**University of Pisa**

Master’s degree in

Data Science & Business Informatics

Laboratory of Data Science



**Project Assignment - Part 1**

**Professors:**

Anna Monreale, Roberto Pellungrini

**Group29:**

Alfonso Ferraro Michele Andreucci

Student ID: 626006 Student ID: 628505

E-mail: [a.ferraro16@studenti.unipi.it](mailto:a.ferraro16@studenti.unipi.it) E-mail: m.andreucci4@studenti.unipi.it

**Introduction**

The datasets we’ll be working with are tennis.csv, male\_players.csv and female\_players.csv. The dataset tennis.csv is full of matches with much information about tournaments, winners, losers and technical features related to how the players performed and scored. On the other hand, male\_players.csv and female\_player.csv are datasets with only two features: names and surnames of the players.

With respect to the first part of this project, our goals are:

1. Create our own Database Schema from a reference figure given by the assignment using SQL Server Management Studio.
2. Write a python program that splits tennis.csv file into 4 chunks of different csv files with their proper names: “match”, “player”, “tournament”, “date” and retrieving the feature “sex” from male\_players.csv and female\_players.csv. These two tasks had to be accomplished without using pandas library.
3. Write a Python program that populates our database with the various tables from the .csv files, establishing schema relations as necessary.

**Assignment 0:**

Below is shown the database schema: Immagine che contiene testo, interni, screenshot

Descrizione generata automaticamente

We also established relationships among the entities:

* From *Date* we have *date\_id* as primary key that relates to *date\_id* as foreign keyin *Tournament*
* From *Tournament* we have *tourney\_id* as primary key that relates to *tourney\_id* as foreign key in *Match*
* From *Geography* we have *country\_ioc* as primary key that relates to *country\_id* as foreign key in Player
* From *Player* we have *player\_id* that relates to both *winner\_id* and *loser\_id* as foreign keys in *Match*.

**Assignment 1:**

The split of tennis.csv into match.csv, player.csv, tournament.csv and date.csv was done with csv library. We freely decided which columns where the most suited to be transferred into the different csv files and choose to execute the data cleaning part once we splitted everything.

The first pre-processing step was to create - with reference to the database schema - all the needed columns in each csv file. Pandas library was used to do so, almost everything was accomplished through the *.append()* method and *DataFrame.join()* method with **how=’outer’**, which is a form union of calling frame’s index (or column if on is specified) with other’s index, and sort it, lexicographically.

Immagine che contiene testo

Descrizione generata automaticamente

**Figure 1: E.g., of player.csv's main columns creation**

Some columns such as *match\_id* and *date\_id* were created through a simple concatenation. The *language* and *continent* features of geography’s entity were obtained via the merging of countries.csv and languages.csv.

The sex feature was retrieved from male\_players.csv and female\_players.csv thanks to csv library, with the addition of a third sex value “Sconosciuto” for all the unmatching/unfound players between the csv files, which was very important for not losing any player that was not present inside the above-mentioned csv files.

Day, month, year and quarter features of date.csv were created with the pandas method *.to\_datetime*

Once all the needed columns were created, it was possible to deal with missing values and columns:

* All unneeded columns such as winner\_id, winner\_name, etc… were deleted
* All the duplicates were managed with the drop*\_.duplicates()* method that was used to drop all *player\_id*, *date\_id, tourney\_id, match\_id* , *country\_ioc*) duplicates since they are the primary keys of our entities.
* The missing values of hand attribute were filled with ‘U’ which was one of the three unique values of the features
* The missing values of ht attribute were filled with the mean of all ht’s values based on sex and approximated to the closest integer

Immagine che contiene testo

Descrizione generata automaticamente

* The missing values of continent and language features were filled with ‘Unknown’
* All the missing values in w\_ace, w\_svpt columns etc… were managed by deleting all those rows
* The missing values of *age* attribute - which was used to get *year\_of\_birthday* feature – were filled with the mean of all age’s values and rounded down, then the column was deleted.

With respect to the composition and order of columns of the database schema, all the attributes of all the csv files were reindexed properly.

**Assignment 2:**

This second part was about populating the database tables with the various csv files, obtained from the previous assignment.

Firstly, a script has been used on SQL server management studio to create the 5 empty tables with the respective columns, then we used these tables to create the assignment: 0’s diagram.

Once everything was settled a function was created and invoked to populate each table.

Everything was executed taking into account the foreign keys and primary keys constraints.