

Course Title:	Fundamentals of Data Engineering
Course Number:	COE 848
Semester/Year (e.g.F2016)	W2024

Instructor:	Dr. Faezeh Ensan
--------------------	------------------

Assignment/Lab Number:	Lab 5
Assignment/Lab Title:	Database Programming

Submission Date:	March 28, 2024
Due Date:	March 28, 2024

Student LAST Name	Student FIRST Name	Student Number	Section	Signature*
Saini	Harsanjam	501055402	04	Harsanjam

*By signing above you attest that you have contributed to this written lab report and confirm that all work you have contributed to this lab report is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a "0" on the work, an "F" in the course, or possibly more severe penalties, as well as a Disciplinary Notice on your academic record under the Student Code of Academic Conduct, which can be found online at: <http://www.ryerson.ca/senate/current/pol60.pdf>

Java Database Programming for Queries

Steps for running the program:

Check Database File: Ensure that the uclTourney.db file exists in the correct location and contains the tables you created. You can use a SQLite database management tool or command-line interface to verify this.

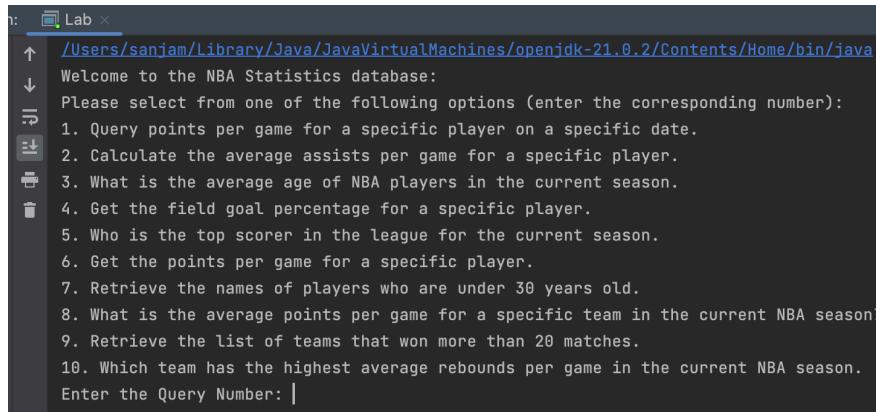
Download SQLite JDBC Driver: First, you need to download the SQLite JDBC driver. You can download it from the official SQLite website or from a reliable Maven repository.

Add Driver to Classpath: After downloading the SQLite JDBC driver, add it to your Java project's classpath. This can be done by either placing the JAR file in your project's lib directory and configuring your IDE build path to include it, or by using a build tool like Maven or Gradle to manage dependencies.

Load the Driver: Before connecting to the SQLite database, you need to load the JDBC driver using ***Class.forName() method***. This step is essential as it registers the driver with the DriverManager, allowing Java to find and use it.

Connection String: Double-check the connection string used in your Java code (***jdbc:sqlite:uclTourney.db***) to ensure it correctly points to the uclTourney.db file.

Initial output when code runs:



```
1: Lab x
  /Users/sanjam/Library/Java/JavaVirtualMachines/openjdk-21.0.2/Contents/Home/bin/java
  Welcome to the NBA Statistics database:
  Please select from one of the following options (enter the corresponding number):
  1. Query points per game for a specific player on a specific date.
  2. Calculate the average assists per game for a specific player.
  3. What is the average age of NBA players in the current season.
  4. Get the field goal percentage for a specific player.
  5. Who is the top scorer in the league for the current season.
  6. Get the points per game for a specific player.
  7. Retrieve the names of players who are under 30 years old.
  8. What is the average points per game for a specific team in the current NBA season?
  9. Retrieve the list of teams that won more than 20 matches.
  10. Which team has the highest average rebounds per game in the current NBA season.
  Enter the Query Number: |
```

The program operates within a continuous loop, allowing users to choose options numbered 1 through 10 to execute various queries. Should users wish to terminate the program, they simply need to input 0.

Switch statements govern the program's flow, enabling it to respond to user input by executing the corresponding action. Each option is associated with a particular case within the switch statement. If the user selects option 0, the program concludes its execution.

10 Query Result

1. To select the points per game for a specific player on a specific date:

```
SELECT PointsPerGame
FROM Player
JOIN Player_Game ON Player.PlayerID = Player_Game.PlayerID
JOIN Game ON Player_Game.GameID = Game.GameID
WHERE Player.Name = 'LeBron James' -- Replace 'SpecificPlayerName' with the
player's actual name
AND Game.Date = '2024-03-10'; -- Replace 'YYYY-MM-DD' with the actual date
```

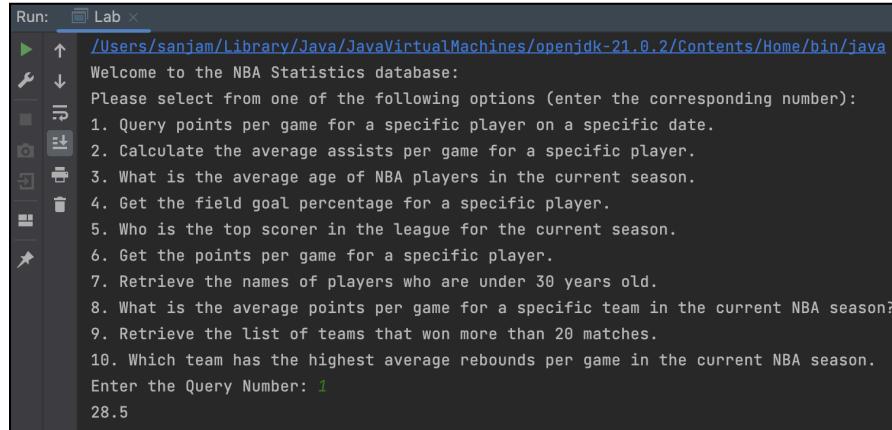
```
sqlite> SELECT PointsPerGame
FROM Player
JOIN Player_Game ON Player.PlayerID = Player_Game.PlayerID
JOIN Game ON Player_Game.GameID = Game.GameID
WHERE Player.Name = 'LeBron James' --- Replace 'SpecificPlayerName' with the
AND Game.Date = '2024-03-10'; --- Replace 'YYYY-MM-DD' with the actual date
28.5
```

Explanation - This query retrieves the points per game for a particular player (in this example, LeBron James) on a specific date (March 10, 2024). It joins the Player, Player_Game, and Game tables on their respective IDs, and then filters the results based on the player's name and the game date.

Java Code:

```
// Method to execute Query #2: Calculate the average assists per game for a specific player
public static void query2() {
    try {
        Class.forName(className:"org.sqlite.JDBC");
        Connection c = DriverManager.getConnection(url:"jdbc:sqlite:uclTourney.db");
        Statement stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery("SELECT AVG(AssistsPerGame) " +
            "FROM Player " +
            "WHERE Name = 'Stephen Curry'");
        while (rs.next()) {
            double avgAssistsPerGame = rs.getDouble(columnIndex:1);
            System.out.println("Average assists per game for Stephen Curry: " + avgAssistsPerGame);
        }
        rs.close();
        stmt.close();
        c.close();
    } catch (Exception e) {
        System.err.println(e.getClass().getName() + ": " + e.getMessage());
        System.exit(status:0);
    }
}
```

Output Result:



The screenshot shows a Java application window titled "Lab". The application is running from the command line at "/Users/sanjam/Library/Java/JavaVirtualMachines/openjdk-21.0.2/Contents/Home/bin/java". The application displays a menu of 10 options related to NBA statistics. The user has selected option 1, which asks for points per game for a specific player on a specific date. The user has entered the value "28.5" and pressed enter.

2. To calculate the average assists per game for a specific player:

```
SELECT AVG(AssistsPerGame)
FROM Player
WHERE Name = 'Stephen Curry'; -- Replace 'SpecificPlayerName' with the player name
```

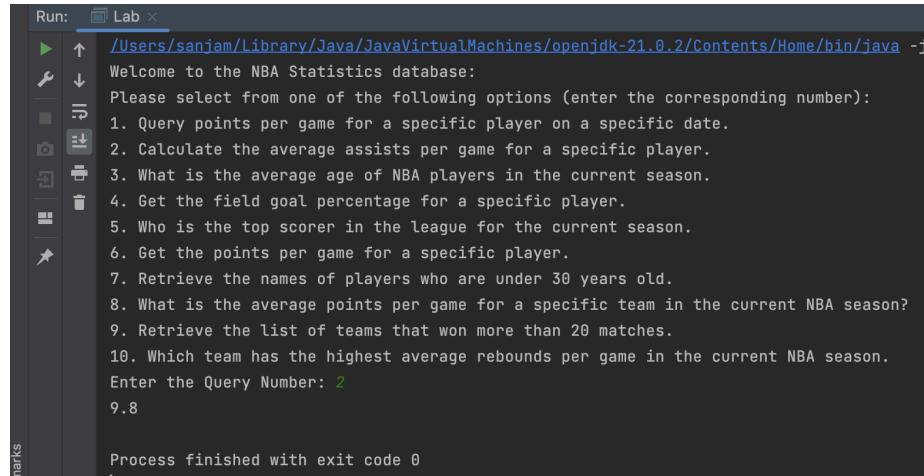
```
sqlite> SELECT AVG(AssistsPerGame)
FROM Player
[WHERE Name = 'Stephen Curry'; -- Replace 'SpecificPlayerName' with the player's actual name
9.8
```

Explanation - This query calculates the average assists per game for a specific player (in this case, Stephen Curry) by selecting the player's assists per game from the Player table and using the AVG function to calculate the average.

Java Code:

```
// Method to execute Query #1: Select points per game for a specific player on a specific date
public static void query1() {
    try {
        Class.forName(className:"org.sqlite.JDBC");
        Connection c = DriverManager.getConnection(url:"jdbc:sqlite:uclTourney.db");
        Statement stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery("SELECT PointsPerGame " +
            "FROM Player " + // Add space after table name
            "JOIN Player_Game ON Player.PlayerID = Player_Game.PlayerID " +
            "JOIN Game ON Player_Game.GameID = Game.GameID " +
            "WHERE Player.Name = 'LeBron James' AND Game.Date = '2024-03-10'");
        while (rs.next()) {
            double pointsPerGame = rs.getDouble(columnLabel:"PointsPerGame");
            System.out.println("Points per game for LeBron James on 2024-03-10: " + pointsPerGame);
        }
        rs.close();
        stmt.close();
        c.close();
    } catch (Exception e) {
        System.err.println(e.getClass().getName() + ": " + e.getMessage());
        System.exit(status:0);
    }
}
```

Output Result:



```
Welcome to the NBA Statistics database:  
Please select from one of the following options (enter the corresponding number):  
1. Query points per game for a specific player on a specific date.  
2. Calculate the average assists per game for a specific player.  
3. What is the average age of NBA players in the current season.  
4. Get the field goal percentage for a specific player.  
5. Who is the top scorer in the league for the current season.  
6. Get the points per game for a specific player.  
7. Retrieve the names of players who are under 30 years old.  
8. What is the average points per game for a specific team in the current NBA season?  
9. Retrieve the list of teams that won more than 20 matches.  
10. Which team has the highest average rebounds per game in the current NBA season.  
Enter the Query Number: 2  
9.8  
Process finished with exit code 0
```

3. What is the average age of NBA players in the current season

```
SELECT AVG(Age)  
FROM Player;
```

```
sqlite> SELECT AVG(Age)  
FROM Player;  
27.7142857142857
```

Explanation - This query calculates the average age of NBA players in the current season by selecting the average of the Age column from the Player table.

Java Code:

```
// Method to execute Query #3: What is the average age of NBA players in the current season  
public static void query3() {  
    try {  
        Class.forName(className:"org.sqlite.JDBC");  
        Connection c = DriverManager.getConnection(url:"jdbc:sqlite:uclTourney.db");  
        Statement stmt = c.createStatement();  
        ResultSet rs = stmt.executeQuery(sql:"SELECT AVG(Age) FROM Player");  
        while (rs.next()) {  
            double avgAge = rs.getDouble(columnIndex:1);  
            System.out.println("Average age of NBA players: " + avgAge);  
        }  
        rs.close();  
        stmt.close();  
        c.close();  
    } catch (Exception e) {  
        System.err.println(e.getClass().getName() + ": " + e.getMessage());  
        System.exit(status:0);  
    }  
}
```

Output Result:



```
Run: Lab x
/Users/sanjam/Library/Java/JavaVirtualMachines/openjdk-21.0.2/Contents/Home/bin/java -javaw
Welcome to the NBA Statistics database:
Please select from one of the following options (enter the corresponding number):
1. Query points per game for a specific player on a specific date.
2. Calculate the average assists per game for a specific player.
3. What is the average age of NBA players in the current season.
4. Get the field goal percentage for a specific player.
5. Who is the top scorer in the league for the current season.
6. Get the points per game for a specific player.
7. Retrieve the names of players who are under 30 years old.
8. What is the average points per game for a specific team in the current NBA season?
9. Retrieve the list of teams that won more than 20 matches.
10. Which team has the highest average rebounds per game in the current NBA season.
Enter the Query Number: 3
27.7142857142857
```

4. To get the field goal percentage for a specific player:

```
SELECT FieldGoalPercentage
FROM Player
WHERE Name = 'Chris Paul'; -- Replace 'SpecificPlayerName' with the player's name
```

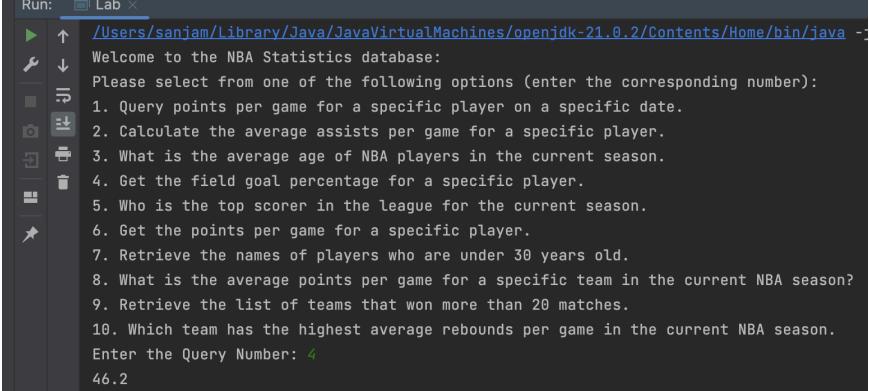
```
sqlite> SELECT FieldGoalPercentage
FROM Player
WHERE Name = 'Chris Paul'; -- Replace 'SpecificPlayerName' with the player's actual name
46.2
```

Explanation - This query retrieves the field goal percentage for a specific player (Chris Paul) by selecting the FieldGoalPercentage column from the Player table where the player's name matches.

Java Code:

```
// Method to execute Query #4: Get the field goal percentage for a specific player
public static void query4() {
    try {
        Class.forName(className:"org.sqlite.JDBC");
        Connection c = DriverManager.getConnection(url:"jdbc:sqlite:uclTourney.db");
        Statement stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery(sql:"SELECT FieldGoalPercentage FROM Player WHERE Name = 'Chris Paul'");
        while (rs.next()) {
            double fieldGoalPercentage = rs.getDouble(columnLabel:"FieldGoalPercentage");
            System.out.println("Field goal percentage for Chris Paul: " + fieldGoalPercentage);
        }
        rs.close();
        stmt.close();
        c.close();
    } catch (Exception e) {
        System.err.println(e.getClass().getName() + ": " + e.getMessage());
        System.exit(status:0);
    }
}
```

Output Result:



Welcome to the NBA Statistics database:
Please select from one of the following options (enter the corresponding number):
1. Query points per game for a specific player on a specific date.
2. Calculate the average assists per game for a specific player.
3. What is the average age of NBA players in the current season.
4. Get the field goal percentage for a specific player.
5. Who is the top scorer in the league for the current season.
6. Get the points per game for a specific player.
7. Retrieve the names of players who are under 30 years old.
8. What is the average points per game for a specific team in the current NBA season?
9. Retrieve the list of teams that won more than 20 matches.
10. Which team has the highest average rebounds per game in the current NBA season.
Enter the Query Number: 46.2

5. Who is the top scorer in the league for the current season

```
SELECT Name  
FROM Player  
WHERE PointsPerGame = (SELECT MAX(PointsPerGame) FROM Player);
```

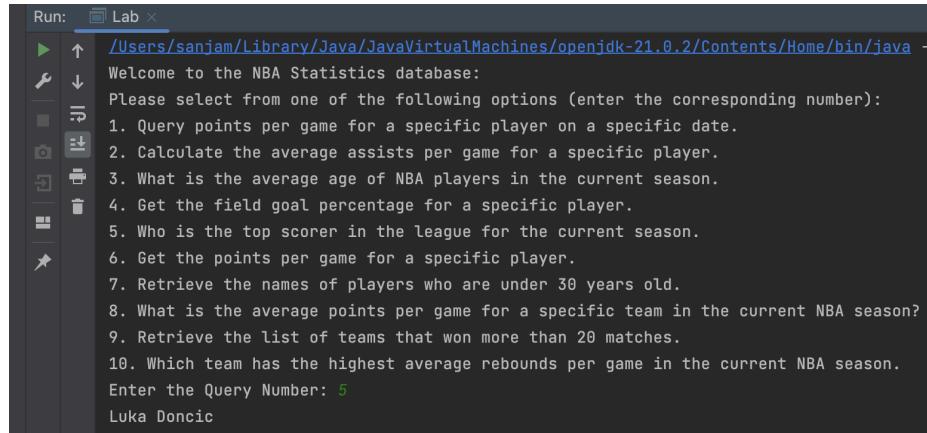
```
sqlite> SELECT Name  
FROM Player  
WHERE PointsPerGame = (SELECT MAX(PointsPerGame) FROM Player);  
Luka Doncic
```

Explanation - This query retrieves the name of the player who has the maximum points per game in the current season by selecting the player's name from the Player table where the PointsPerGame equals the maximum points per game in the whole league.

Java Code:

```
// Method to execute Query #5: Who is the top scorer in the league for the current season  
public static void query5() {  
    try {  
        Class.forName("org.sqlite.JDBC");  
        Connection c = DriverManager.getConnection(url:"jdbc:sqlite:uclTourney.db");  
        Statement stmt = c.createStatement();  
        ResultSet rs = stmt.executeQuery(sql:"SELECT Name FROM Player WHERE PointsPerGame = (SELECT MAX(PointsPerGame) FROM Player)");  
        while (rs.next()) {  
            String topScorer = rs.getString(columnLabel:"Name");  
            System.out.println("Top scorer in the league: " + topScorer);  
        }  
        rs.close();  
        stmt.close();  
        c.close();  
    } catch (Exception e) {  
        System.err.println(e.getClass().getName() + ": " + e.getMessage());  
        System.exit(status:0);  
    }  
}
```

Output Result:



Welcome to the NBA Statistics database:
Please select from one of the following options (enter the corresponding number):
1. Query points per game for a specific player on a specific date.
2. Calculate the average assists per game for a specific player.
3. What is the average age of NBA players in the current season.
4. Get the field goal percentage for a specific player.
5. Who is the top scorer in the league for the current season.
6. Get the points per game for a specific player.
7. Retrieve the names of players who are under 30 years old.
8. What is the average points per game for a specific team in the current NBA season?
9. Retrieve the list of teams that won more than 20 matches.
10. Which team has the highest average rebounds per game in the current NBA season.
Enter the Query Number: 5
Luka Doncic

6. To get the points per game for a specific player:

```
SELECT PointsPerGame  
FROM Player  
WHERE Name = 'Anthony Davis'; -- Replace 'SpecificPlayerName' with the player name
```

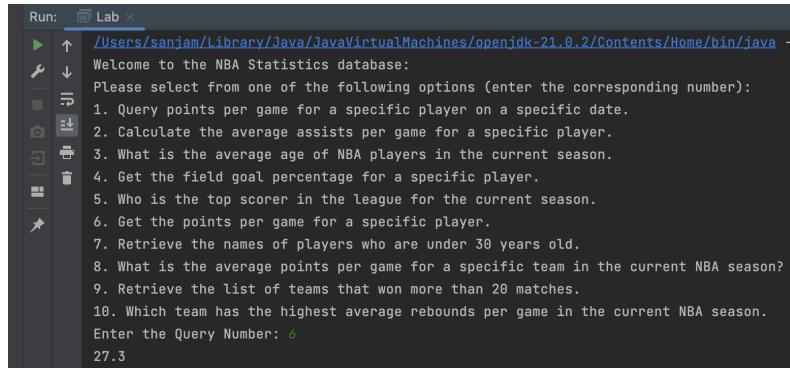
```
sqlite> SELECT PointsPerGame  
FROM Player  
WHERE Name = 'Anthony Davis';  
27.3
```

Explanation - This query retrieves the points per game for a specific player (here, Anthony Davis) by selecting the PointsPerGame column from the Player table where the player's name matches.

Java Code:

```
// Method to execute Query #6: Get the points per game for a specific player  
public static void query6() {  
    try {  
  
        Class.forName(className:"org.sqlite.JDBC");  
        Connection c = DriverManager.getConnection(url:"jdbc:sqlite:uclTourney.db");  
        Statement stmt = c.createStatement();  
        ResultSet rs = stmt.executeQuery(sql:"SELECT PointsPerGame FROM Player WHERE Name = 'Anthony Davis'");  
        while (rs.next()) {  
            double pointsPerGame = rs.getDouble(columnLabel:"PointsPerGame");  
            System.out.println("Points per game for Anthony Davis: " + pointsPerGame);  
        }  
        rs.close();  
        stmt.close();  
        c.close();  
    } catch (Exception e) {  
        System.err.println(e.getClass().getName() + ": " + e.getMessage());  
        System.exit(status:0);  
    }  
}
```

Output Result:



```
Run: Lab
↑ /Users/sanjam/Library/Java/JavaVirtualMachines/openjdk-21.0.2/Contents/Home/bin/java -jar
↓
Welcome to the NBA Statistics database:
Please select from one of the following options (enter the corresponding number):
1. Query points per game for a specific player on a specific date.
2. Calculate the average assists per game for a specific player.
3. What is the average age of NBA players in the current season.
4. Get the field goal percentage for a specific player.
5. Who is the top scorer in the league for the current season.
6. Get the points per game for a specific player.
7. Retrieve the names of players who are under 30 years old.
8. What is the average points per game for a specific team in the current NBA season?
9. Retrieve the list of teams that won more than 20 matches.
10. Which team has the highest average rebounds per game in the current NBA season.
Enter the Query Number: 6
27.3
```

7. To retrieve the names of players who are under 30 years old

```
SELECT Name
FROM Player
WHERE Age < 30;
```

```
sqlite> SELECT Name
FROM Player
WHERE Age < 30;
Donovan Mitchell
Karl-Anthony Towns
Devin Vassell
Tyrese Haliburton
Jalen Suggs
Dejounte Murray
Cole Anthony
Cade Cunningham
Giannis Antetokounmpo
Anthony Davis
Luka Doncic
Devin Booker
Joel Embiid
```

Explanation - This query retrieves the names of players who are under 30 years old by selecting the names from the Player table where the Age column is less than 30.

Java Code:

```
// Method to execute Query #7: Retrieve the names of players who are under 30 years old
public static void query7() {
    try {
        Class.forName(className:"org.sqlite.JDBC");
        Connection c = DriverManager.getConnection(url:"jdbc:sqlite:uclTourney.db");
        Statement stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery(sql:"SELECT Name FROM Player WHERE Age < 30");
        while (rs.next()) {
            String playerName = rs.getString(columnLabel:"Name");
            System.out.println("Player under 30 years old: " + playerName);
        }
        rs.close();
        stmt.close();
        c.close();
    } catch (Exception e) {
        System.err.println(e.getClass().getName() + ": " + e.getMessage());
        System.exit(status:0);
    }
}
```

Output Result:

```
Run: Lab
/Users/sanjam/Library/Java/JavaVirtualMachines/openjdk-21.0.2/Contents/Home/bin/java -jav
Welcome to the NBA Statistics database:
Please select from one of the following options (enter the corresponding number):
1. Query points per game for a specific player on a specific date.
2. Calculate the average assists per game for a specific player.
3. What is the average age of NBA players in the current season.
4. Get the field goal percentage for a specific player.
5. Who is the top scorer in the league for the current season.
6. Get the points per game for a specific player.
7. Retrieve the names of players who are under 30 years old.
8. What is the average points per game for a specific team in the current NBA season?
9. Retrieve the list of teams that won more than 20 matches.
10. Which team has the highest average rebounds per game in the current NBA season.
Enter the Query Number: 7
Donovan Mitchell
Karl-Anthony Towns
Devin Vassell
Tyrese Haliburton
Jalen Suggs
Dejounte Murray
Cole Anthony
Cade Cunningham
Giannis Antetokounmpo
Anthony Davis
Luka Doncic
Devin Booker
Joel Embiid
```

8. What is the average points per game for a specific team in the current NBA season?

```
SELECT AVG(PointsPerGame)
FROM Player
WHERE TeamID = (SELECT TeamID FROM Team WHERE TeamName = 'Los
Angeles Lakers');
```

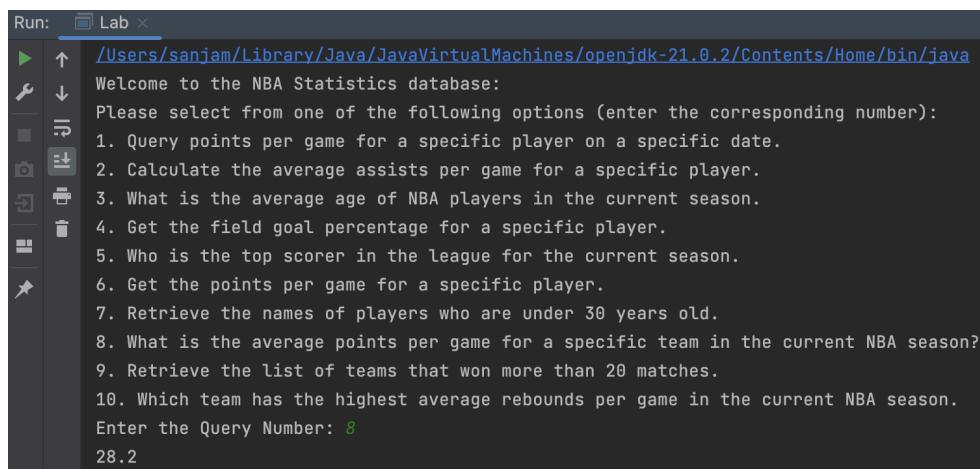
```
sqlite> SELECT AVG(PointsPerGame)
FROM Player
WHERE TeamID = (SELECT TeamID FROM Team WHERE TeamName = 'Los Angeles Lakers');
28.2
```

Explanation - This query calculates the average points per game for a specific team (here, Los Angeles Lakers) in the current NBA season by selecting the average of PointsPerGame from the Player table where the TeamID matches the TeamID of the Los Angeles Lakers team.

Java Code:

```
// Method to execute Query #8: What is the average points per game for a specific team in the current NBA season
public static void query8() {
    try {
        Class.forName(className:"org.sqlite.JDBC");
        Connection c = DriverManager.getConnection(url:"jdbc:sqlite:uclTourney.db");
        Statement stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery("SELECT AVG(PointsPerGame) " +
            "FROM Player " +
            "WHERE TeamID = (SELECT TeamID FROM Team WHERE TeamName = 'Los Angeles Lakers')");
        while (rs.next()) {
            double avgPointsPerGame = rs.getDouble(columnIndex:1);
            System.out.println("Average points per game for Los Angeles Lakers: " + avgPointsPerGame);
        }
        rs.close();
        stmt.close();
        c.close();
    } catch (Exception e) {
        System.err.println(e.getClass().getName() + ": " + e.getMessage());
        System.exit(status:0);
    }
}
```

Output Result:



```
Welcome to the NBA Statistics database:
Please select from one of the following options (enter the corresponding number):
1. Query points per game for a specific player on a specific date.
2. Calculate the average assists per game for a specific player.
3. What is the average age of NBA players in the current season.
4. Get the field goal percentage for a specific player.
5. Who is the top scorer in the league for the current season.
6. Get the points per game for a specific player.
7. Retrieve the names of players who are under 30 years old.
8. What is the average points per game for a specific team in the current NBA season?
9. Retrieve the list of teams that won more than 20 matches.
10. Which team has the highest average rebounds per game in the current NBA season.

Enter the Query Number: 8
28.2
```

9. To retrieve the list of teams that won more than 20 matches

```
SELECT TeamName
FROM Team
WHERE NumberofWins > 20;
```

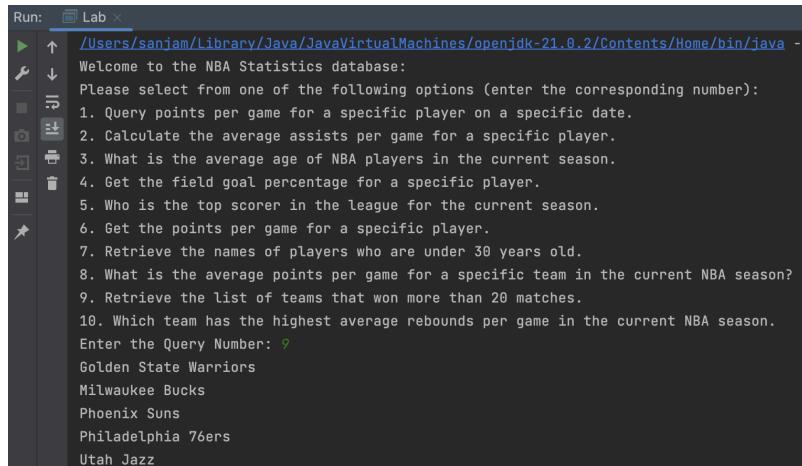
```
sqlite> SELECT TeamName
      FROM Team
     WHERE NumberofWins > 20;
Golden State Warriors
Milwaukee Bucks
Phoenix Suns
Philadelphia 76ers
Utah Jazz
```

Explanation - This query retrieves the names of teams that won more than 20 matches by selecting the TeamName from the Team table where the NumberofWins column is greater than 20.

Java Code:

```
// Method to execute Query #9: Retrieve the list of teams that won more than 20 matches
public static void query9() {
    try {
        Class.forName(className:"org.sqlite.JDBC");
        Connection c = DriverManager.getConnection(url:"jdbc:sqlite:uclTourney.db");
        Statement stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery(sql:"SELECT TeamName FROM Team WHERE NumberofWins > 20");
        while (rs.next()) {
            String teamName = rs.getString(columnLabel:"TeamName");
            System.out.println("Team with more than 20 wins: " + teamName);
        }
        rs.close();
        stmt.close();
        c.close();
    } catch (Exception e) {
        System.err.println(e.getClass().getName() + ": " + e.getMessage());
        System.exit(status:0);
    }
}
```

Output Result:



The screenshot shows a Java application window titled "Lab". The console output is as follows:

```
Welcome to the NBA Statistics database:  
Please select from one of the following options (enter the corresponding number):  
1. Query points per game for a specific player on a specific date.  
2. Calculate the average assists per game for a specific player.  
3. What is the average age of NBA players in the current season.  
4. Get the field goal percentage for a specific player.  
5. Who is the top scorer in the league for the current season.  
6. Get the points per game for a specific player.  
7. Retrieve the names of players who are under 30 years old.  
8. What is the average points per game for a specific team in the current NBA season?  
9. Retrieve the list of teams that won more than 20 matches.  
10. Which team has the highest average rebounds per game in the current NBA season.  
Enter the Query Number: 9  
Golden State Warriors  
Milwaukee Bucks  
Phoenix Suns  
Philadelphia 76ers  
Utah Jazz
```

10. Which team has the highest average rebounds per game in the current NBA season

```
SELECT TeamName  
FROM Team  
WHERE TeamID = (  
    SELECT TeamID  
    FROM Player  
    GROUP BY TeamID  
    ORDER BY AVG(ReboundsPerGame) DESC  
    LIMIT 1);
```

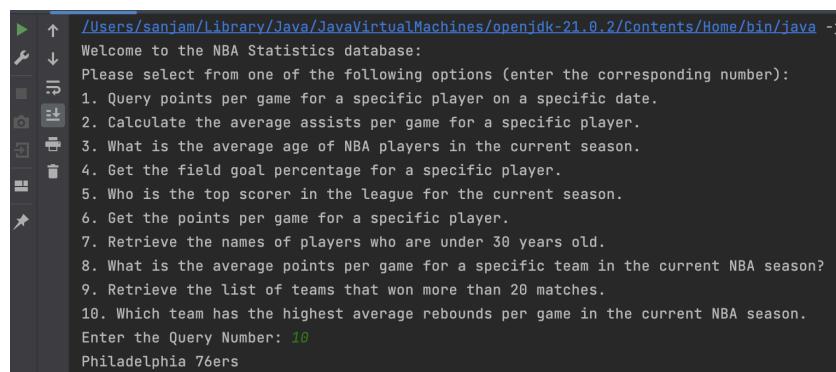
```
sqlite> SELECT TeamName
FROM Team
WHERE TeamID = (
    SELECT TeamID
    FROM Player
    GROUP BY TeamID
    ORDER BY AVG(ReboundsPerGame) DESC
    LIMIT 1
);
Philadelphia 76ers
```

Explanation - This query retrieves the team name that has the highest average rebounds per game in the current NBA season. It does this by grouping players by their team IDs, calculating the average rebounds per game for each team, sorting the results in descending order of average rebounds per game, and selecting the team name from the Team table corresponding to the team with the highest average rebounds per game.

Java Code:

```
// Method to execute Query #10: Which team has the highest average rebounds per game in the current NBA season?
public static void query10() {
    try {
        Class.forName(className:"org.sqlite.JDBC");
        Connection c = DriverManager.getConnection(url:"jdbc:sqlite:uclTourney.db");
        Statement stmt = c.createStatement();
        ResultSet rs = stmt.executeQuery("SELECT TeamName " +
            "FROM Team " +
            "WHERE TeamID = (SELECT TeamID FROM Player " +
            "GROUP BY TeamID " +
            "ORDER BY AVG(ReboundsPerGame) DESC " +
            "LIMIT 1)");
        while (rs.next()) {
            String teamName = rs.getString(columnLabel:"TeamName");
            System.out.println("Team with the highest average rebounds per game: " + teamName);
        }
        rs.close();
        stmt.close();
        c.close();
    } catch (Exception e) {
        System.err.println(e.getClass().getName() + ": " + e.getMessage());
        System.exit(status:0);
    }
}
```

Output Result:



```
/Users/sanjam/Library/Java/JavaVirtualMachines/openjdk-11.0.2/Contents/Home/bin/java -jar NBAStatistics.jar
Welcome to the NBA Statistics database:
Please select from one of the following options (enter the corresponding number):
1. Query points per game for a specific player on a specific date.
2. Calculate the average assists per game for a specific player.
3. What is the average age of NBA players in the current season.
4. Get the field goal percentage for a specific player.
5. Who is the top scorer in the league for the current season.
6. Get the points per game for a specific player.
7. Retrieve the names of players who are under 30 years old.
8. What is the average points per game for a specific team in the current NBA season?
9. Retrieve the list of teams that won more than 20 matches.
10. Which team has the highest average rebounds per game in the current NBA season.
Enter the Query Number: 10
Philadelphia 76ers
```

SQL Code .dump:

```
Last login: Thu Mar 14 00:00:29 on ttys000
[(base) sanjam@Harsanjams-MacBook-Air ~ % cd Downloads/sqlite-tools-osx-x64-3450000
(base) sanjam@Harsanjams-MacBook-Air sqlite-tools-osx-x64-3450000 % ls
[dssd.db          sqlite3          test.db
sqldiff          sqlite3_analyzer    uclTourney.db
(base) sanjam@Harsanjams-MacBook-Air sqlite-tools-osx-x64-3450000 % sqlite3
[SQLite version 3.39.3 2022-09-05 11:02:23
Enter ".help" for usage hints.
Connected to a transient in-memory database.
Use ".open FILENAME" to reopen on a persistent database.
sqlite> .open uclTourney.db
[sqlite> .dump
[PRAGMA foreign_keys=OFF;
BEGIN TRANSACTION;
```

Team:

```
CREATE TABLE Team (
    TeamID INT PRIMARY KEY,
    TeamName VARCHAR(255) NOT NULL,
    Sponsor VARCHAR(255),
    FoundedYear INT,
    WinLossRecord VARCHAR(20),
    NumberofWins INT,
    NumberofLosses INT,
    NumberofTies INT
, ManagerID INT);
INSERT INTO Team VALUES(1,'Los Angeles Lakers','Nike',1947,'20-15',20,15,0,1);
INSERT INTO Team VALUES(2,'Golden State Warriors','Under Armour',1946,'22-3',22,3,0,2);
INSERT INTO Team VALUES(3,'Brooklyn Nets','Adidas',1967,'18-17',18,17,0,3);
INSERT INTO Team VALUES(4,'Milwaukee Bucks','Puma',1968,'23-12',23,12,0,4);
INSERT INTO Team VALUES(5,'Phoenix Suns','State Farm',1968,'21-14',21,14,0,5);
INSERT INTO Team VALUES(6,'Philadelphia 76ers','StubHub',1946,'24-11',24,11,0,6);
INSERT INTO Team VALUES(7,'Utah Jazz','Qualtrics',1974,'25-10',25,10,0,7);
INSERT INTO Team VALUES(8,'Miami Heat','American Airlines',1988,'20-15',20,15,0,8);
INSERT INTO Team VALUES(9,'Minnesota Timberwolves','Fitbit',1989,'18-17',18,17,0,9);
INSERT INTO Team VALUES(10,'Chicago Bulls','United Airlines',1966,'19-16',19,16,0,10);
INSERT INTO Team VALUES(11,'San Antonio Spurs','Frost Bank',1967,'15-20',15,20,0,11);
INSERT INTO Team VALUES(12,'Orlando Magic','Disney',1989,'14-21',14,21,0,12);
INSERT INTO Team VALUES(13,'Detroit Pistons','Rocket Mortgage',1941,'10-19',10,19,0,13);
INSERT INTO Team VALUES(14,'Washington Wizards','Capital One',1961,'11-18',11,18,0,14);
```

Game:

```
CREATE TABLE Game (
    GameID INT PRIMARY KEY,
    Date DATE,
    Venue VARCHAR(255),
    Attendance INT,
    FinalScores VARCHAR(255), -- assuming scores are stored as a string
    Awards VARCHAR(255) -- assuming awards can be multiple and stored as a string
);
```

```
INSERT INTO Game VALUES(1, '2024-03-10', 'Staples Center', 18000, NULL, '1');
INSERT INTO Game VALUES(2, '2024-03-12', 'Chase Center', 15000, NULL, '2');
INSERT INTO Game VALUES(3, '2024-03-15', 'Barclays Center', 17000, NULL, '3');
INSERT INTO Game VALUES(4, '2024-03-17', 'Fiserv Forum', 16000, NULL, '4');
INSERT INTO Game VALUES(5, '2024-03-20', 'Talking Stick Resort Arena', 18000, NULL, '5');
INSERT INTO Game VALUES(6, '2024-03-22', 'Wells Fargo Center', 16000, NULL, '6');
INSERT INTO Game VALUES(7, '2024-03-25', 'Vivint Arena', 20000, NULL, '7');
INSERT INTO Game VALUES(8, '2024-03-27', 'American Airlines Arena', 19000, NULL, '8');
INSERT INTO Game VALUES(9, '2024-03-30', 'Target Center', 18000, NULL, '9');
INSERT INTO Game VALUES(10, '2024-04-01', 'United Center', 17000, NULL, '10');
INSERT INTO Game VALUES(11, '2024-04-03', 'AT&T Center', 16000, NULL, '11');
INSERT INTO Game VALUES(12, '2024-04-05', 'Amway Center', 15000, NULL, '12');
INSERT INTO Game VALUES(13, '2024-04-08', 'Little Caesars Arena', 16000, NULL, '13');
INSERT INTO Game VALUES(14, '2024-04-10', 'Capital One Arena', 15000, NULL, '14');
```

State/Conference:

```
CREATE TABLE State (
    StateID INT PRIMARY KEY,
    Name VARCHAR(255) NOT NULL,
    Conference VARCHAR(50) NOT NULL
);
INSERT INTO State VALUES(1, 'California', 'West');
INSERT INTO State VALUES(2, 'New York', 'East');
INSERT INTO State VALUES(3, 'Texas', 'West');
INSERT INTO State VALUES(4, 'Wisconsin', 'East');
INSERT INTO State VALUES(5, 'Arizona', 'West');
INSERT INTO State VALUES(6, 'Pennsylvania', 'East');
INSERT INTO State VALUES(7, 'Utah', 'West');
INSERT INTO State VALUES(8, 'Florida', 'East');
INSERT INTO State VALUES(9, 'Minnesota', 'West');
INSERT INTO State VALUES(10, 'Illinois', 'East');
INSERT INTO State VALUES(11, 'Michigan', 'East');
INSERT INTO State VALUES(12, 'Washington', 'East');
INSERT INTO State VALUES(13, 'Oregon', 'West');
INSERT INTO State VALUES(14, 'Colorado', 'West');
```

Manager:

```
CREATE TABLE Manager (
    ManagerID INT PRIMARY KEY,
    Name VARCHAR(255) NOT NULL,
    TeamID INT,
    FOREIGN KEY (TeamID) REFERENCES Team(TeamID)
);
INSERT INTO Manager VALUES(1, 'Frank Vogel', NULL);
INSERT INTO Manager VALUES(2, 'Steve Kerr', NULL);
INSERT INTO Manager VALUES(3, 'Steve Nash', NULL);
INSERT INTO Manager VALUES(4, 'Mike Budenholzer', NULL);
INSERT INTO Manager VALUES(5, 'Monty Williams', NULL);
INSERT INTO Manager VALUES(6, 'Doc Rivers', NULL);
INSERT INTO Manager VALUES(7, 'Quin Snyder', NULL);
INSERT INTO Manager VALUES(8, 'Erik Spoelstra', NULL);
INSERT INTO Manager VALUES(9, 'Chris Finch', NULL);
INSERT INTO Manager VALUES(10, 'Billy Donovan', NULL);
INSERT INTO Manager VALUES(11, 'Gregg Popovich', NULL);
INSERT INTO Manager VALUES(12, 'Jamahl Mosley', NULL);
INSERT INTO Manager VALUES(13, 'Dwane Casey', NULL);
INSERT INTO Manager VALUES(14, 'Wes Unseld Jr.', NULL);
```

```
CREATE TABLE Player_Game (
    PlayerID INT,
    GameID INT,
    PRIMARY KEY (PlayerID, GameID),
    FOREIGN KEY (PlayerID) REFERENCES Player(PlayerID),
    FOREIGN KEY (GameID) REFERENCES Game(GameID)
);
INSERT INTO Player_Game VALUES(1,1);
INSERT INTO Player_Game VALUES(2,1);
INSERT INTO Player_Game VALUES(3,2);
INSERT INTO Player_Game VALUES(4,3);
INSERT INTO Player_Game VALUES(5,3);
INSERT INTO Player_Game VALUES(6,4);
INSERT INTO Player_Game VALUES(7,5);
INSERT INTO Player_Game VALUES(8,5);
INSERT INTO Player_Game VALUES(9,6);
INSERT INTO Player_Game VALUES(10,7);
INSERT INTO Player_Game VALUES(11,7);
INSERT INTO Player_Game VALUES(12,8);
INSERT INTO Player_Game VALUES(13,9);
INSERT INTO Player_Game VALUES(14,9);
INSERT INTO Player_Game VALUES(15,10);
INSERT INTO Player_Game VALUES(16,11);
INSERT INTO Player_Game VALUES(17,11);
INSERT INTO Player_Game VALUES(18,12);
INSERT INTO Player_Game VALUES(19,13);
INSERT INTO Player_Game VALUES(20,13);
INSERT INTO Player_Game VALUES(21,14);

CREATE TABLE Game_Team (
    GameID INT,
    TeamID INT,
    PRIMARY KEY (GameID, TeamID),
    FOREIGN KEY (GameID) REFERENCES Game(GameID),
    FOREIGN KEY (TeamID) REFERENCES Team(TeamID)
);
INSERT INTO Game_Team VALUES(1,1);
INSERT INTO Game_Team VALUES(1,2);
INSERT INTO Game_Team VALUES(2,1);
INSERT INTO Game_Team VALUES(2,2);
INSERT INTO Game_Team VALUES(3,3);
INSERT INTO Game_Team VALUES(3,4);
INSERT INTO Game_Team VALUES(4,3);
INSERT INTO Game_Team VALUES(4,4);
INSERT INTO Game_Team VALUES(5,5);
INSERT INTO Game_Team VALUES(5,6);
INSERT INTO Game_Team VALUES(6,5);
INSERT INTO Game_Team VALUES(6,6);
INSERT INTO Game_Team VALUES(7,7);
INSERT INTO Game_Team VALUES(7,8);
INSERT INTO Game_Team VALUES(8,7);
INSERT INTO Game_Team VALUES(8,8);
INSERT INTO Game_Team VALUES(9,9);
INSERT INTO Game_Team VALUES(9,10);
INSERT INTO Game_Team VALUES(10,9);
INSERT INTO Game_Team VALUES(10,10);
INSERT INTO Game_Team VALUES(11,11);
INSERT INTO Game_Team VALUES(11,12);
INSERT INTO Game_Team VALUES(12,11);
INSERT INTO Game_Team VALUES(12,12);
INSERT INTO Game_Team VALUES(13,13);
INSERT INTO Game_Team VALUES(13,14);
INSERT INTO Game_Team VALUES(14,13);
INSERT INTO Game_Team VALUES(14,14);
```

Player: