

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ  
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ  
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»  
Кафедра ИИТ

ЛАБОРАТОРНАЯ РАБОТА №9

По дисциплине: «Современные платформы программирования»

Выполнил:

Студент ФЭИС

3-го курса, группы ПО-5

Прокопчик Е.А.

Проверил:

Крощенко А.А.

Брест 2022

Цель работы: приобрести практические навыки разработки баз данных и начальной интеграции БД с кодом Java с помощью JDBC

Общее задание:

Реализовать базу данных из не менее 5 таблиц на заданную тематику. При реализации продумать типизацию полей и внешние ключи в таблицах. Визуализировать разработанную БД с помощью схемы, на которой отображены все таблицы и связи между ними (пример, схема на рис. 1). На языке Java с использованием JDBC реализовать подключение к БД и выполнить основные типы запросов, продемонстрировать результаты преподавателю и включить тексты составленных запросов в отчет. Основные типы запросов – 1. На выборку/на выборку с упорядочиванием (SELECT); 2. На добавление (INSERT INTO); 3. На удаление (DELETE FROM); 4. На модификацию (UPDATE). Базу данные можно реализовать в любой СУБД (MySQL, PostgreSQL, SQLite и др.)

12) База данных «Европейские футбольные чемпионаты»

## Код программы

### DBWorker.java

```
package com.spplab9.database;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class DBWorker {
    private final String URL = "jdbc:mysql://localhost:3306/lab9spp";
    private final String USERNAME = "root";
    private final String PASSWORD = "root";

    private Connection connection;

    public Connection getConnection() {return connection;}
    public void setConnection(Connection connection) {this.connection = connection;}

    public DBWorker() {
        try{
            connection = DriverManager.getConnection(URL, USERNAME,
PASSWORD);
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

## Team.java

```
package com.spplab9.database;

public class Team {

    private int team_id;
    private int league_id;
    private String team_name;
    private String country;

    public Team() {
    }

    public Team(int league_id, String team_name, String country) {
        this.league_id = league_id;
        this.team_name = team_name;
        this.country = country;
    }

    public Team(int team_id, int league_id, String team_name, String country)
    {
        this.team_id = team_id;
        this.league_id = league_id;
        this.team_name = team_name;
        this.country = country;
    }

    public int getTeam_id() {return team_id;}
    public void setTeam_id(int team_id) {this.team_id = team_id;}

    public int getLeague_id() {return league_id;}
    public void setLeague_id(int league_id) {this.league_id = league_id;}

    public String getTeam_name() {return team_name;}
    public void setTeam_name(String team_name) {this.team_name = team_name;}

    public String getCountry() {return country;}
    public void setCountry(String country) {this.country = country;}

    @Override
    public String toString() {
        return getClass().getSimpleName() + "{id: " + team_id
            + ", league_id: " + league_id
            + ", team_name: " + team_name
            + ", country: " + country
            + "}";
    }
}
```

## Match.java

```
package com.spplab9.database;

import java.sql.Date;

public class Match {

    private int match_id;
    private String league_name;
    private String stadium_name;
    private String team1 name;
```

```

private String team2_name;
private Date match_date;

public Match(){

}

    public Match(String league_name, String stadium_name, String team1_name,
String team2_name, Date match_date){

        this.league_name = league_name;
        this.stadium_name = stadium_name;
        this.team1_name = team1_name;
        this.team2_name = team2_name;
        this.match_date = match_date;
    }

    public Match(int match_id, String league_name, String stadium_name,
String team1_name, String team2_name, Date match_date ){

        this.match_id = match_id;
        this.league_name = league_name;
        this.stadium_name = stadium_name;
        this.team1_name = team1_name;
        this.team2_name = team2_name;
        this.match_date = match_date;
    }

    public int getMatch_id() {return match_id;}
    public void setMatch_id(int match_id) {this.match_id = match_id;}

    public String getLeague_name() {return league_name;}
    public void setLeague_name(String league_name) {this.league_name =
league_name;}

    public String getStadium_name() {return stadium_name;}
    public void setStadium_name(String stadium_name) {this.stadium_name =
stadium_name;}

    public String getTeam1_name() {return team1_name;}
    public void setTeam1_name(String team1_name) {this.team1_name =
team1_name;}

    public String getTeam2_name() {return team2_name;}
    public void setTeam2_name(String team2_name) {this.team2_name =
team2_name;}

    public Date getMatch_date() {return match_date;}
    public void setMatch_date(Date match_date) {this.match_date =
match_date;}

    @Override
    public String toString() {
        return getClass().getSimpleName() + "{id: " + match_id
            + ", league_name: " + league_name
            + ", stadium_name: " + stadium_name
            + ", team1_name: " + team1_name
            + ", team2_name: " + team2_name
            + ", match_date: " + match_date
            + "}";
    }
}

```

## Main.java

```
package com.spplab9.database;

import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

public class Main {

    public static void main(String[] args) {
        DBWorker worker = new DBWorker();

        String querySSelect = "select * from teams";
        try {
            Statement statement = worker.getConnection().createStatement();
            ResultSet resultSet = statement.executeQuery(querySSelect);

            while(resultSet.next()) {
                Team team = new Team();
                team.setTeam_id(resultSet.getInt("team_id"));
                team.setLeague_id(resultSet.getInt("league_id"));
                team.setTeam_name(resultSet.getString("team_name"));
                team.setCountry(resultSet.getString("country"));

                System.out.println(team);
            }
            System.out.println("-----");
        } catch (SQLException e) {
            e.printStackTrace();
        }

        try {
            Statement statement = worker.getConnection().createStatement();
            statement.execute("INSERT INTO matches(league_name, stadium_name, team1_name, team2_name, match_date) VALUES ('Belarusian Premier League', 'Barysaŭ-Arena', 'BATE', 'Dynamo Minsk', '2022.03.01')");
            statement.execute("INSERT INTO matches(league_name, stadium_name, team1_name, team2_name, match_date) VALUES ('Belarusian Premier League', 'Barysaŭ-Arena', 'BATE', 'Dynamo Minsk', '2018.01.01')");
        } catch (SQLException e) {
            e.printStackTrace();
        }

        String querySelectSort = "select * from matches order by match_date";
        try {
            Statement statement = worker.getConnection().createStatement();
            ResultSet resultSet = statement.executeQuery(querySelectSort);

            while(resultSet.next()) {
                Match match = new Match();
                match.setMatch_id(resultSet.getInt("match_id"));
                match.setLeague_name(resultSet.getString("league_name"));
                match.setStadium_name(resultSet.getString("stadium_name"));
                match.setTeam1_name(resultSet.getString("team1_name"));
                match.setTeam2_name(resultSet.getString("team2_name"));
                match.setMatch_date(resultSet.getDate("match_date"));

                System.out.println(match);
            }
            System.out.println("-----");
        } catch (SQLException e) {
            e.printStackTrace();
        }
    }
}
```

```

    } catch (SQLException e) {
        e.printStackTrace();
    }

    try {
        Statement statement = worker.getConnection().createStatement();
        statement.execute("DELETE FROM matches WHERE match_date = '2018.01.01'");
        statement.execute("UPDATE matches SET match_date = '2022.07.01' WHERE match_date = '2022.03.01'");

        ResultSet resultSet = statement.executeQuery(querySelectSort);

        while(resultSet.next()) {
            Match match = new Match();
            match.setMatch_id(resultSet.getInt("match_id"));
            match.setLeague_name(resultSet.getString("league_name"));
            match.setStadium_name(resultSet.getString("stadium_name"));
            match.setTeam1_name(resultSet.getString("team1_name"));
            match.setTeam2_name(resultSet.getString("team2_name"));
            match.setMatch_date(resultSet.getDate("match_date"));

            System.out.println(match);
        }
        System.out.println("-----");
    } catch (SQLException e) {
        e.printStackTrace();
    }
}
}

```

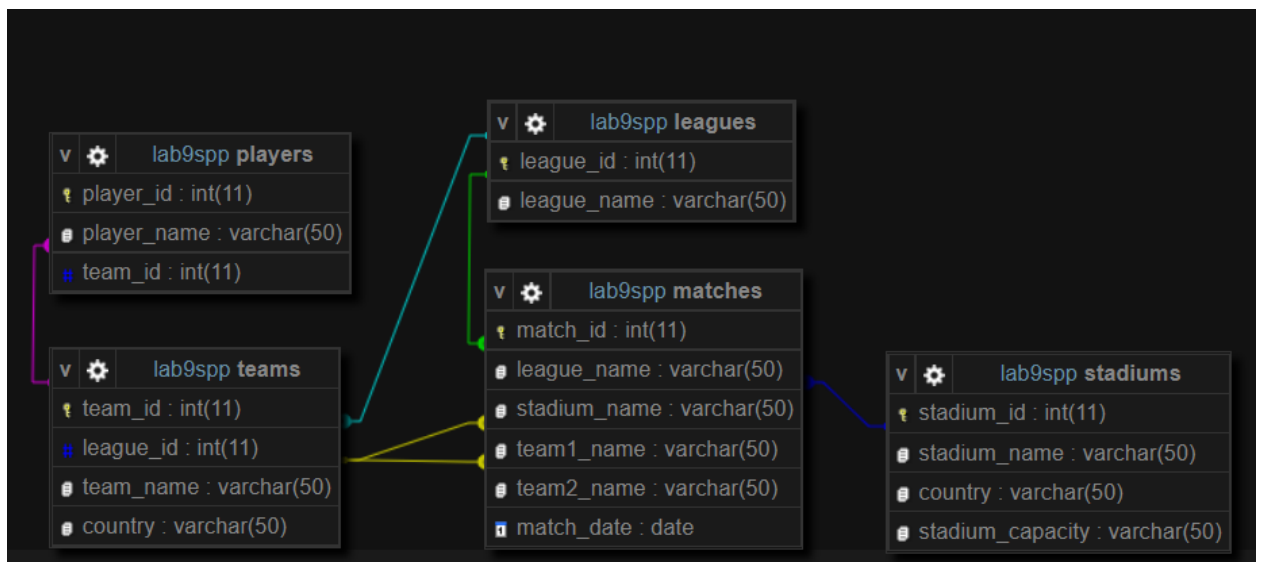


Схема БД

Вывод: приобрел практические навыки разработки баз данных и начальной интеграции БД с кодом Java с помощью JDBC