Database Design Document:

Group 9 Team Members:
Jiangying Zhong
Jingxuan Li
Hongyi Luo
Liangze Ni
Junfei Ren

Database Purpose:
This database is designed for the fans who are crazy about European football players, video games, and the matches. They could use this database to check every match and which players participate in and their team and abilities. On top of that, they also could construct a predict model to predict the performance of each player and team. Such as which team has a higher win rate in which place and which players in which season or time have better performance than others.

Business Problem Addressed:
Analyze the impact of physical fitness on football players.
Analyze the strengths and weaknesses of football players.
Analyze the future development and value of football players.
Analyze where different players fit on the field.
Analyze which country and club have better football players.
Analyze the dominant foot of a football player.
Analyze the relationship between worth and wages and liquidated damages.

Business Rules:
Each player has only has one ID and name.
Each player only belongs to one nationality.
Each player has several positions, but only one best position.
Each country has only one country ID and name.
Each contract has only one contract ID.
Each contract contains only one team ID and one Player ID, which means each player only works for one team.
Each team has only one team ID.

Each team can only join one league.

Each match has only one match ID.

Each match contains only one homecountryID and one awaycountryID, which are both referenced from Country(CountryID).

Each position has only one position ID and name.

One player can play multiple positions, and multiple players can play each position.

One club can only participate in one league,but multiple clubs can participate in a league.

Players whose best position is not GK cannot be involved in Table GK.

Design Decision:

Entity Name	Why Entity Included	How Entity is Related to Other Entities
Player	It shows basic information about the Player,like ID, name and nationality. Also, in this entity, it will tell the player's wage and best position.	It is the core entity in this database. The primary key of this entity is player ID. It relates to the rest entities, such as Physical Fitness, Professional Skills, GK, Contract and Country. It connects to other entities with one-to-one relationships, one-to-many relationships and also many-to-many relationships.
Physic Fitness	In this entity, it shows the body index of players like height and weight. It also uses a kind of score attributes called PhysicalityTotal indicating the strength of players. The most important thing in this entity is that we calculate age by date of birth.	The physic fitness entity is directly related to the Player entity. One player has unique physical fitness. It is a one-to-one relationship.
Professional Skills	In this entity, it shows the ratings in one-hundred-point scale of basic skills, such as pace total, shooting total, passing total, dribbling total and defending total.	The professional skills entity is straightly related to the Player entity. One player has only one rating for each professional skill. It is a one-to-one relationship.
GK	GK is a really important aspect in evaluating goalkeepers. It includes the scores of positioning, diving, kicking, handling, reflexes and total rating.	The GK entity is related to the Player entity by PlayerID. One player who can play as a goalkeeper has unique ratings of GK attributes. It is a one-to-one relationship.
Contract	Contract is the entity that highly connects the player and club. It contains both TeamID and PlayerID. Also, it has contractID as the primary key and several other attributes like joined time, end time and release clause of different players.	Contract entity is related to the Player entity and Team entity using PlayerID and TeamID respectively. Each player can only work for one team, however, one team can absorb one or more players. Hence, the relationship between player and contract should be one-to-one, while the relationship between team and contract should be one-to-many.

Team	It shows the basic information about thier long name, short name, ID and other FK to connect with other dataset.	Team entity is related to the Team ability and Players. Each player can only work for one team and one team could have many players. Team could have many abilities and ability could attribute to many teams.Therefore, the relationship between Players and Team is one to many and the relationship between Team to Team Ability is many to many.
League	It shows the basic information about the league, including ID, name and the ID of teams that join the league.	League entity is directly related to the team entity by using teamID as the foreign key. One team can join one league at a time, while a league should contain various teams. So it is a one-to-many relationship.
Team_Ability	Team_Abality includes ID, PlayersID, Overall Ability, Name.	It relates to Team. Each Team could have multiple abilities and ability could be assigned to many teams. Therefore, it is a many to many relationship.
Country	In this entity, it only involves country ID and name.	It relates to both player and match by using countryID as foreign key. Each player has only one nationality, but a country has multiple players. So there is a one-to-many relationship between them. At the same time, a country can participate in multiple games, and each game has one home country team. So there is a one-to-many relationship between them.
Match	Match entity includes the CountryID, the start date, ID.	It relates to the country. Each country could hold or not hold many Matches but each Match must be held in one country. It is a one-to-many relationship.
Player-Position	This entity is such an associate entity. It combines player and position by using positionID and playerID.	Because the relationship between player and position is many-to-many, we create an associate entity as normalization to reach the 3rd normalized form.
Position	In this entity, it only has an ID and name.	Position entity is related to Player entity indeed. One player can play multiple positions, and a position can also be played by multiple players. Consequently, it is a many-to-many relationship.