

Final Assignment User Guide

Name: Daniil Sergeev

Student ID: 21936856

Lab: Thursdays, 10am-12pm, Building 314, Superlab

General Software Information

Linux version and distribution: Ubuntu 22.04.4 LTS

MySQL version: 8.0.39

Virtual machine info: VirtualBox with 4 processors and 4096Mb Memory allocated

Explanation of Database Design

The first part of building this database, required for data analysis. The source of all the data came from the provided 2024 Paris Olympic Summer Games website. The design was done in a way similar to the way it was done in the mid-semester exam where the entities and attributes were first thought of. Afterward, the relationships between the entities were designed in a way that can lead to interesting and useful queries. Next, the Cardinality and participation constraints were put in place. After completing these parts, the ER diagram could be drawn using Chen's notation. Draw.io was the software used to draw the ER diagram. From the diagram, the Relational Schema can be derived using the steps listed in lecture 4.

Explanation of Implementation

After the design, I wrote a simple SQL script called CreateDB.sql from the Relational Schema design which creates an empty database called "Olympics_21936856" as well as all the tables in the database along with their primary and foreign keys, some of which are composite. Next, the CSV files were downloaded and filtered using Excel and later followed by simple Python scripts to get rid of characters that may cause issues when inserting data into the tables. After I designed a simple Python script that converts a CSV file into an SQL insert file, I also added a use database at the top to make sure the correct database and table are getting worked on. The Python script was then modified for every other CSV file to convert them too. Finally, a file was made to call all the other SQL insert scripts in the correct order to prevent issues with referential integrity.

Designing Queries and Further things

After the database was implemented, queries were made in a style similar to what the prac tests had. The Stored Procedures and views were created in the same style as the queries. When building the final part, the Python interface I had to take into consideration that the user may not have everything they need. The first part I built was a simple bash script that installs all the dependencies as well as a text file that has a command that will give the file execute permissions. After that file is run the user will have all the dependencies to implement the database. Next, Python requires a user to be able to connect to a database so I wrote another bash script like the last one but to create a new user which will be used by the MySQL Python connection. Finally, I created a python application that has a menu to perform CRUD (Create, Read, Update, Delete) operations on the database, I also copied the file with all the queries and broken them up into separate files with one query each, and the python app uses those prebuilt files for read operations.

Guide to Use the Database

Setting Up

To Start off unzip (if needed) and open the main directory of the project where you will see several other directories as well as an SQL file named CreateDB.sql. You will then need to open the terminal in this directory and open MySQL. If you don't have it, you can install it using the following commands in the terminal.

`Sudo apt-get update`

`Sudo apt install mysql-server`

Once MySQL is installed (if needed) you can open it. If done on a personal machine, you can do it by the following command.

`Sudo mysql`

If done on a lab computer, you can open MySQL using this command with the password myUserPassword.

`mysql -u me -p`

Once MySQL is open you can run the CreateDB file which will build the database and all its associated tables using this command.

`SOURCE CreateDB.sql;`

Generating Insert Files

Once the database has been created you can type `quit` into MySQL to leave it. This next part is not required as the Insert files have already been generated but if you are curious navigate into the CSV_Files directory and copy all the CSV files. Next, navigate back to the main directory of the project and go into the Convert_CSV_To_Insert directory, copy all the files into it where you can open a new terminal and run all the Python scripts in any order using `python3 filename.py`. Below is an example for one of the 7 files.

`python3 Insert_Medals.py`

Inserting Data

If you chose not to generate the insert files you can navigate into the Insert_SQL_Files directory, otherwise the generated files will work. You can then open MySQL in the way that was shown previously. There are 2 ways to insert data into the database a manual way and an automated way.

Automated method (Note script takes about 10 minutes to run due to the size of the dataset)

`SOURCE Insert_All.sql;`

Manual method: These files require to be inserted in a specific way to prevent issues with referential integrity, insert commands in this order and wait for them to finish before moving on to the next file.

```
USE Olympics_21936856;
```

```
SOURCE Insert_Athletes.sql;
```

```
SOURCE Insert_Coaches.sql;
```

```
SOURCE Insert_Teams.sql;
```

```
SOURCE Insert_Medals;
```

```
SOURCE Insert_Schedules.sql;
```

```
SOURCE Insert_Events.sql;
```

```
SOURCE Insert_Technical_Officials.sql;
```

Using Queries

Navigating to the Queries directory from the main project directory you will see a Queries.sql file which can be sourced like the previous files by opening MySQL in this directory, but a better alternative is to open the file and copy and paste individual queries into MySQL as there are explanation comments as well as giving you the opportunity to look at every individual query to see what it does.

Advanced Database Features

From the main directory, you can navigate to the Increasing_Functionality directory, and you will see several SQL files all of which can be sourced but are better copied and pasted into MySQL to see exactly what is going on. For the stored procedures copy the parts in block by block to see what they do paying special attention to copy everything within and including the DELIMITER fields. For views just copy individual blocks in and read the comments to see what each part does.

Python Database Implementation

Navigating to the Python_DB_Implementation from the main project directory you will see a further 3 directories. It's better to go through them to what is labelled (1, 2, 3) as it will make sure the user has everything needed to run the Python database application. Opening the first directory create a new terminal, open the permissions.txt file and paste the command into the terminal which will give the bash script permission to run.

```
sudo chmod 777 Install_Dependencies.sh
```

Next, run the Install_Dependencies script which will install everything needed using the following command to run it.

```
sudo ./Install_Dependencies.sh
```

Once the script finished running you need to create a user called dsuser (already created on lab computers), navigate to the second directory and open a new terminal there. Firstly you must give the bash script permission to run and then run it using these commands.

```
sudo chmod 777 Create_User.sh
```

```
sudo ./Create_User.sh
```

Once the dependencies have been installed and the user has been created you can finally run the Python database application. Navigate into the 3rd directory and run the Python script using the following command.

```
python3 PythonDB.py
```

Don't worry about the SQL files in this directory as they are used by the Python program. Once running the user is prompted to enter a number between 1 and 5 for different options. This application utilizes exception handling to prevent crashes due to invalid input.