**AMERICAN INTERNATIONAL UNIVERSITY BANGLADESH (AIUB)**

**FACULTY OF SCIENCE & TECHNOLOGY**

A picture containing calendar

Description automatically generated

Course Title

**INTRODUCTION TO DATABASE**

**Summer 2023-2024**

**Section: [M]**

**PROJECT TITLE**

TOURNAMENT MANAGEMENT SYSTEM

**Supervised By**

Md. Sajid Bin Faisal

**Submitted By: Group no: 01**

|  |  |
| --- | --- |
| **Name** | **ID** |
| **Sirajum Munir** | **22-48549-3** |
| **Anindita Bhattacharjee** | **22-48606-3** |
| **Fariar Rahman Arpa** | **22-49040-3** |
| **Min Shariar Digonto** | **23-50689-1** |

*Page no 1*

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **TOPICS** | | **Page no.** |
| **Title Page** | | **1** |
| **Table of Content** | | **2** |
| **1.** | **Introduction** | **3** |
| **2.** | **Case Study** | **3** |
| **3.** | **ER Diagram** | **4** |
| **4.** | **Normalization** | **5-6** |
| **5.** | **Finalization** | **7** |
| **6.** | **Table Creation** | **7-10** |
| **7.** | **Data Insertion** | **10-15** |
| **8.** | **Query Test** | **16-21** |
| **9.** | **Conclusion** |  |
|  |  |  |

*Page no 2*

**Introduction**

This project is about an online game tournament management system where data of different teams,their coaches,matches are stored orderly.This project ensures the smooth management of the online game tournament. While online gaming brings numerous economic benefits such as job creation, technological innovation, development of esports and so on. Online gaming stands as a powerhouse, driving economic growth and shaping modern entertainment landscapes worldwide.

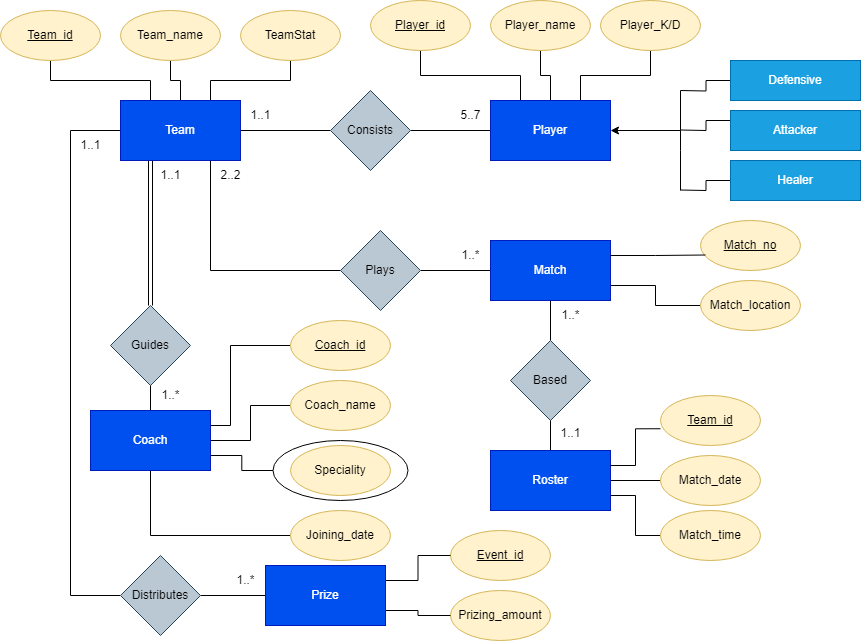
**Case Study**

**TOURNAMENT MANAGEMENT SYSTEM**

In a tournament management system, there is information about teams, matches, players and coaches. Each team has a team id, team name, team state. Teams consist of five players per team. There can be a maximum number of seven players in a team. Players have name, id and date of joining and specialties are also listed. Each team must have a minimum number of one coach. Coaches have coach\_id,coach\_name,coach\_joiningdate and speacilities.And each coach can only guide one team. There might be coaches without a team but there is no team without coaches. Teams play matches. Matches have match number, match location. Each team has many matches, but every match must have only two teams. Matches are based on rosters. Rosters have team id, match date and match time. In a roster, there can be information of many matches. Players can be specialized into 3 categories. These are defensive, attacker and healer. Prices are distributed among the teams. Each team can achieve many prizes and one prize can be distributed to one team.

*Page no 3*

**ER\_Diagram**



*Page no 4*

**Normalization**

**Plays:**

**UNF:** team\_slot, team\_id, team\_no, mat\_no, mat\_loc.

**1NF:** team\_slot, team\_id(PK), team\_no, mat\_no(PK), mat\_loc.

**2NF:**  1) team\_slot, team\_id(PK), team\_no

2) mat\_no, mat\_loc(PK), team\_id(FK)

**3NF:**  1) team\_slot, team\_id(PK), team\_no

2) mat\_no, mat\_loc(PK), team\_id(FK)

**Based:**

**UNF:** mat\_no,mat\_loc,team\_id,mat\_date,mat\_time

**1NF:** mat\_no(PK),mat\_loc,team\_id(PK),mat\_date,mat\_time

**2NF:** 1) mat\_no(PK),mat\_loc,team\_id(FK)

2) team\_id(PK),mat\_date,mat\_time

**3NF:** 1) mat\_no(PK),mat\_loc,team\_id(FK)

2) team\_id(PK),mat\_date,mat\_time

**Consists:**

**UNF:** pl\_name,pl\_id,pl\_K/D ratio,team\_no,team\_id,team\_stat

**1NF:** pl\_name,pl\_id(PK),pl\_K/D ratio,team\_no,team\_id(PK),team\_stat

**2NF:** 1) pl\_name,pl\_id(PK),pl\_K/D ratio, ,team\_id(FK)

2) team\_no, team\_id(PK),team\_stat

**3NF: :** 1) pl\_name,pl\_id(PK),pl\_K/D ratio, ,team\_id(FK)

2) team\_no, team\_id(PK),team\_stat

*Page no 5*

**Guides:**

**UNF:** team\_id,team\_no,team\_stat,co\_id,co\_name,co\_jd,co\_spe.

**1NF:** team\_id(PK),team\_no,team\_stat,co\_id(PK),co\_name,co\_jd,co\_spe.

**2NF:** 1) team\_id(PK),team\_no,team\_stat

2) co\_id(PK),co\_name,co\_jd,co\_spe, team\_id(FK),

**3NF:** 1) team\_id(PK),team\_no,team\_stat

2) co\_id(PK),co\_name,co\_jd,co\_spe, team\_id(FK),

**Distributes:**

**UNF:** team\_stat,team\_id,team\_no,p\_amount,event\_id

**1NF:** team\_stat,team\_id(PK),team\_no,p\_amount,event\_id(PK)

**2NF:** 1) team\_stat,team\_id(PK),team\_no

2) p\_amount,event\_id(PK), team\_id(FK),

**3NF:** 1) team\_stat,team\_id(PK),team\_no

2) p\_amount,event\_id(PK), team\_id(FK),

**Tables after normalization:**

1) team\_stat, team\_id(PK), team\_no

2) mat\_no(PK), mat\_loc, team\_id(FK)

3) mat\_no(PK),mat\_loc,team\_id(FK)

4) team\_id(PK),mat\_date,mat\_time

5) pl\_name,pl\_id(PK),pl\_K/D ratio, ,team\_id(FK)

6) team\_no, team\_id(PK),team\_stat

7) team\_id(PK),team\_no,team\_stat

8) co\_id(PK),co\_name,co\_jd,co\_spe, team\_id(FK)

9) team\_stat,team\_id(PK),team\_no

10) p\_amount,event\_id(PK), team\_id(FK),

*Page no 6*

**Final Tables(6 Tables):**

1) mat\_no(PK),mat\_loc,team\_id(FK)

2) team\_id(PK),mat\_date,mat\_time

3) pl\_name,pl\_id(PK),pl\_K/D ratio, ,team\_id(FK)

4) team\_id(PK),team\_no,team\_stat

5) co\_id(PK),co\_name,co\_jd,co\_spe, team\_id(FK)

6) p\_amount,event\_id(PK), team\_id(FK)

**Tables Creation:**

**TEAM**

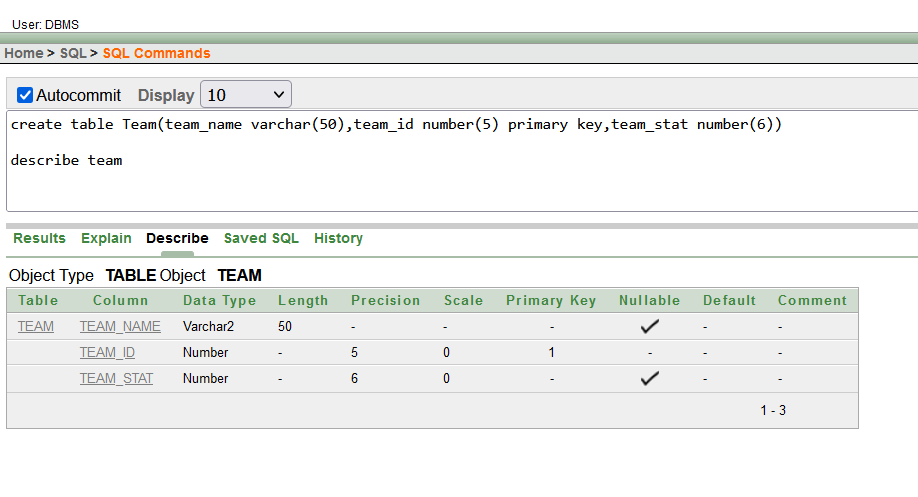
****

Fig 6.1 Team Table Creation

*Page no 7*

**CONSIST**

A screenshot of a computer

Description automatically generated

Fig 6.2 Consist Table Creation

**PLAY**

A screenshot of a computer

Description automatically generated

Fig 6.3 Play Table Creation

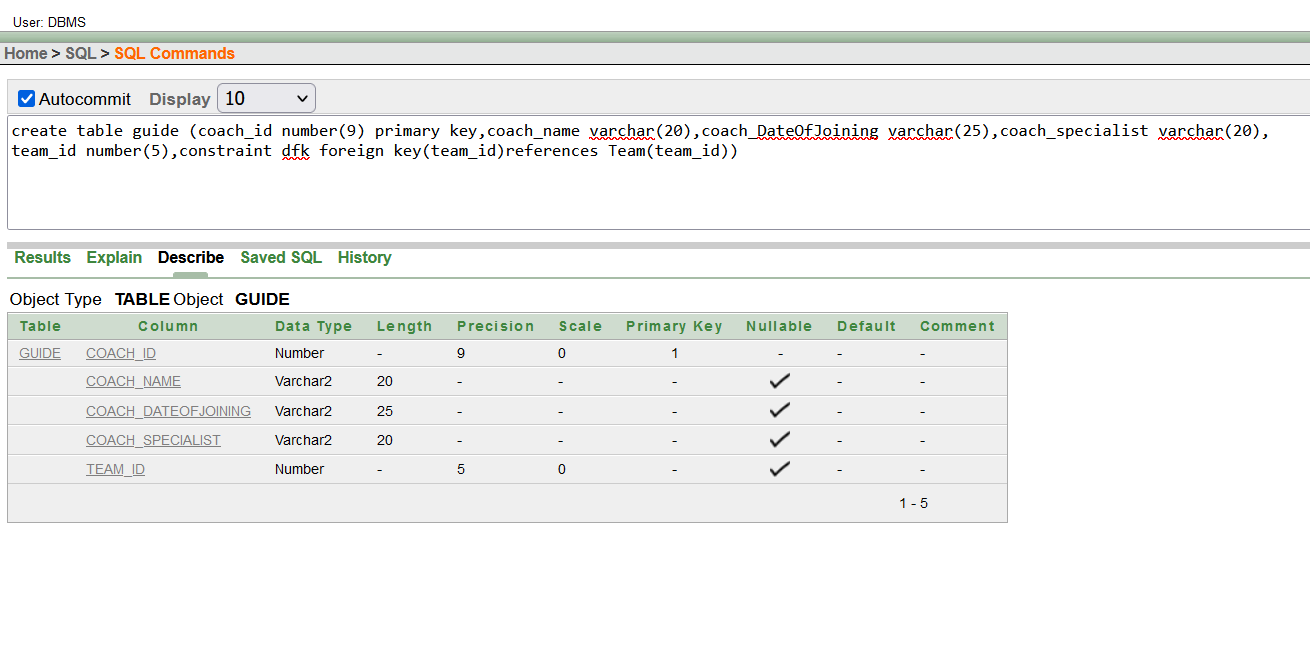
*Page no 8*

**ROSTER**

A screenshot of a computer

Description automatically generated

Fig 6.4 Roster Table Creation

**GUIDE**Fig 6.5 Guide Table Creation

*Page no 9*

**DISTRIBUTE**

A screenshot of a computer

Description automatically generated

Fig 6.6 Distribute Table Creation

**Value Insert:** A screenshot of a computer

Description automatically generated

Fig 7.1 Value insertion for Team Table

*Page no 10*

A screenshot of a computer

Description automatically generated

Fig 7.2 Value insertion for Consist Table

A screenshot of a computer

Description automatically generated

Fig 7.3 Value insertion for Play Table

*Page no 11*

A screenshot of a computer

Description automatically generated

Fig 7.4 Value insertion for Roster Table

A screenshot of a computer

Description automatically generated

Fig 7.5 Value insertion for Guide Table

*Page no 12*

A screenshot of a computer

Description automatically generated

Fig 7.6 Value insertion for Distribute Table

*Page no 13*

**Query Test:**

1.Simple Query:

Show the team name and team stat from table team where team id is 69.

A screenshot of a computer

Description automatically generated

Fig 8.1 Simple Query

*Page no 14*

2.Single Row Function

Show prizing amount, event id, team id and prizing amount \*5 replacing null with 0

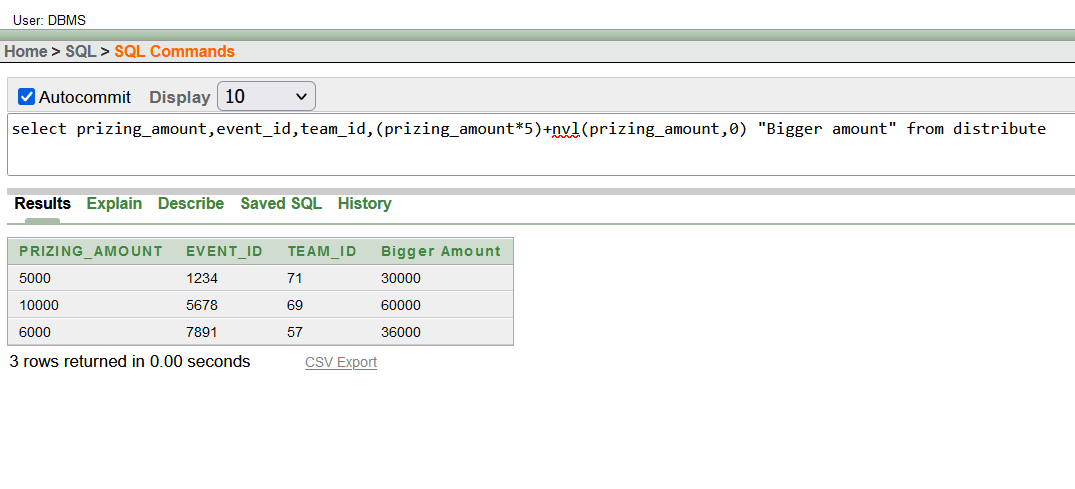


Fig 8.2 Single row function subquery

3.Group Function

Show the count of team id 71 from table play.

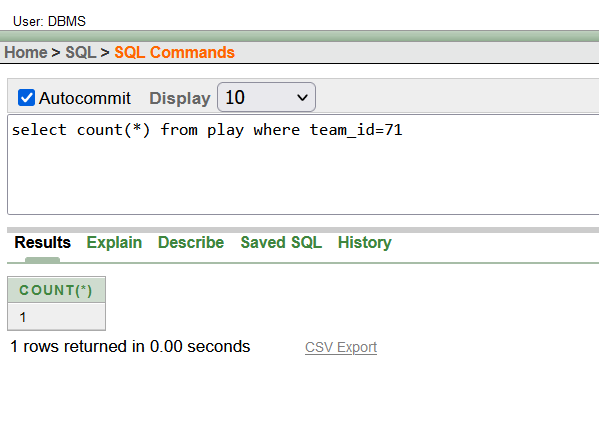


Fig 8.3 Group Function Query

*Page no 15*

4.Single Row Subquery

Show the team id, team stat higher than team Makarov from team table.

A screenshot of a computer

Description automatically generated

Fig 8.4 Single Row Subquery

5.Multiple Row Subquery

Show the coach id, name, date of joining where date of joining is higher than any other date of joining.

A screenshot of a computer

Description automatically generated

Fig 8.5 Multiple Row Subquery

*Page no 16*

6.Joining (Self-Join)

Show the self-join from consist of table.

A screenshot of a computer

Description automatically generated

Fig 8.6 Self Join Query

Equijoin

Show the match number, location, date from Play and Roster Table

A screenshot of a computer

Description automatically generated

Fig 8.7 EquiJoin Query

*Page no 17*

Simple view

Create a view names as prize where the prizing amount over 5000 will be shown over the columns event\_id,prizing\_amount and team\_id

A screenshot of a computer

Description automatically generated

Fig 8.8 Simple View Creation Command

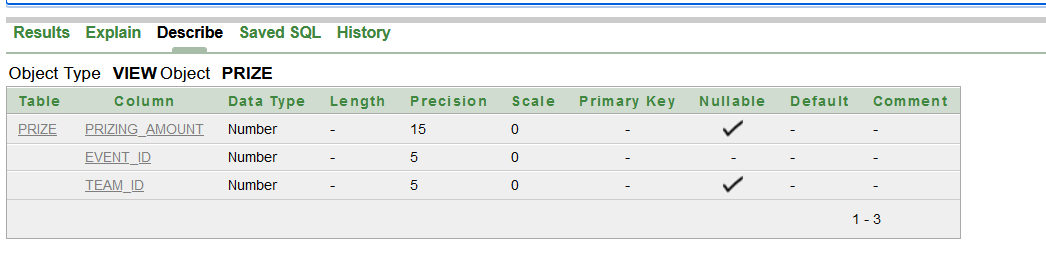


Fig 8.9 Description of the Simple View

A screenshot of a computer

Description automatically generated

Fig 8.10 Result of the simple view as a whole table

*Page no 18*

Complex View

Create a view named togthertables where team name,team stat and match date will be shown from table team and roster as they have team ID column in common.

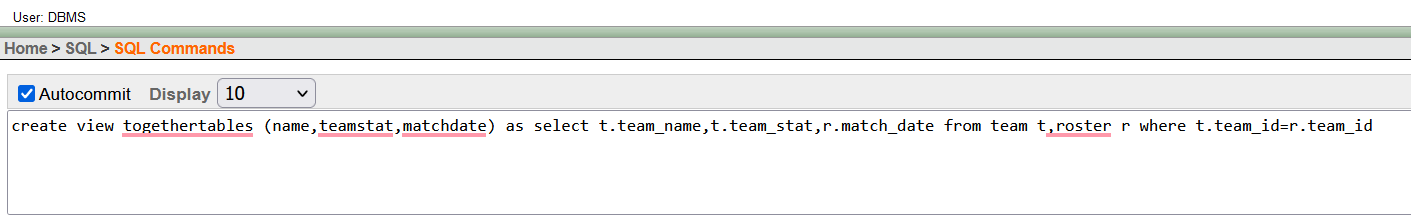


Fig 8.11 Complex view creation command

A screenshot of a computer

Description automatically generated

Fig 8.12 Description of the Complex view

A screenshot of a computer

Description automatically generated

Fig 8.13 Result of the Complex view as a whole table.

*Page no 19*

**Name: Anindita Bhattacharjee**

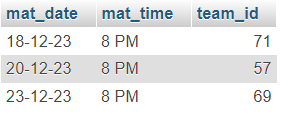
**DATABASE CONNECTION**

Procedure:

* Firstly, created a database named Tournament1 using XAMPP software and then started Apache and MySQL admin panel. Then, created two tables named Distribution and Roster.

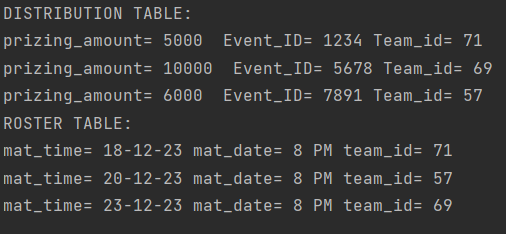
A screenshot of a computer

Description automatically generated



* After that, added a jar file (mysql-connector-java-8.0.28 version) to the project library classpath.
* Using IDE (IntelliJ IDEA), registered the driver(“com.mysql.cj.jdbc.Driver”).
* Connected the MySQL server with the IDE (IntelliJ IDEA) through Connection function.
* Created a statement object to execute SQL queries and connecting with the connection function to collect the data from the MySQL server to execute.
* Then, executed the SQL query using executeQuery() and store results in a ResultSet.
* After that, used the Connection Close() function for disconnecting database and the program executed with this output.

*Page no 20*

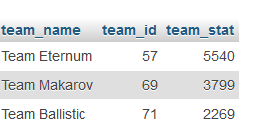


**Database Connection**

Name: Sirajum Munir

Procedure:

* Firstly, jar file “mysql-connector-j-8.2.0” was downloaded and kept in the C drive.
* XAMPP Control Panel v3.3.0 was downloaded and installed and from there Apache and MySql was started. From the Admin action, MySql was accessed.
* In MySQL, two tables named Team and Consist were created with value insertion.

A screenshot of a computer

Description automatically generated

*Page no 21*

* The NetBeans IDE was used and the jarfile was added to the project library class path.

A blue and white rectangular object

Description automatically generated

* The driver was registered with this line of code.



* MySQL server was connected with the java code through this line.



* Created a statement object st to Create statement.



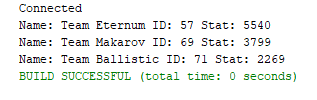
* Then Created an object rs of ResultSet and executed the query using executeQuery() to call all data from table.



* Then with a line of code the columns of the tables were printed.



* After that, used the Connection Close () function.
* Exception throwing statements were also used in this code to catch and throw any type of exceptions.

A close-up of a number

Description automatically generated *Page no 22*

Conclusion:

This project was about a Tournament management system and made by Oracle DBMS. Firstly, an ER diagram was made then the ER diagram was normalized to the finalization point and from those final tables, Database was created in Oracle with multiple commands and then value was inserted. Several queries were tested. Then the project tables were connected to Java code with several steps individually by every team member. This was the project overall.