRIMARY KEY,
    year INT,
    city VARCHAR,
    sport VARCHAR,
    discipline VARCHAR,
    athlete VARCHAR,
    country VARCHAR,
    gender VARCHAR,
    event VARCHAR,
    medal VARCHAR
);

CREATE TABLE countries (
    country_code VARCHAR PRIMARY KEY,
    country VARCHAR,
    population INTEGER,
    gdp_per_capita INTEGER,
```

Data Output Explain Messages Notifications

We then used jupyter notebook and pandas to load our transformed CSV files into the database tables

Create Database Connection

```
In [ ]:  # Create Connection
         connection_string = "postgres:password:5432/olympics_db"
         engine = create_engine(f'postgresql://{connection_string}')
```

```
In [ ]:  # Confirm Tables
         engine.table_names()
```

Load DataFrame into Database

```
In [ ]:  # Load summer_df into database
         summer_df.to_sql(name='summer', con=engine, if_exists='append', index=True)
```

```
In [ ]:  # Load winter_df into database
         winter_df.to_sql(name='winter', con=engine, if_exists='append', index=True)
```

```
In [ ]:  # Load country_df into database
         country_df.to_sql(name='countries', con=engine, if_exists='append', index=True)
```

In order to confirm that the CSV files loaded correctly into the database on postgres sql, we queried the database and tables from jupyter notebook .

```
In [ ]:  # Query summer table
         pd.read_sql_query('select * from summer', con=engine).head()
```

```
In [ ]:  # Query winter table
         pd.read_sql_query('select * from winter', con=engine).head()
```

```
In [ ]:  # Query countries table
         pd.read_sql_query('select * from countries', con=engine).head()
```

We then used SQL queries to join and obtain the answers to our questions in the background section.

Query: 11 SQL query statements have been written to answer 5 questions in the Background section.

Query Editor Query History

```
--Count of Medal by countries order by GDP for summer olympic
SELECT countries.country, gdp_per_capita AS GDP, COUNT(medal) AS medal_quantity
FROM summer
JOIN countries ON summer.country_code = countries.country_code
GROUP BY country
ORDER BY GDP DESC;


--Count of Medal by countries order by GDP for winter olympic
SELECT countries.country, gdp_per_capita AS GDP, COUNT(medal) AS medal_quantity
FROM winter
JOIN countries ON inter.country_code = countries.country_code
GROUP BY country
ORDER BY GDP DESC;


--People who joined both summer and winter olympic in same year
SELECT summer.year, summer.athlete, summer.country, summer.gender, summer.city AS Summer_City, summer.sport AS summer_Sport, summer.
FROM summer
JOIN winter ON summer.athlete = winter.athlete AND summer.Year = winter.Year;
```

Data Output Explain Messages Notifications

	year integer	athlete character varying	country character varying	gender character varying	summer_city character varying	summer_sport character varying	summer_discipline character varying	summer_event character varying	summer_me character va
1	1932	BRUNET, Pierre	FRA	Men	Los Angeles	Rowing	Rowing	Pair-Oared Shell With C...	Bronze
2	1988	LUDING, Christa	GDR	Women	Seoul	Cycling	Cycling Track	Sprint	Silver
3	1988	LUDING, Christa	GDR	Women	Seoul	Cycling	Cycling Track	Sprint	Silver

After our queries, we obtained the following answers for our questions in the ETL process.

VI. Which country won most Gold/Silver/Bronze medal for summer olympic?
    A. Gold: USA
    B. Silver: USA
    C. Bronze: USA

VII. Which country won most Gold/Silver/Bronze medal for winter olympic?
    A. Gold: Canada
    B. Silver: USA
    C. Bronze: Finland

VIII. Does GDP correlate with medals won?
    A. No

IX. Does population correlate with the amount of medals a country has?
    A. No

X. Who joined olympic both summer and winter for same year?
    A. Pierre Brunet(1932) and Christa Luding(1988)