EDABA – Laboratory + Tutorial

Project – Polish Volleyball League

Description and aim of Polish volleybal league database

The aim of this task is to prepare database simulating activity of a real organization, company, shop and such like. I came up with an idea to store information about Polish Volleyball League. This database should contain data referring to players, coaches, teams, sports hall, main referees, game. The main aim of this solution is to facilitate easy way of displaying historical data. I ensure access for a single season data namely 2020/2021. I would like to retrieve information about results of the games. The league consists of 14 teams every team plays 2 times with each other – once as host of the game and once as the guest of the game. It means that there are 182 games in total. Each team can have at maximum 14 players. Coaches can have different roles such as first coach, assistant, physical preparation coach. When it comes to referees, I would take into consideration only main referees. Each team has the specific sports hall where the host games are played. Game entity stores information about host and guests of the match. The score of the team belongs to the range <0;3>. The winning team scores 2 or 3 points, the losing team scores 0 or 1 point.

Structure of database

Entities:

1. Team

- **Name #*** (name of the team, primary key, required value)
- **Website** (official website of the team, non mandatory property because not each team needs this type of site)
- **Points** (The number which specifies how many points the particular team scored in total)
- **Main sponsor** (the name of company which is responsible for financial support of the team)
- **Matches won** (small number which gives information how many games were won by the team)
- **Matches lost** (small number which gives information how many game were lost by the team)

2. Player

- **Id #*** (Special identifier of player record, primary key, required value)
- **First Name** (property specifying first name of player)
- **Last Name** (property specifying last name of player)
- **Shirt Number** (property specifying the shirt number)
- **Position** (propery specifying the position in the team)
- **Age** (specifies how old the player is)

3. Coach

- **Id #*** (Special identifier of coach record, primary key, required value)
- **First Name** (property specifying first name of coach)
- **Last Name** (property specifying last name of coach)
- **Age** (property specifying the age of coach)
- **Role** (property which specifies the role of the coach whether it is first coach, assistant, physical preparation coach)

4. Game

- **Number of the game #*** (Special identifier of player record, small integer value)
- **Date** (Specifies the date of the game with time of start of the match)
- **Attendance** (number which describes how much spectators were present during the game)
- **Result** (3 characters representation of result between team in form „number:number" e.g. '3:1' or '2:3'. Numbers refer to number of sets winned by the specific team.)
- **Home Score** (number which specifies how many points were scored by host team. Score is equal to amount of points scored by specific team. Team can gain 3 points for winning match 3:0 or 3:1, 2 for winning 3:2, 1 for losing 2:3, 0 for losing)
- **Away Score** (number which specifies how many points were scored by guest team)

5. Main Referee

- **Id #*** (Special identifier of main referee, primary key, required value)
- **First Name** (property specifying first name of main referee)
- **Last Name** (property specifying last name of main referee)
- **Age** (property specifying the age of main referee)
- **License** (special document which proves the experience of the main referee)

6. Sports Hall

- **Name #*** (Special identifier of sports hall, primary key, required value, name of the hall)
- **City** (address information, it describes where the hall is located)
- **Street** (address information, it describes where the hall is located)
- **Capacity** (number which points out how many spectators can be inside the home field)

Relationships

Team<->Sports Hall (Each team can own exactly one sports hall. Each sports hall can be assigned to exactly one team. Therefore the relationship between team and sports hall is **one-to-one**)

Team<->Player (Player can play in exactly one team. Team can consists of 14 players at maximum, so many players can play in a specific team. That is why the relationship between team and player is **one-to-many**)

Team<->Coach (Each coach can work in exactly one team due to competition. Each team can have many cooaches with different roles namely: first coach, assistant, physical preparation coach. That is why relationship between team and coach is **one-to-many**)

Main referee<->Game (For each single game only one referee can be assigned. Each referee can be responsible for some different games. The relationship between those 2 entities is **one-to-many**)

Team<-Host->Game (Two teams can play in one game. One team is a host of the game. There can be exactly one host of the game. The particular team can host over a dozen games. Therefore the relations between host team and game is **one-to-many**)

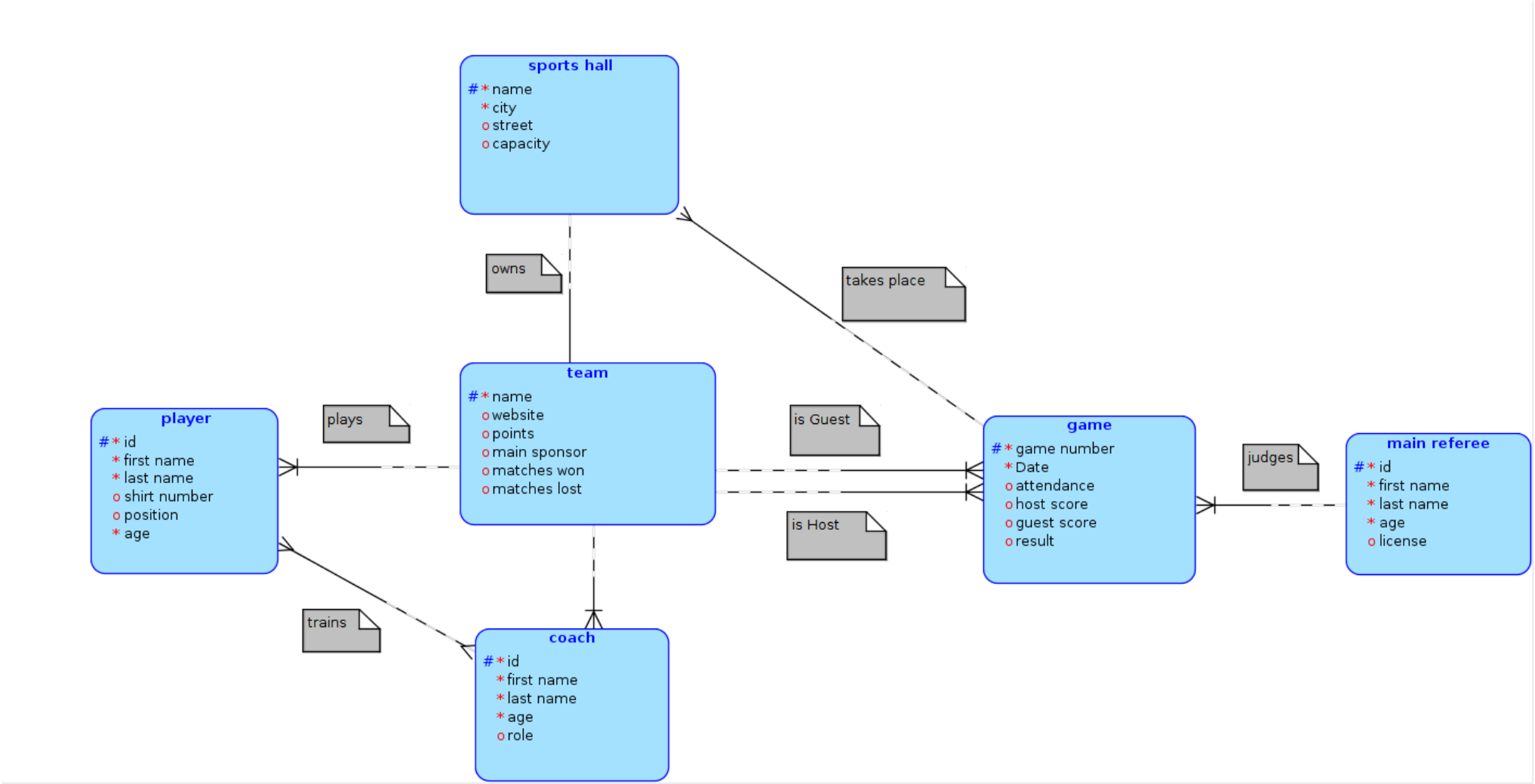
Team<-Guest->Game (Two teams can play in one game. One team is a guest of the game. There can be exactly one guest of the specific game. The particular team can be the guest over a dozen games. Therefore the relations between guest team and game is **one-to-many**)

First coach<->Player (Each first coach is responsible for group of players. Each player can be trained be only one first coach. That is why relationship between first coach and player is **one-to-many**)

Game<->Sports hall (Each game needs sports hall to hold a match. Game can take place in only one sports hall. Team can play many games in one sports hall. Therefore the relationship between those 2 entities is **one-to-many**).

Player<->Coach (Each coach is responsible for one team which consists of 14 players. Each player is trained by some different coaches like: first coach, assistant coach, physical preparation coach. Players are trained by some coaches and coach can train some players. Therefore the relationship between player and coach is **many-to-many**)

ER diagram


sports hall
#*name
*city
o street
o capacity
owns
takes place
team
#*name
o website
o points
o main sponsor
o matches won
o matches lost
plays
is Guest
is Host
player
#*id
*first name
*last name
o shirt number
o position
*age
game
#*game number
*Date
o attendance
o host score
o guest score
o result
judges
main referee
#*id
*first name
*last name
*age
o license
trains
coach
#*id
*first name
*last name
*age
o role

Description of mapping to relational schema

Sports hall (Sports hall has the name property which is the primary key of this entity. Sports hall is in one-to-one relationship with specific team. The information about the hall assigned to the given team is stored on team entity. Sports hall is also in relationship with game entity. Information about sports hall related with given game is stored on game entity.)

Team (Each team has its unique name which is marked as a primary key. This entity is in direct relation with 4 other entities. It stores information about the sports hall assigned to this team as a foreign key to relate those 2 entities. Other relationship properties as stored on remaining 3 entities namely: player, game, coach.)

Player (Each player has the id which is the primary key. This entity forms relationship with team. In team there can be many players but each player can play in only one team. Therefore each player stores information about the team in which he plays. This information is stored on player entity as a foreign key. Player forms also many-to-many relationship with coach entity. In order to form such relationship the additional entity **training** is used. Training entity is in one-to-many relationship with player entity.)

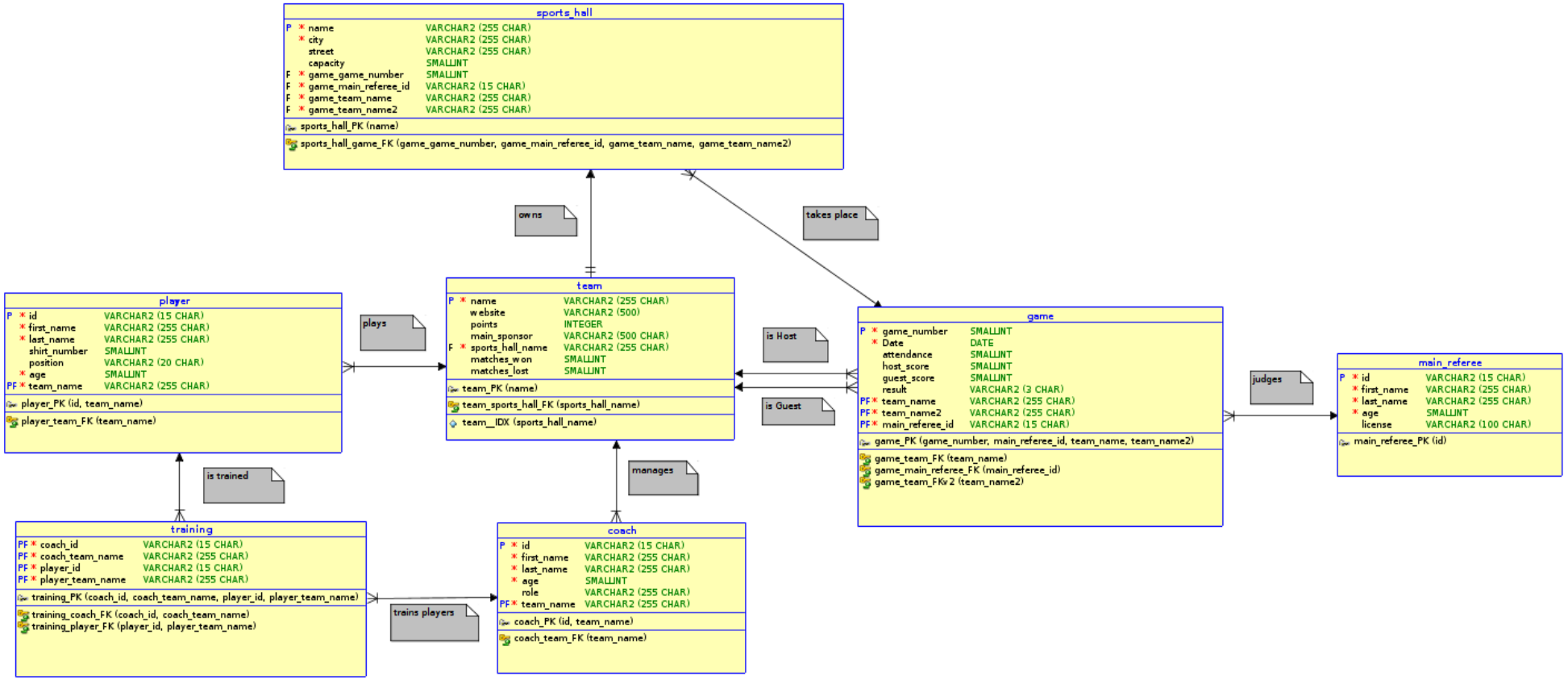
Game (This entity has 4 relationship. It stores information in which sports hall the game takes place, who is the host team and guest team of specific game and it stores information about main referee assigned to judge this game. All those values are stored as foreign keys on this entity in order to form one-to-many relationship with those entities.

Coach (Id of the coach is a primary key of this entity. Coach is in relation with team. Team can have many coaches but the particular coach can work only in one team. That is why each coach stores information about his employing club. This data is stored as foreign key on coach entity. Coach is in relation also with player entity. The foreign key is stored on player entity. Moreover coach entity forms **many-to-many** relationship with player entity. In order to form such relationship the additional entity **training** is used. Training entity is in one-to-many relationship with coach entity.)

Training (Additional entity which facilitates to create many-to-many relationship between coaches and players. It stores foreign keys both to coach and player entities.)

Main referee (This entity is in relation with game entity. The foreign key is stored in game entity. The primary key of main referee is Id of this referee.)

Engineering diagram – relational schema



DDL File

DDL file was generated using Oracle Database 21c type and it is attached as separate file.