# Project Assignment

Cristiano Landi, Anna Monreale

## Introduction

Given the .csv file students need to perform the following operations:

- Build a datawarehouse and populate it
- Build an ETL process
- Build a Data Cube
- MDX query
- Create a Dashboard

## Assignments

The project is about tennis matches; the data are a simplified version of this dataset available on GitHub. This project aims to simulate a decision support system for a sports federation for tennis.

Attached to this document, you can find 3 distinct files:

1. **fact.csv**: contains the main body of the data.
2. **tourney.json**: contains the data about the tournaments.
3. **countries.xml**: contains additional geographic data.

### ***Assignment 1:*** *data understanding*

Without modifying the date, understand the data you are working with. How do the files relate to each other? Are there missing values? Can the missing values be recovered/filled **easily**? Can you integrate additional data (coordinates, hierarchical GeoHash/Uber H3/Google S2 encoding for spatial data, additional weather conditions, etc.) from external sources with a **reasonable effort**?

ONLY for this assignment, you can use any software/package you want!

## FROM NOW ON, YOU CAN NOT USE ANY PACKAGES IMPLEMENTING BULK DATA MANIPULATION (LIKE PANDAS)

### ***Assignment 2:*** *data cleaning*

Given the information collected in the previous assignment, address the problem related to the missing data (if any) and integrate the additional data (if any).

### ***Assignment 3:*** *DW Schema*

It's time to switch from operational databases to analysis-oriented data warehouses. To design the DW schema, since we're simulating a Tennis Federation, the fact table should capture details of tennis matches. You can use the data warehouse schema in Figure 1 as a reference. Next, create the data warehouse tables on the server[1]. You must use the database named *Group_ID_DB* (example: Group_01_DB) as specified in the credentials' email.
Please note that the DW schema in Figure 1 is just a suggestion; **you can modify it as you prefer**. You can use both Python and SQL Server Management Studio to create the DW.

### ***Assignment 4:*** *Data preparation*

Write a Python program that splits the data into different files, one for each table in the data warehouse you proposed in the previous step.

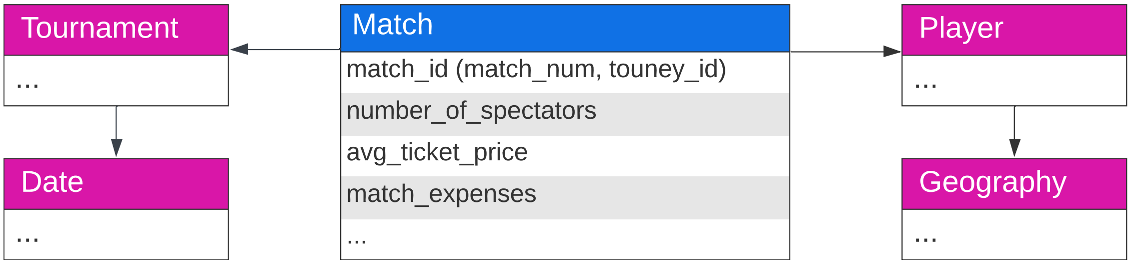

Figure 1: Datawarehouse schema of reference. Fact table in blue.

### ***Assignment 5***: *Data uploading demo*

Duplicate each table, renaming them as TABLENAME_SSIS. Then, create an SSIS project and a Python script that populates the corresponding set of tables with 20% of the data you prepared in Assignment 4. The integrity of the data should be preserved.
Please note that this operation could take a while, so design the code accordingly!

### ***Assignment 6***: *Data uploading*

Based on the performance of assignment 5, use the fastest method to upload all the data to the database.

At this point, you should have two fact tables and their corresponding (duplicated) dimension tables. From now on, perform all assignments using the fact table that contains all the data[2].

### ***Assignment 7***: *nemesis*

For any given player, his or her *nemesis* is the player against whom he or she lost most matches. For each year, list every player with the respective nemesis and the number of matches lost. Use SSIS.

### ***Assignment 8***: *age-outlier matches*

A match is labeled with *age-outlier* if the difference in years between the winner and the loser is more than $avg(agedifference) * 1.5$ w.r.t. all the matches in the same tournament. For each year, list the player that participated in the most age-outlier matches. Use SSIS.

[2] We suggest using the smaller fact table if youwant to verify the correctness of your results

### ***Assignment 9****: looser*

Show the player that lost the most matches for each continent. Use MDX.

### ***Assignment 10****: number of players*

For each tournament, show the number of players. Use MDX.

### ***Assignment 11****: profit*

For each quarter, calculate the percentage increase or decrease in profit for each tournament compared to the profit in the same tournament during the corresponding quarter of the previous year. Use MDX.

### ***Assignment 12****: Dashboard n.1*

Create a dashboard that shows the geographical distribution of winner rank points and loser rank points.

### ***Assignment 13****: Dashboard n.2*

Create a plot/dashboard that you deem interesting, including the data available in your cube, focusing on the financial informations.

All the files must be uploaded to Google Drive (using your university account). Then, you need to generate a sharing link and send it to **cristiano.landi@phd.unipi.it** and **anna.monreale@unipi.it**, along with a report of up to 10 pages in length.