Members:

Jiawei Liu	Jiapo Tai
Yuechen Luo	Hongru Du

- **Motivation for the Idea**

  In addition to the basic functions of an Olympic Games database, we would like to conduct some interesting queries such as "Is there a limitation on the strength/speed of mankind", "Is the total performance of a nation in Olympic Games dependent on its GDP per capita/population scale" We are curious about whether the sky is the limit, and we regard these questions as a way of interpreting the so-called Olympic Spirit.

- **Basic Features**

  Our web application can be divided into three kinds of queries, namely (1) queries based on sports events (2) queries based on national performance and (3) rankings.

  In section (1), we expect this database to be capable of answering questions like "what is the development tendency of Olympic records" or "how are the performance of silver winners compared to corresponding gold winners in history".

  In section (2), our DB should be able to do some statistics that are grouped by countries. In other words, it counts the total medals achieved by a nation and it can calculate the number of Olympic medals per capita.

  In section (3), this DB does some sorting based on either a sport event or a nation. i.e. It can return the ranking of medals achieved by different countries in a single Olympic game, or it can return the total historical ranking of medals achieved by different countries.

- **Possible Advanced Features**

  We will try to visualize the results of the queries and demonstrate them via figures, charts and cartoons. Simple statistical efforts such as fitting and regression might also be applied in order to indicate the trends of Olympic records.

- **Technology and Tools to Be Used**

  The back-end of this database will rely heavily on SQL and relational schemas while

Node.js will be adopted as the front-end platform of displaying the queries. MongoDB, as an additional, alternative solution, might be used as well if necessary.

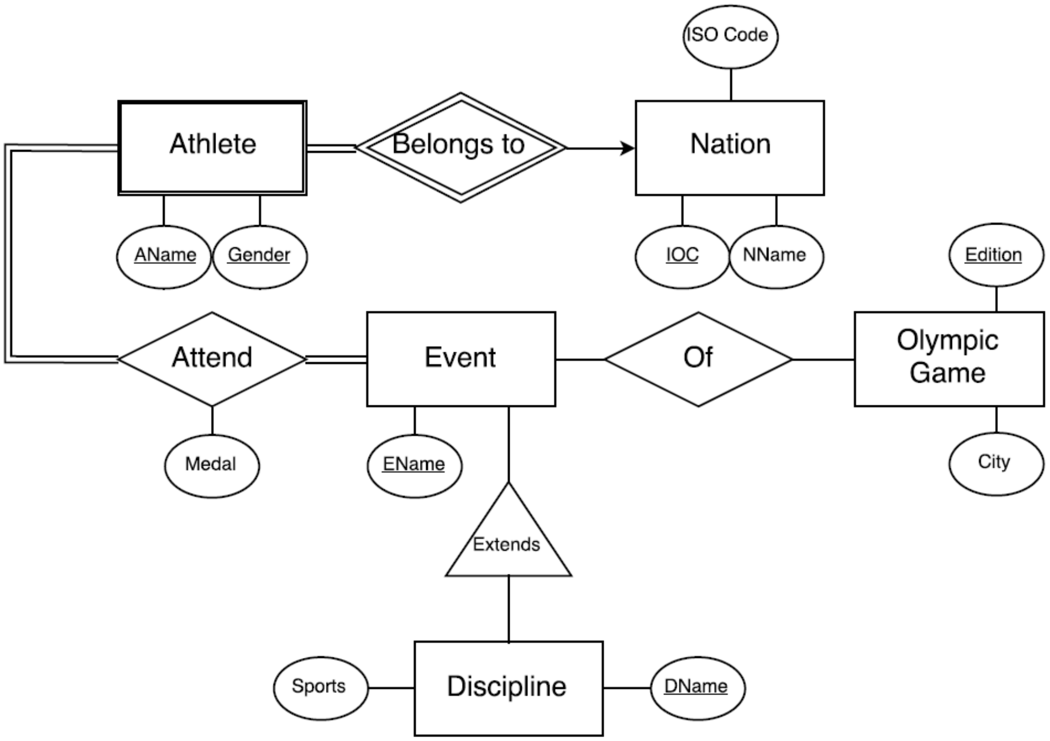
- **Complimentary Sources**

  In addition to the provided .csv file we will gather information from XML sources such as webpages. We will ingest the data via parsing the source code of the websites and injecting them to the relational database.

- **Member Responsibility**

  DDL/ER design: Hongru Du
  Importing data from CSV: Jiapo Tai
  XML data gathering: Yuechen Luo, Hongru Du
  Node.js: Jiapo Tai, Jiawei Liu, Hongru Du, Yuechen Luo
  Test/Demo: Jiawei Liu, Yuechen Luo
  Presentation: TBD

- **Schema Design**



- **SQL DDL**

```
CREATE TABLE Nation (
        IOC char(3),
        NName varchar(255),
        ISOCode char(2),
        PRIMARY KEY (IOC)
);

CREATE TABLE Athlete_Belong_To (
        IOC char(3),
        AName varchar(255).
        Gender varchar(5),
        PRIMARY KEY (IOC, AName, Gender),
        FOREIGN KEY (IOC) REFERENCES Nation(IOC),
        CHECK (Gender in ("Men","Women"))
);

CREATE TABLE Event (
        EName varchar(255),
        DName varchar(255),
        PRIMARY KEY (EName, DName)
);

CREATE TABLE Discilpine (
        DName varchar(255),
        Sports varchar(255),
        PRIMARY KEY (DName)
);

CREATE TABLE Attend (
        AName varchar(255),
        Gender varchar(5),
        EName varchar(255),
        DName varchar(255),
        Medal varchar(6),
        PRIMARY KEY (AName, Gender, EName, DName),
        FOREIGN KEY (AName, Gender)
            REFERENCES Athlete_Belong_To(AName, Gender),
        FOREIGN KEY (EName, DName) REFERENCES Event(EName, DName),
        CHECK (Gender in ("Men","Women")),
        CHECK (Medal in ("Gold","Silver","Bronze"))
);

CREATE TABLE Olympic_Game (
        Edition integer,
        City varchar(255),
        PRIMARY KEY Edition,
        CHECK (Edition >= 1908)
);

CREATE TABLE Event_Of (
        EName varchar(255),
        DName varchar(255),
        Edition integer,
        PRIMARY KEY (EName, DName, Edition),
        FOREIGN KEY (EName, DName) REFERENCES Event(EName, DName),
        FOREIGN KEY (Edition) REFERENCES Olympic_Game(Edition),
        CHECK (Edition >= 1908)
);
```