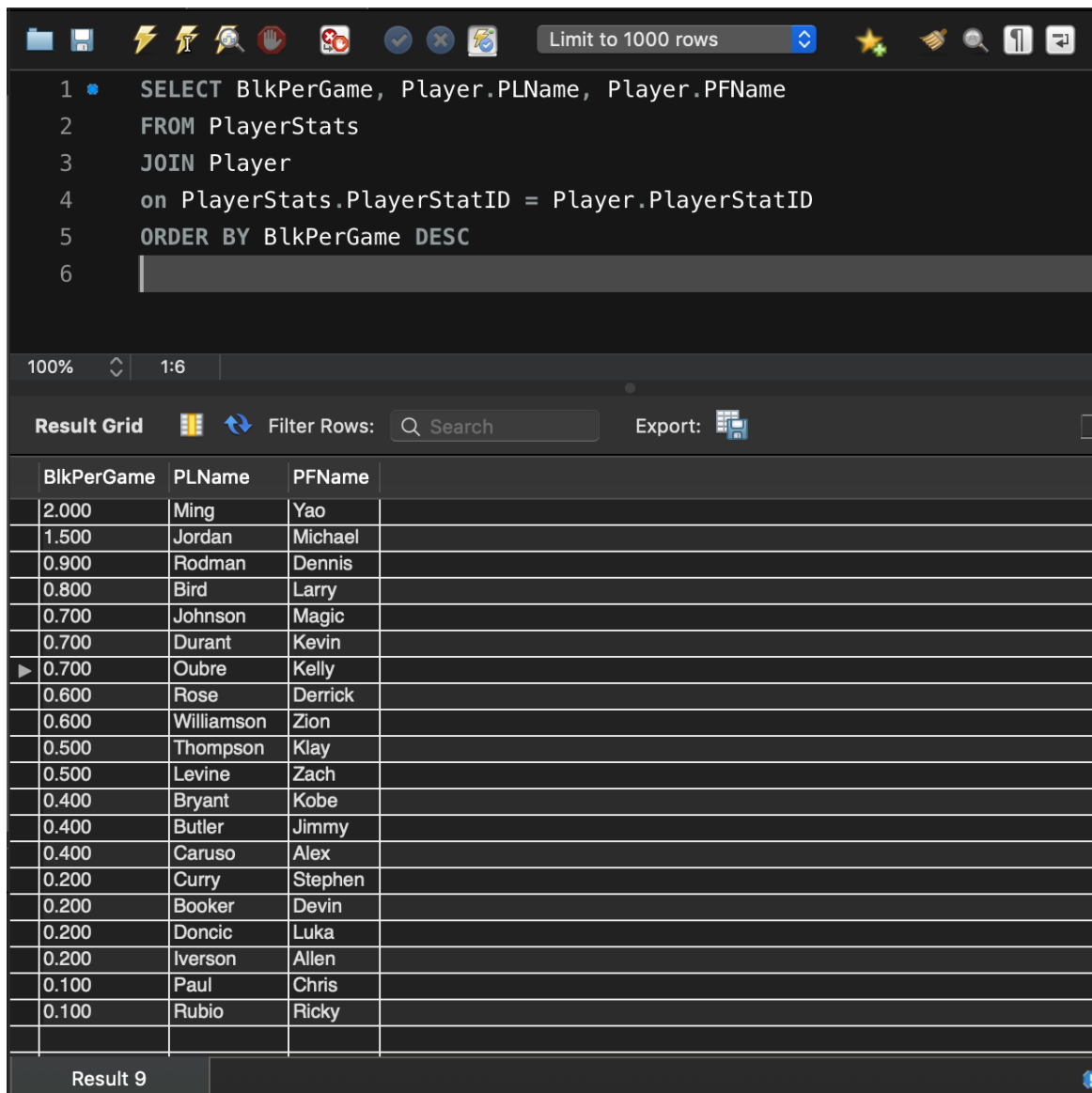


Database Project Assignment 6: Queries and Results

Name: Josh Valentino

1. From the players we have data on, who has the highest blocks per game? Please list all the players in order by their blocks. (Complex 1)

```
SELECT BlkPerGame, Player.PLName, Player.PFName
FROM PlayerStats
JOIN Player
on PlayerStats.PlayerStatID = Player.PlayerStatID
ORDER BY BlkPerGame DESC
```



Limit to 1000 rows

```
1 SELECT BlkPerGame, Player.PLName, Player.PFName
2 FROM PlayerStats
3 JOIN Player
4 on PlayerStats.PlayerStatID = Player.PlayerStatID
5 ORDER BY BlkPerGame DESC
6
```

100% 1:6

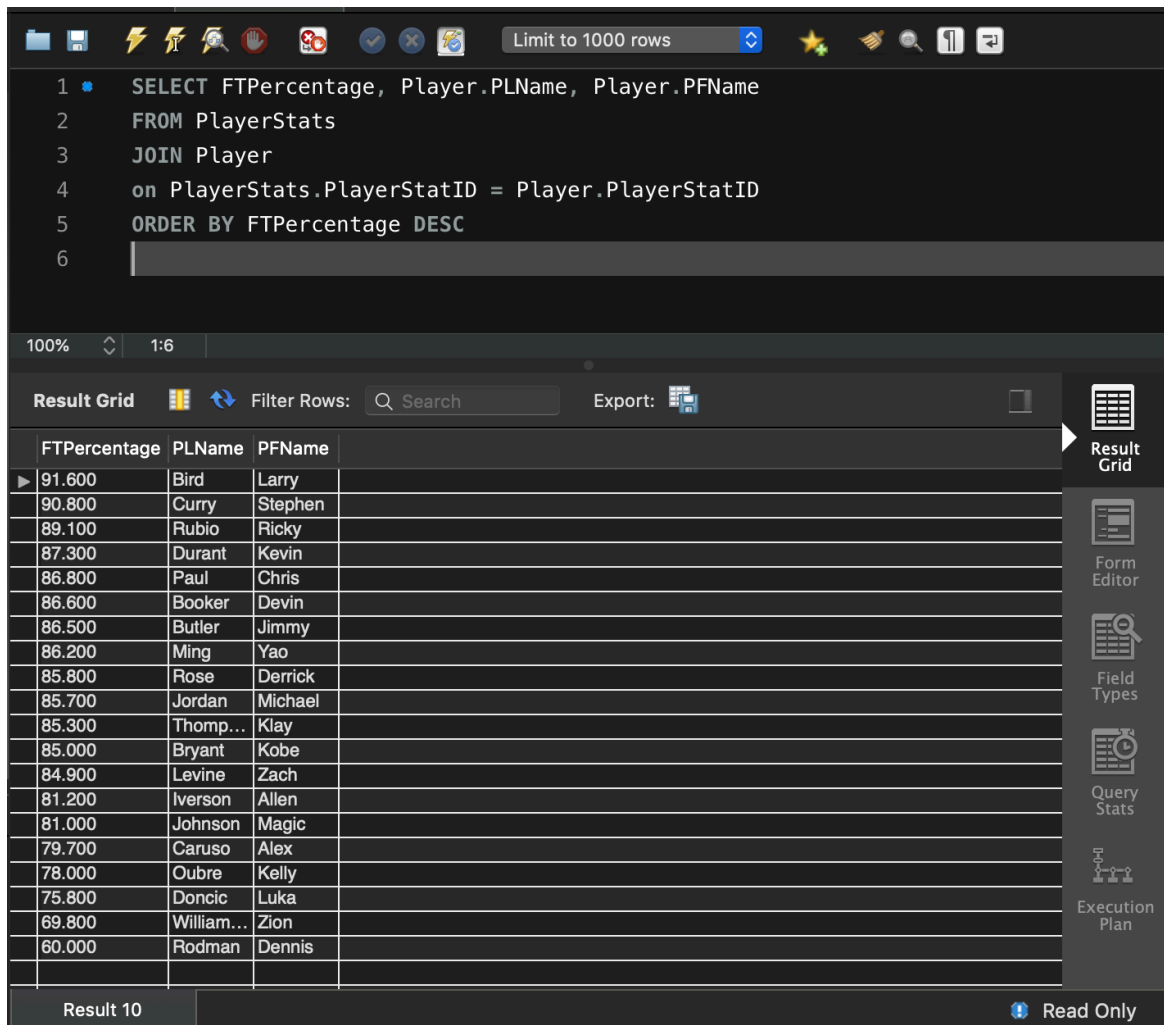
Result Grid Filter Rows: Search Export:

	BlkPerGame	PLName	PFName
	2.000	Ming	Yao
	1.500	Jordan	Michael
	0.900	Rodman	Dennis
	0.800	Bird	Larry
	0.700	Johnson	Magic
	0.700	Durant	Kevin
▶	0.700	Oubre	Kelly
	0.600	Rose	Derrick
	0.600	Williamson	Zion
	0.500	Thompson	Klay
	0.500	Levine	Zach
	0.400	Bryant	Kobe
	0.400	Butler	Jimmy
	0.400	Caruso	Alex
	0.200	Curry	Stephen
	0.200	Booker	Devin
	0.200	Doncic	Luka
	0.200	Iverson	Allen
	0.100	Paul	Chris
	0.100	Rubio	Ricky

Result 9

2. From the players we have data on, who has the best free throw percentage? Please list all the players in descending order by percentage. (Complex 2)

```
SELECT FTPercentage, Player.PLName, Player.PFName
FROM PlayerStats
JOIN Player
on PlayerStats.PlayerStatID = Player.PlayerStatID
ORDER BY FTPercentage DESC
```



The screenshot shows a database query tool interface. At the top, there's a toolbar with various icons and a 'Limit to 1000 rows' dropdown. Below the toolbar, the SQL query is entered in a text area. The query is as follows:

```
1 SELECT FTPercentage, Player.PLName, Player.PFName
2 FROM PlayerStats
3 JOIN Player
4 on PlayerStats.PlayerStatID = Player.PlayerStatID
5 ORDER BY FTPercentage DESC
6
```

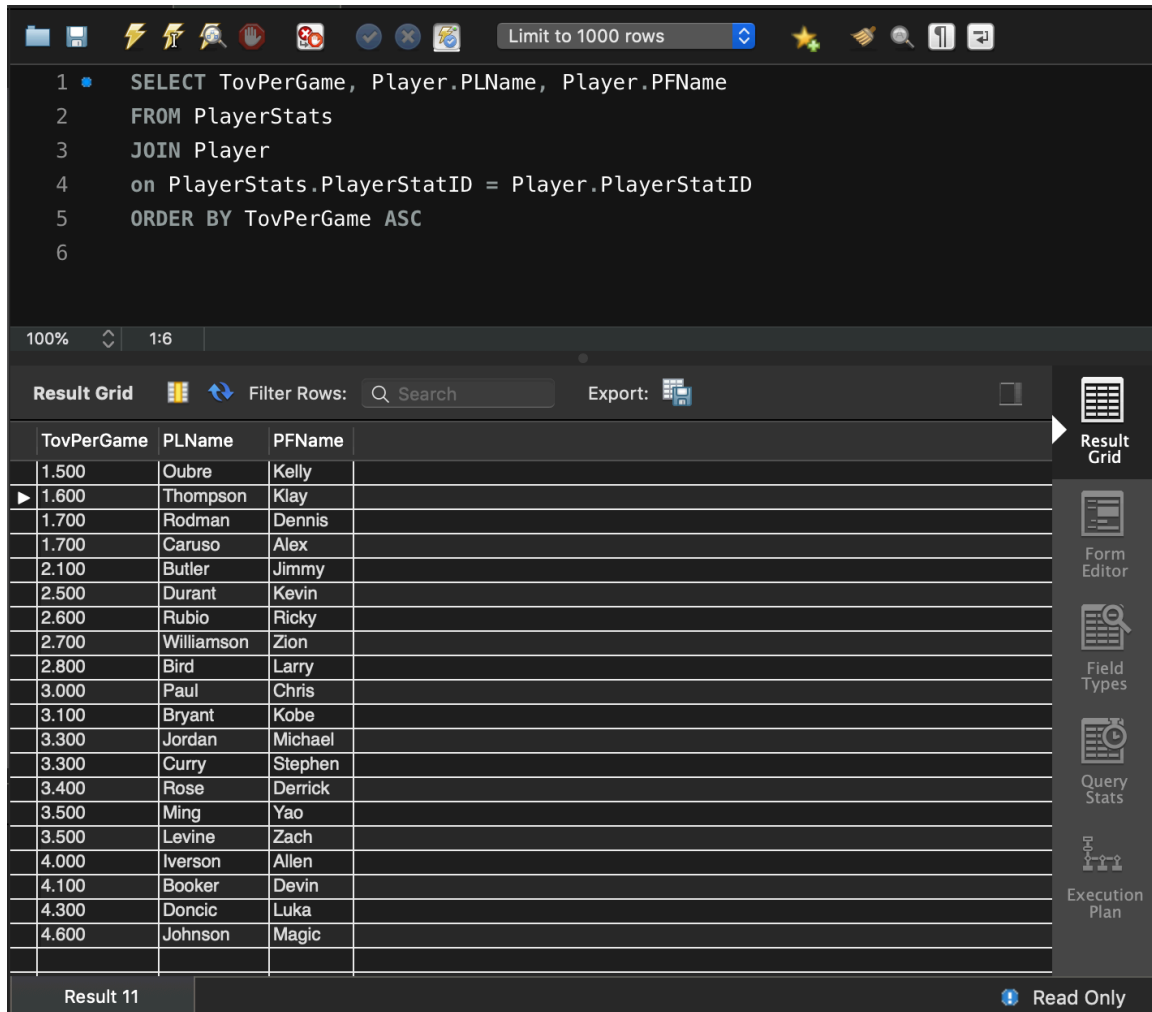
Below the query editor, there's a 'Result Grid' section. It includes a 'Filter Rows' search bar and an 'Export' button. The results are displayed in a table with the following columns: FTPercentage, PLName, and PFName. The table contains 20 rows of data, sorted by FTPercentage in descending order. The first row shows Larry Bird with a percentage of 91.600. The last row shows Dennis Rodman with a percentage of 60.000.

FTPercentage	PLName	PFName
91.600	Bird	Larry
90.800	Curry	Stephen
89.100	Rubio	Ricky
87.300	Durant	Kevin
86.800	Paul	Chris
86.600	Booker	Devin
86.500	Butler	Jimmy
86.200	Ming	Yao
85.800	Rose	Derrick
85.700	Jordan	Michael
85.300	Thomp...	Klay
85.000	Bryant	Kobe
84.900	Levine	Zach
81.200	Iverson	Allen
81.000	Johnson	Magic
79.700	Caruso	Alex
78.000	Oubre	Kelly
75.800	Doncic	Luka
69.800	William...	Zion
60.000	Rodman	Dennis

At the bottom of the interface, there's a 'Result 10' label and a 'Read Only' status indicator.

3. From the players we have data on, who has the lowest turnovers per game? Please list all the players in ascending order by turnovers. (Complex 3)

```
SELECT TovPerGame, Player.PLName, Player.PFName
FROM PlayerStats
JOIN Player
on PlayerStats.PlayerStatID = Player.PlayerStatID
ORDER BY TovPerGame ASC
```



Limit to 1000 rows

```
1 SELECT TovPerGame, Player.PLName, Player.PFName
2 FROM PlayerStats
3 JOIN Player
4 on PlayerStats.PlayerStatID = Player.PlayerStatID
5 ORDER BY TovPerGame ASC
6
```

100% 1:6

Result Grid Filter Rows: Search Export:

TovPerGame	PLName	PFName
1.500	Oubre	Kelly
1.600	Thompson	Klay
1.700	Rodman	Dennis
1.700	Caruso	Alex
2.100	Butler	Jimmy
2.500	Durant	Kevin
2.600	Rubio	Ricky
2.700	Williamson	Zion
2.800	Bird	Larry
3.000	Paul	Chris
3.100	Bryant	Kobe
3.300	Jordan	Michael
3.300	Curry	Stephen
3.400	Rose	Derrick
3.500	Ming	Yao
3.500	Levine	Zach
4.000	Iverson	Allen
4.100	Booker	Devin
4.300	Doncic	Luka
4.600	Johnson	Magic

Result 11 Read Only

4. From the players we have data on, how many of them are shooting guards?

```
SELECT COUNT(*) as "Number of SGs"
FROM Player
WHERE PositionID = 2
```

[illegible]

5. From the players we have data on, how many of them are point guards? Please list all the players names that are point guards. (Complex 4)

```
SELECT PositionName, Player.PLName, Player.PFName,
(SELECT COUNT(*) FROM Player WHERE PositionID = 1) as "Number of PGs"
FROM Positions
JOIN Player
on Positions.PositionID = Player.PositionID
WHERE Positions.PositionID = 1;
```

The screenshot shows the SQL Studio interface. At the top, there are tabs for "Query 1" and "SQL File 1*". Below the toolbar, a SQL query is displayed:

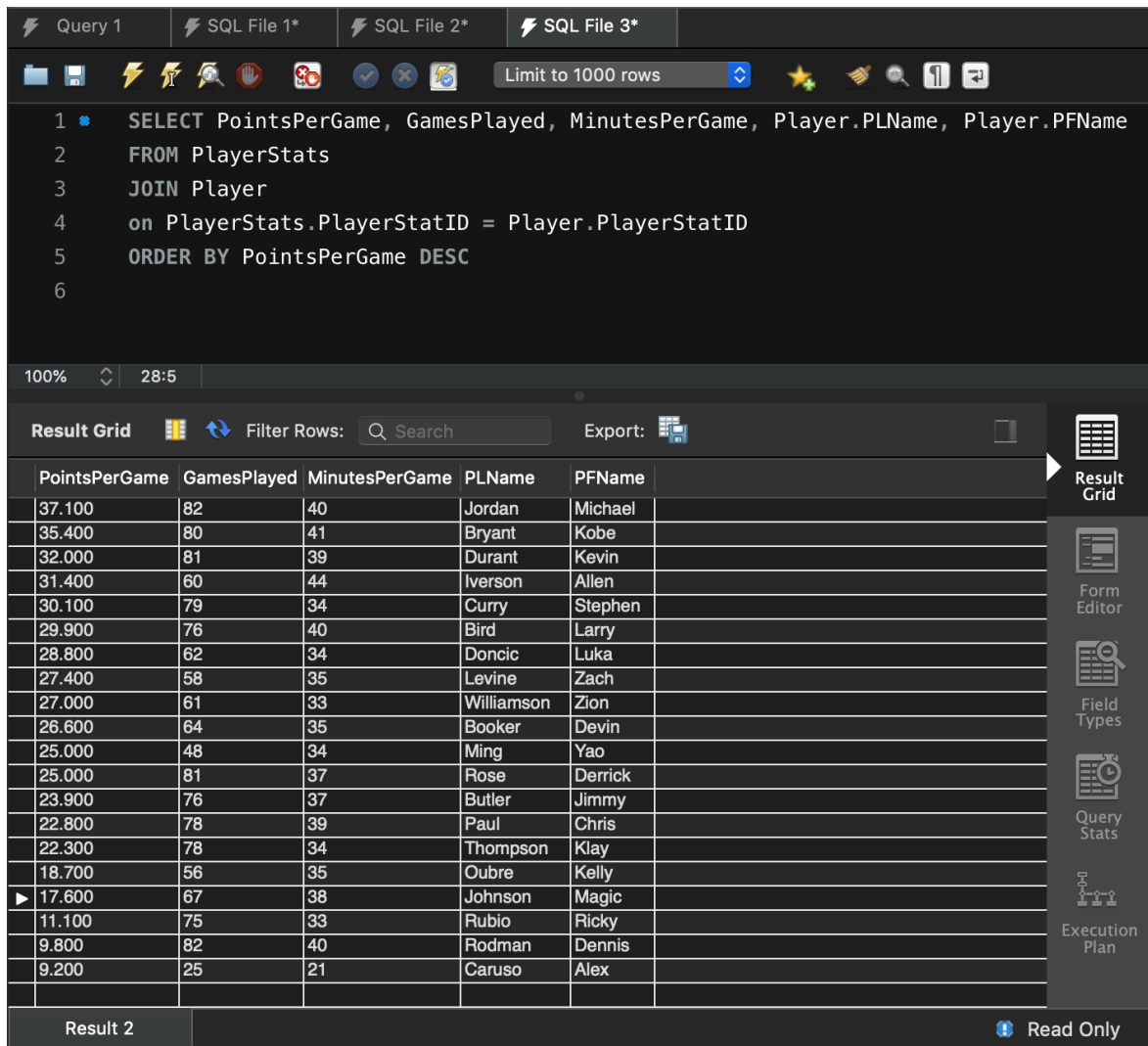
```
SELECT PositionName, Player.PLName, Player.PFName,  
(SELECT COUNT(*) FROM Player WHERE PositionID = 1) as "Number of PGs"  
FROM Positions  
JOIN Player  
on Positions.PositionID = Player.PositionID  
WHERE Positions.PositionID = 1;
```

The query results are shown below the editor. The first row indicates "100%" zoom level and "1:7" rows visible.

PositionName	PLName	PFName	Number of PGs
Point Guard	Johnson	Magic	7
Point Guard	Curry	Stephen	7
Point Guard	Rose	Derrick	7
Point Guard	Paul	Chris	7
Point Guard	Rubio	Ricky	7
Point Guard	Caruso	Alex	7
Point Guard	Iverson	Allan	7

6. From the players we have data on, who has the most points per game? Please list all the players in descending order by points per game. Also list they're games played, and minutes per game to give a better picture of their scoring that season. (Complex 5)

```
SELECT PointsPerGame, GamesPlayed, MinutesPerGame, Player.PLName,
Player.PFName
FROM PlayerStats
JOIN Player
on PlayerStats.PlayerStatID = Player.PlayerStatID
ORDER BY PointsPerGame DESC
```



The screenshot shows a SQL IDE interface. At the top, there are tabs for 'Query 1', 'SQL File 1*', 'SQL File 2*', and 'SQL File 3*'. Below the tabs is a toolbar with various icons, including a 'Limit to 1000 rows' dropdown. The main area displays a SQL query with line numbers 1 through 6. Below the query, the 'Result Grid' is visible, showing a table with 6 columns: PointsPerGame, GamesPlayed, MinutesPerGame, PLName, PFName, and an empty column. The table contains 20 rows of data, sorted by PointsPerGame in descending order. On the right side of the interface, there is a vertical toolbar with icons for 'Result Grid', 'Form Editor', 'Field Types', 'Query Stats', and 'Execution Plan'. At the bottom, there is a 'Result 2' tab and a 'Read Only' status indicator.

PointsPerGame	GamesPlayed	MinutesPerGame	PLName	PFName	
37.100	82	40	Jordan	Michael	
35.400	80	41	Bryant	Kobe	
32.000	81	39	Durant	Kevin	
31.400	60	44	Iverson	Allen	
30.100	79	34	Curry	Stephen	
29.900	76	40	Bird	Larry	
28.800	62	34	Doncic	Luka	
27.400	58	35	Levine	Zach	
27.000	61	33	Williamson	Zion	
26.600	64	35	Booker	Devin	
25.000	48	34	Ming	Yao	
25.000	81	37	Rose	Derrick	
23.900	76	37	Butler	Jimmy	
22.800	78	39	Paul	Chris	
22.300	78	34	Thompson	Klay	
18.700	56	35	Oubre	Kelly	
17.600	67	38	Johnson	Magic	
11.100	75	33	Rubio	Ricky	
9.800	82	40	Rodman	Dennis	
9.200	25	21	Caruso	Alex	

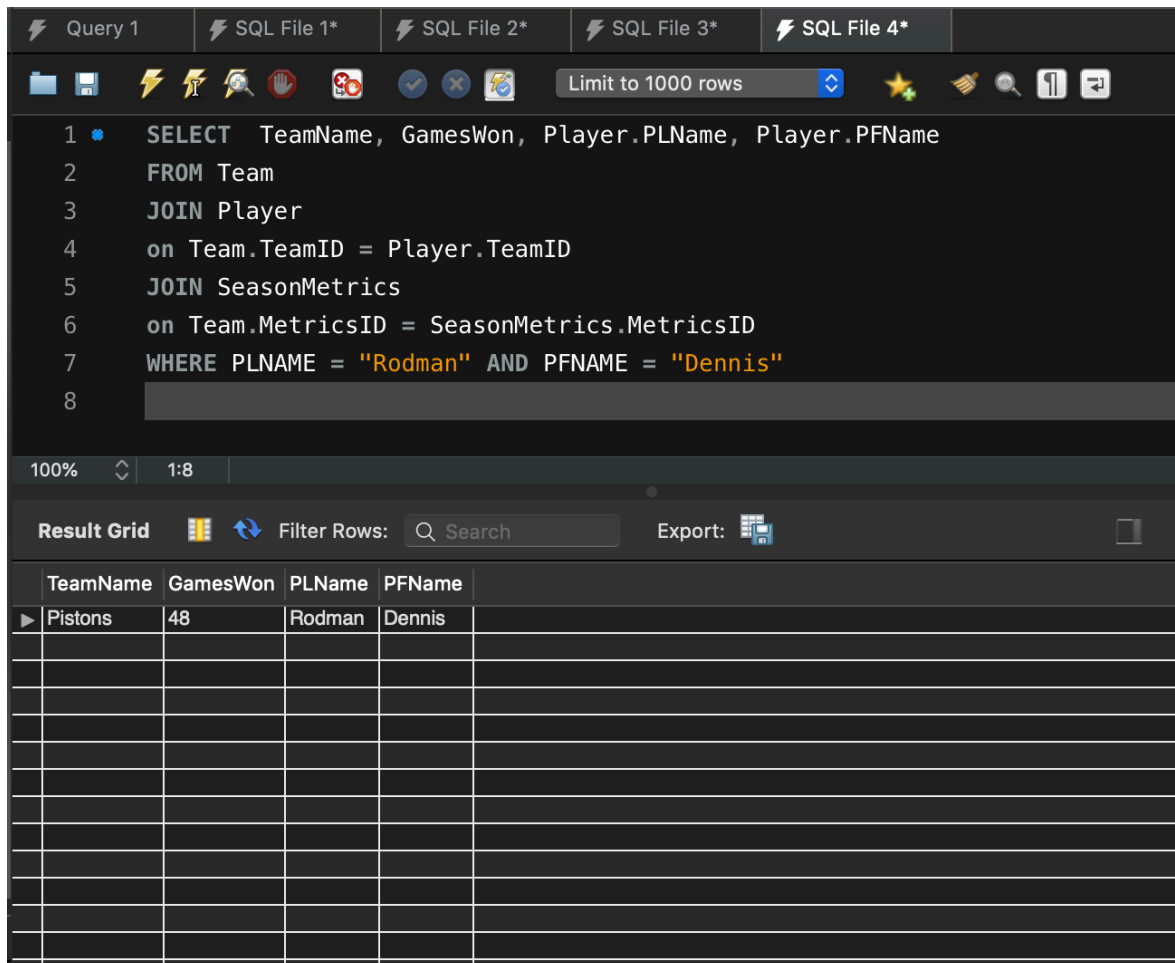
7. After the previous query, we see that hall of famer Dennis Rodman only had 9.8 points per game that season. Could we see his rebounds per game? (Complex 6)

```
SELECT TotalReboundsPerGame, Player.PLName, Player.PFName
FROM PlayerStats
JOIN Player
on PlayerStats.PlayerStatID = Player.PlayerStatID
WHERE PLNAME = "Rodman" AND PFNAME = "Dennis"
```

[illegible]

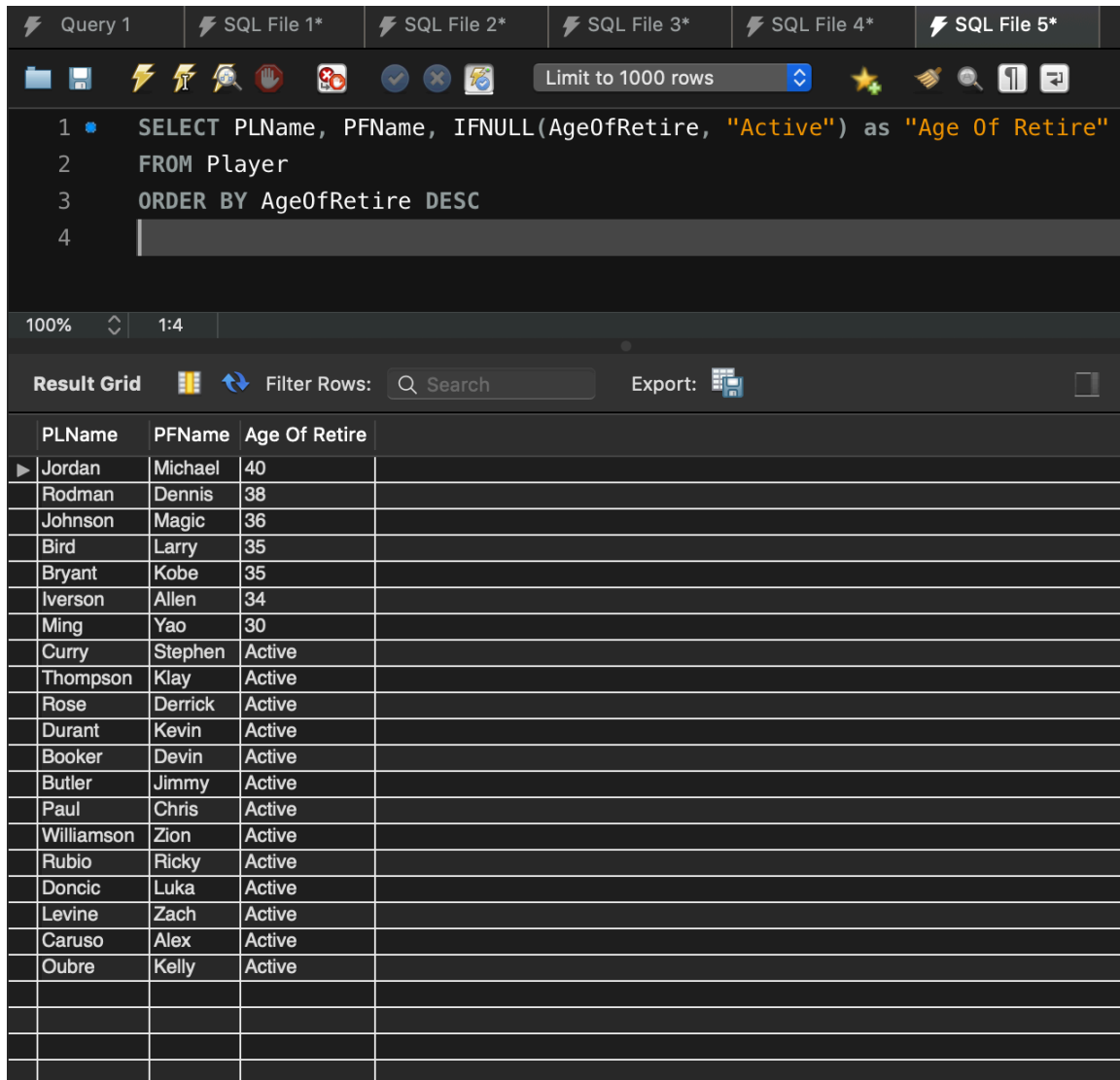
8. Looking at Dennis Rodman stats is very interesting. Could we see what team he played for that year and how many wins they had? (Complex 7)

```
SELECT TeamName, GamesWon, Player.PLName, Player.PFName
FROM Team
JOIN Player
on Team.TeamID = Player.TeamID
JOIN SeasonMetrics
on Team.MetricsID = SeasonMetrics.MetricsID
WHERE PLNAME = "Rodman" AND PFNAME = "Dennis"
```



9. What age did the players we have data on retire? (Complex 8)

```
SELECT PLName, PFName, IFNULL(AgeOfRetire, "Active") as "Age Of Retire"  
FROM Player  
ORDER BY AgeOfRetire DESC
```

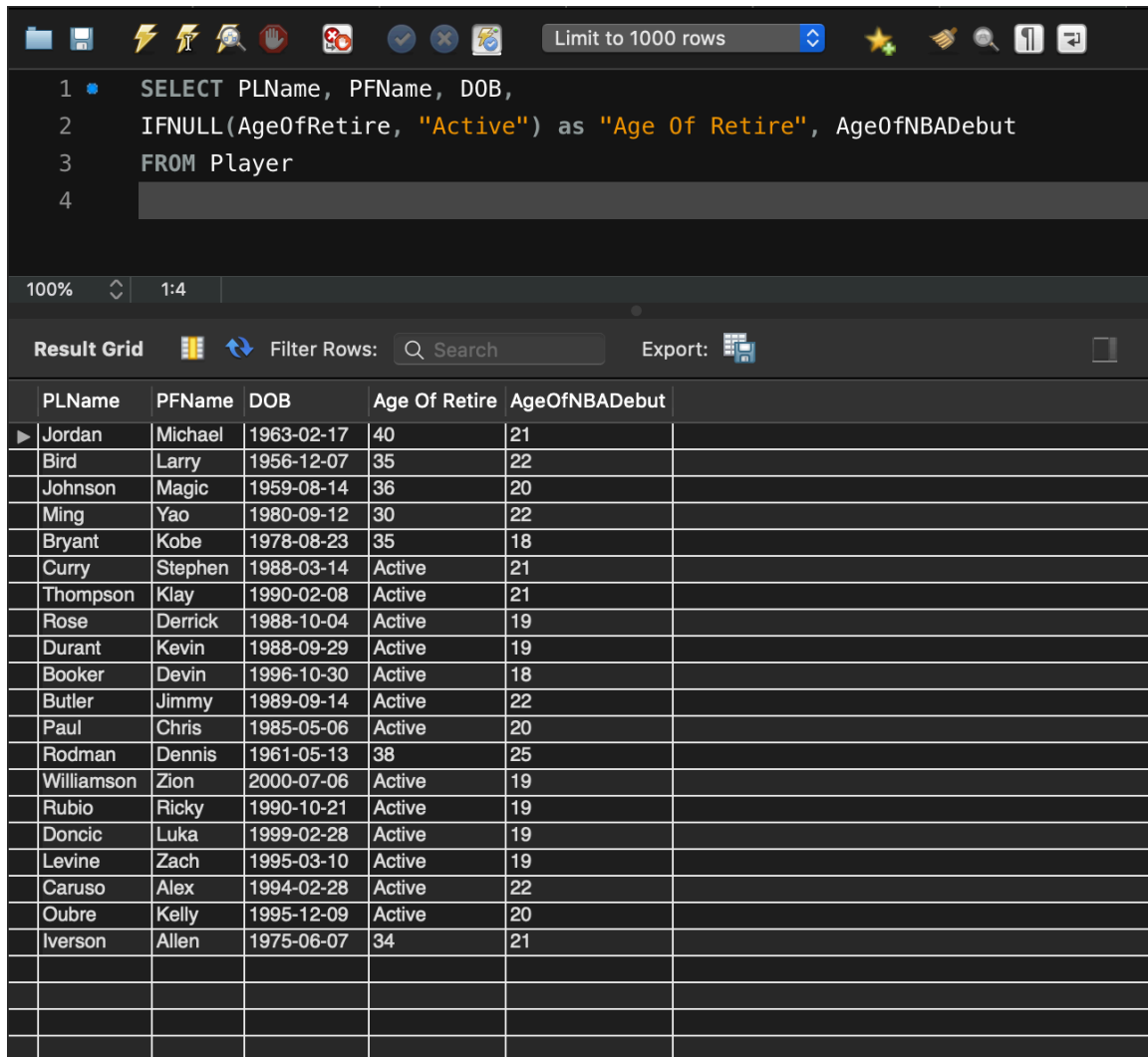


The screenshot shows a SQL IDE interface. At the top, there are tabs for 'Query 1' and 'SQL File 1*' through 'SQL File 5*'. Below the tabs is a toolbar with various icons, including a 'Limit to 1000 rows' dropdown. The main area displays the SQL query from the previous block. Below the query editor, there is a 'Result Grid' section with a search bar and an 'Export' button. The result grid contains a table with the following data:

PLName	PFName	Age Of Retire
Jordan	Michael	40
Rodman	Dennis	38
Johnson	Magic	36
Bird	Larry	35
Bryant	Kobe	35
Iverson	Allen	34
Ming	Yao	30
Curry	Stephen	Active
Thompson	Klay	Active
Rose	Derrick	Active
Durant	Kevin	Active
Booker	Devin	Active
Butler	Jimmy	Active
Paul	Chris	Active
Williamson	Zion	Active
Rubio	Ricky	Active
Doncic	Luka	Active
Levine	Zach	Active
Caruso	Alex	Active
Oubre	Kelly	Active

10. Show us all the data we have on birthdays, age of retire, and age of debut.

```
SELECT PLName, PFName, DOB,  
IFNULL(AgeOfRetire, "Active") as "Age Of Retire", AgeOfNBADebut  
FROM Player
```

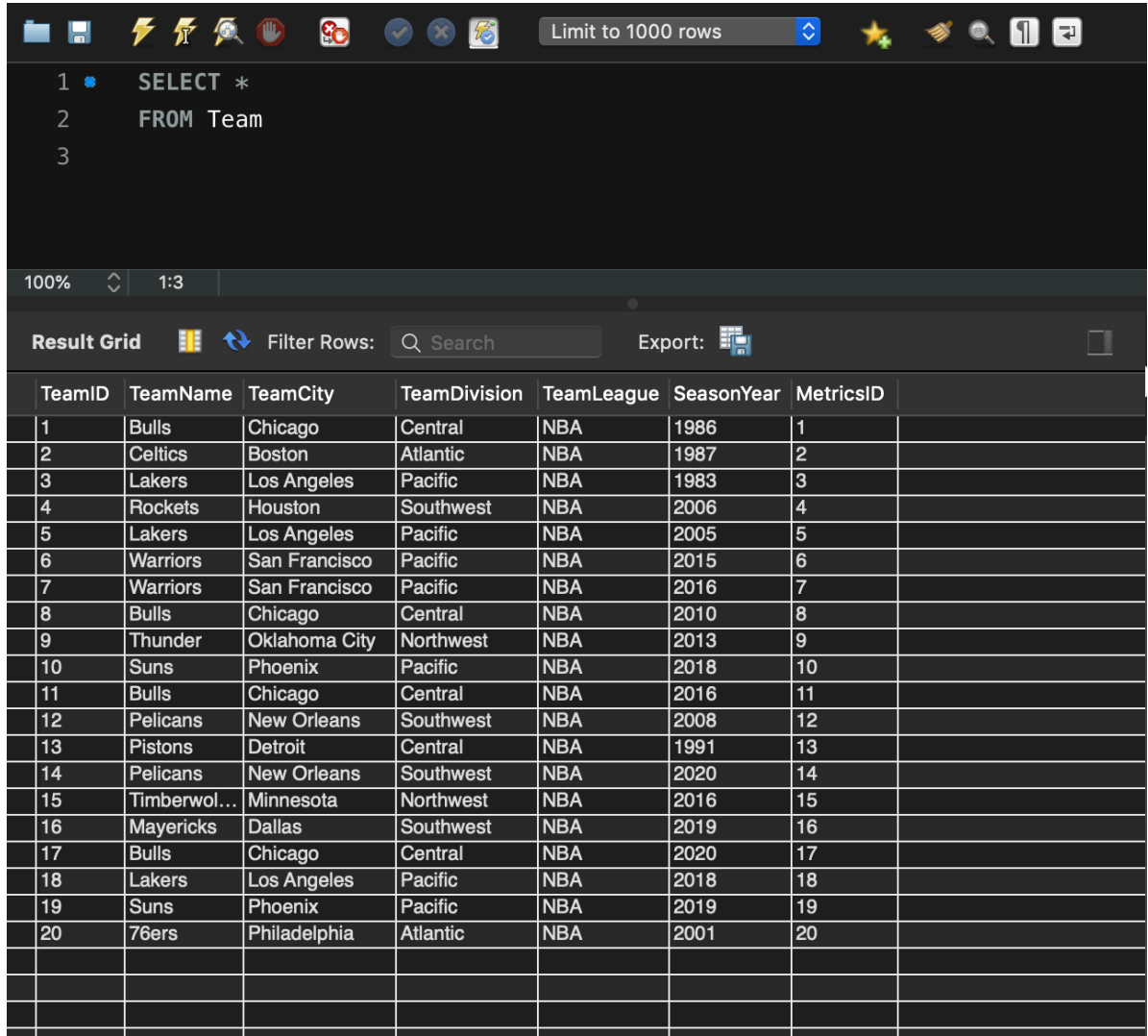


The screenshot shows a database query interface. At the top, there's a toolbar with various icons and a "Limit to 1000 rows" dropdown. Below the toolbar, the SQL query is entered in a text area. The query is:
1 SELECT PLName, PFName, DOB,
2 IFNULL(AgeOfRetire, "Active") as "Age Of Retire", AgeOfNBADebut
3 FROM Player
4
Below the query, there's a "Result Grid" section. It includes a "Filter Rows:" search bar and an "Export:" button. The results are displayed in a table with the following columns: PLName, PFName, DOB, Age Of Retire, and AgeOfNBADebut. The table contains 20 rows of data, including players like Jordan, Bird, Johnson, Ming, Bryant, Curry, Thompson, Rose, Durant, Booker, Butler, Paul, Rodman, Williamson, Rubio, Doncic, Levine, Caruso, Oubre, and Iverson.

PLName	PFName	DOB	Age Of Retire	AgeOfNBADebut
Jordan	Michael	1963-02-17	40	21
Bird	Larry	1956-12-07	35	22
Johnson	Magic	1959-08-14	36	20
Ming	Yao	1980-09-12	30	22
Bryant	Kobe	1978-08-23	35	18
Curry	Stephen	1988-03-14	Active	21
Thompson	Klay	1990-02-08	Active	21
Rose	Derrick	1988-10-04	Active	19
Durant	Kevin	1988-09-29	Active	19
Booker	Devin	1996-10-30	Active	18
Butler	Jimmy	1989-09-14	Active	22
Paul	Chris	1985-05-06	Active	20
Rodman	Dennis	1961-05-13	38	25
Williamson	Zion	2000-07-06	Active	19
Rubio	Ricky	1990-10-21	Active	19
Doncic	Luka	1999-02-28	Active	19
Levine	Zach	1995-03-10	Active	19
Caruso	Alex	1994-02-28	Active	22
Oubre	Kelly	1995-12-09	Active	20
Iverson	Allen	1975-06-07	34	21

11. Display all the team data we have.

```
SELECT *  
FROM Team
```



The screenshot shows a database application window. At the top, there's a toolbar with various icons and a 'Limit to 1000 rows' dropdown. Below the toolbar, the SQL query is entered in a text area:

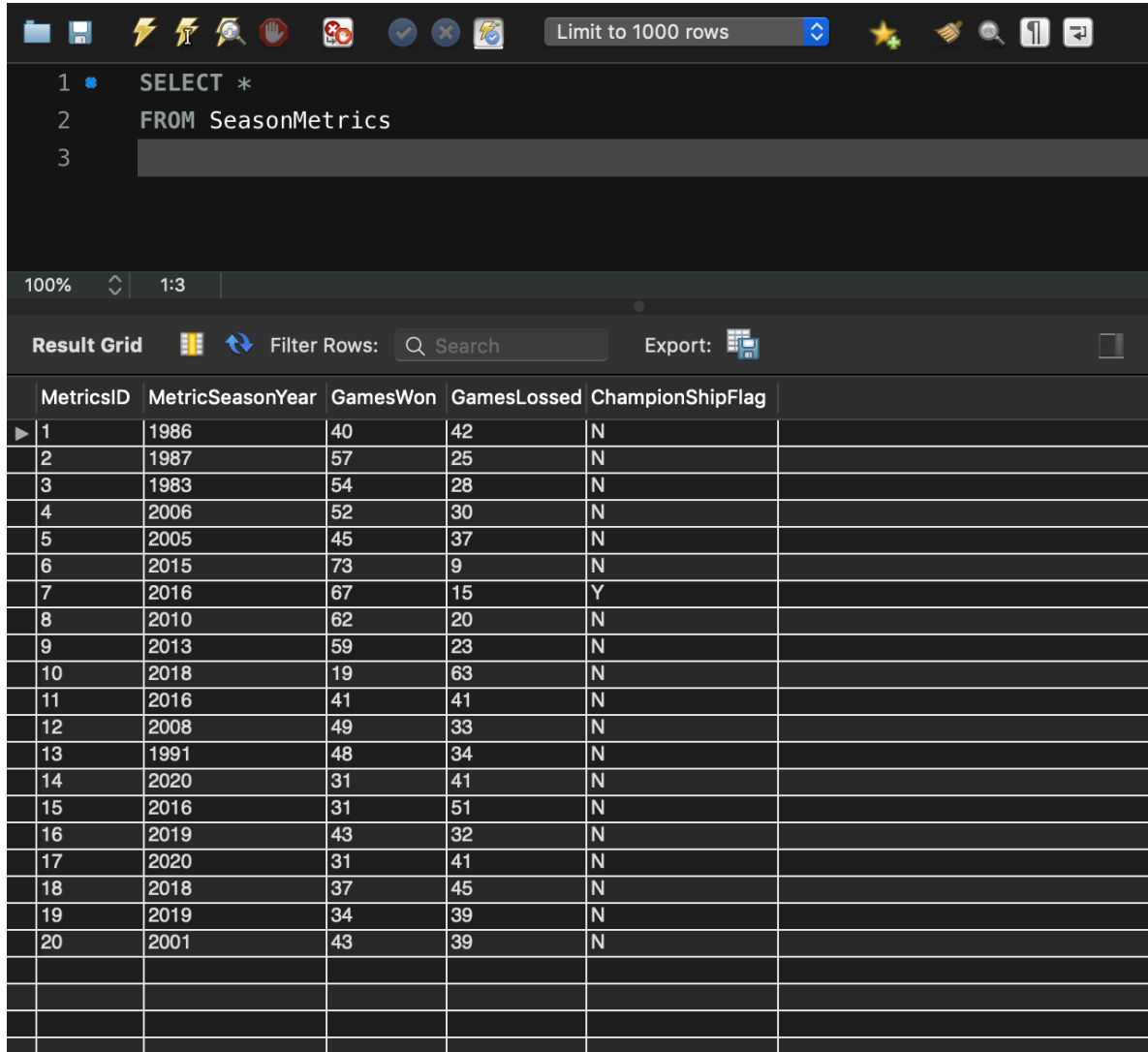
```
1 SELECT *  
2 FROM Team  
3
```

Below the query editor, there's a 'Result Grid' section. It includes a 'Filter Rows' search bar and an 'Export' button. The results are displayed in a table with the following columns: TeamID, TeamName, TeamCity, TeamDivision, TeamLeague, SeasonYear, MetricsID, and an empty column. The table contains 20 rows of data.

TeamID	TeamName	TeamCity	TeamDivision	TeamLeague	SeasonYear	MetricsID	
1	Bulls	Chicago	Central	NBA	1986	1	
2	Celtics	Boston	Atlantic	NBA	1987	2	
3	Lakers	Los Angeles	Pacific	NBA	1983	3	
4	Rockets	Houston	Southwest	NBA	2006	4	
5	Lakers	Los Angeles	Pacific	NBA	2005	5	
6	Warriors	San Francisco	Pacific	NBA	2015	6	
7	Warriors	San Francisco	Pacific	NBA	2016	7	
8	Bulls	Chicago	Central	NBA	2010	8	
9	Thunder	Oklahoma City	Northwest	NBA	2013	9	
10	Suns	Phoenix	Pacific	NBA	2018	10	
11	Bulls	Chicago	Central	NBA	2016	11	
12	Pelicans	New Orleans	Southwest	NBA	2008	12	
13	Pistons	Detroit	Central	NBA	1991	13	
14	Pelicans	New Orleans	Southwest	NBA	2020	14	
15	Timberwol...	Minnesota	Northwest	NBA	2016	15	
16	Mavericks	Dallas	Southwest	NBA	2019	16	
17	Bulls	Chicago	Central	NBA	2020	17	
18	Lakers	Los Angeles	Pacific	NBA	2018	18	
19	Suns	Phoenix	Pacific	NBA	2019	19	
20	76ers	Philadelphia	Atlantic	NBA	2001	20	

12. Display all the season metric data we have.

```
SELECT *  
FROM SeasonMetrics
```



The screenshot shows a SQL IDE interface. At the top, there's a toolbar with various icons and a 'Limit to 1000 rows' button. Below the toolbar, the SQL query is entered in a text area:

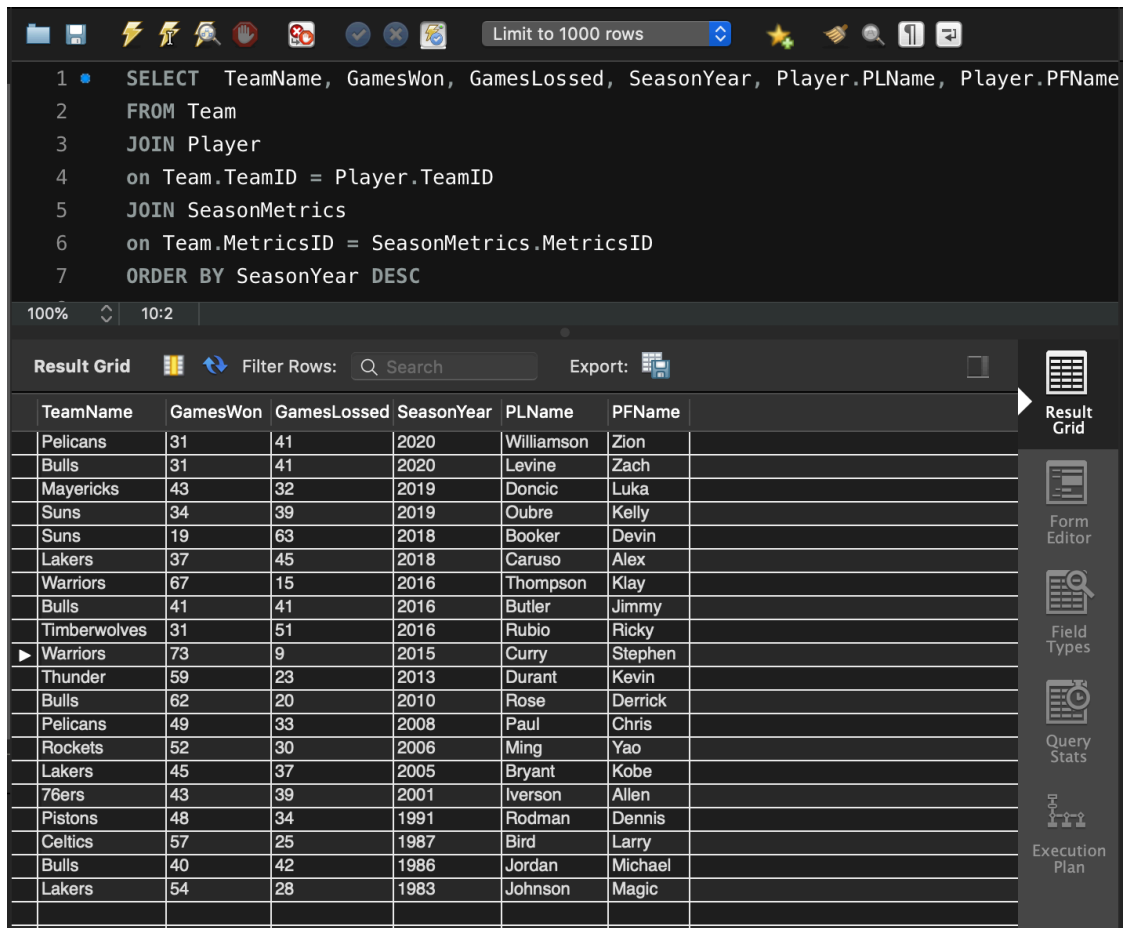
```
1 SELECT *  
2 FROM SeasonMetrics  
3
```

Below the query editor, there's a 'Result Grid' section. It includes a 'Filter Rows' search bar and an 'Export' button. The results are displayed in a table with the following columns: MetricsID, MetricSeasonYear, GamesWon, GamesLost, and ChampionshipFlag. The table contains 20 rows of data.

	MetricsID	MetricSeasonYear	GamesWon	GamesLost	ChampionshipFlag
▶	1	1986	40	42	N
	2	1987	57	25	N
	3	1983	54	28	N
	4	2006	52	30	N
	5	2005	45	37	N
	6	2015	73	9	N
	7	2016	67	15	Y
	8	2010	62	20	N
	9	2013	59	23	N
	10	2018	19	63	N
	11	2016	41	41	N
	12	2008	49	33	N
	13	1991	48	34	N
	14	2020	31	41	N
	15	2016	31	51	N
	16	2019	43	32	N
	17	2020	31	41	N
	18	2018	37	45	N
	19	2019	34	39	N
	20	2001	43	39	N

13. Show each player, the season year, and the wins and losses of their team. (Complex 9)

```
SELECT TeamName, GamesWon, GamesLosted, SeasonYear, Player.PLName,
Player.PFName
FROM Team
JOIN Player
on Team.TeamID = Player.TeamID
JOIN SeasonMetrics
on Team.MetricsID = SeasonMetrics.MetricsID
ORDER BY SeasonYear DESC
```

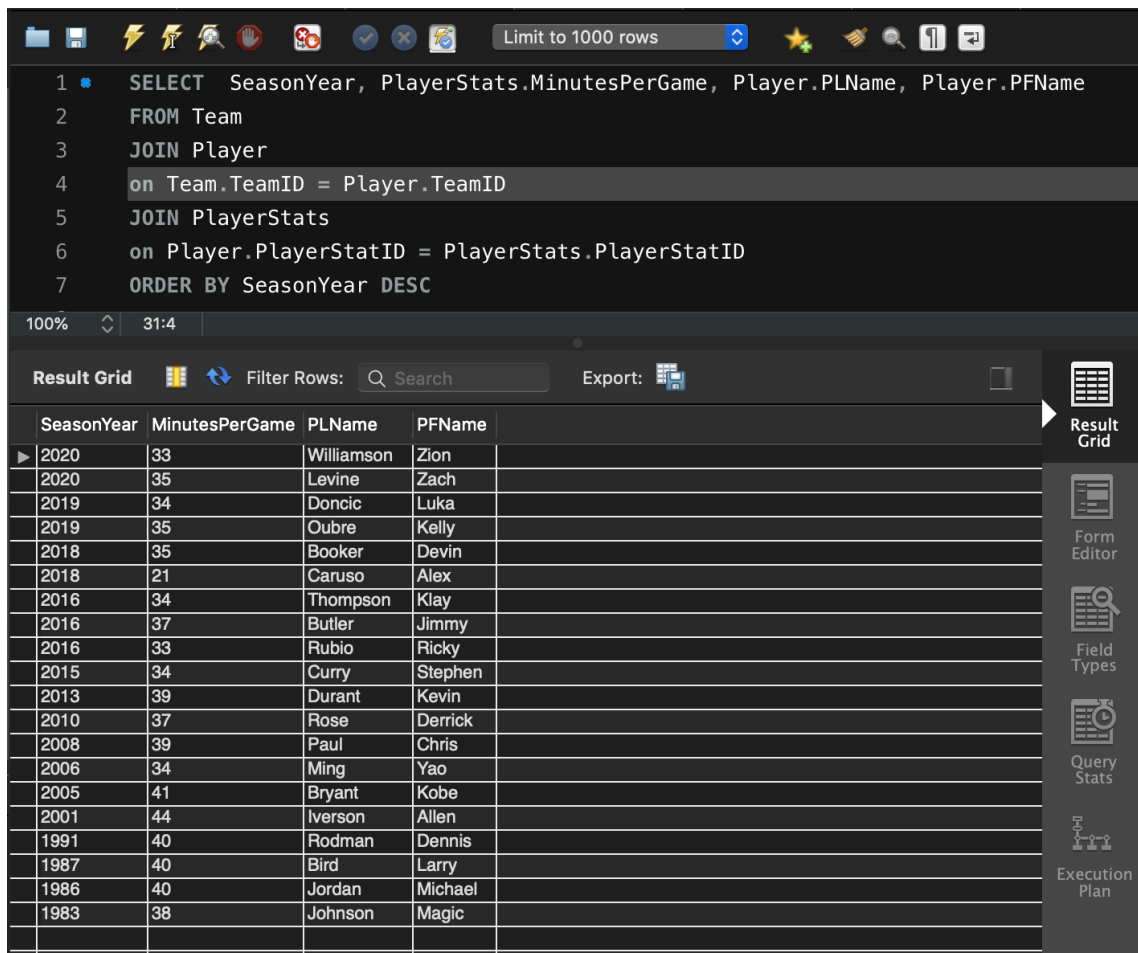


The screenshot shows a database query editor with a dark theme. The top toolbar includes icons for file operations, a search icon, and a 'Limit to 1000 rows' dropdown. The SQL query is entered in the main editor area. Below the query, the 'Result Grid' tab is active, displaying a table of results. The table has columns for TeamName, GamesWon, GamesLosted, SeasonYear, PLName, and PFName. The results are ordered by SeasonYear in descending order. On the right side of the interface, there is a vertical toolbar with icons for 'Result Grid', 'Form Editor', 'Field Types', 'Query Stats', and 'Execution Plan'.

TeamName	GamesWon	GamesLosted	SeasonYear	PLName	PFName
Pelicans	31	41	2020	Williamson	Zion
Bulls	31	41	2020	Levine	Zach
Mayericks	43	32	2019	Doncic	Luka
Suns	34	39	2019	Oubre	Kelly
Suns	19	63	2018	Booker	Devin
Lakers	37	45	2018	Caruso	Alex
Warriors	67	15	2016	Thompson	Klay
Bulls	41	41	2016	Butler	Jimmy
Timberwolves	31	51	2016	Rubio	Ricky
Warriors	73	9	2015	Curry	Stephen
Thunder	59	23	2013	Durant	Kevin
Bulls	62	20	2010	Rose	Derrick
Pelicans	49	33	2008	Paul	Chris
Rockets	52	30	2006	Ming	Yao
Lakers	45	37	2005	Bryant	Kobe
76ers	43	39	2001	Iverson	Allen
Pistons	48	34	1991	Rodman	Dennis
Celtics	57	25	1987	Bird	Larry
Bulls	40	42	1986	Jordan	Michael
Lakers	54	28	1983	Johnson	Magic

14. Show each player, their minutes per game, and their season year. (Complex 10)

```
SELECT SeasonYear, PlayerStats.MinutesPerGame, Player.PLName, Player.PFName
FROM Team
JOIN Player
on Team.TeamID = Player.TeamID
JOIN PlayerStats
on Player.PlayerStatID = PlayerStats.PlayerStatID
ORDER BY SeasonYear DESC
```



The screenshot shows a database query tool interface. The top toolbar includes icons for file operations, a search icon, and a 'Limit to 1000 rows' dropdown. The SQL editor contains the following query:

```
1 SELECT SeasonYear, PlayerStats.MinutesPerGame, Player.PLName, Player.PFName
2 FROM Team
3 JOIN Player
4 on Team.TeamID = Player.TeamID
5 JOIN PlayerStats
6 on Player.PlayerStatID = PlayerStats.PlayerStatID
7 ORDER BY SeasonYear DESC
```

Below the query editor, the 'Result Grid' tab is active, displaying a table with 5 columns: SeasonYear, MinutesPerGame, PLName, PFName, and an empty column. The results are ordered by SeasonYear in descending order.

SeasonYear	MinutesPerGame	PLName	PFName	
2020	33	Williamson	Zion	
2020	35	Levine	Zach	
2019	34	Doncic	Luka	
2019	35	Oubre	Kelly	
2018	35	Booker	Devin	
2018	21	Caruso	Alex	
2016	34	Thompson	Klay	
2016	37	Butler	Jimmy	
2016	33	Rubio	Ricky	
2015	34	Curry	Stephen	
2013	39	Durant	Kevin	
2010	37	Rose	Derrick	
2008	39	Paul	Chris	
2006	34	Ming	Yao	
2005	41	Bryant	Kobe	
2001	44	Iverson	Allen	
1991	40	Rodman	Dennis	
1987	40	Bird	Larry	
1986	40	Jordan	Michael	
1983	38	Johnson	Magic	

On the right side of the interface, there is a vertical toolbar with icons for 'Result Grid', 'Form Editor', 'Field Types', 'Query Stats', and 'Execution Plan'.

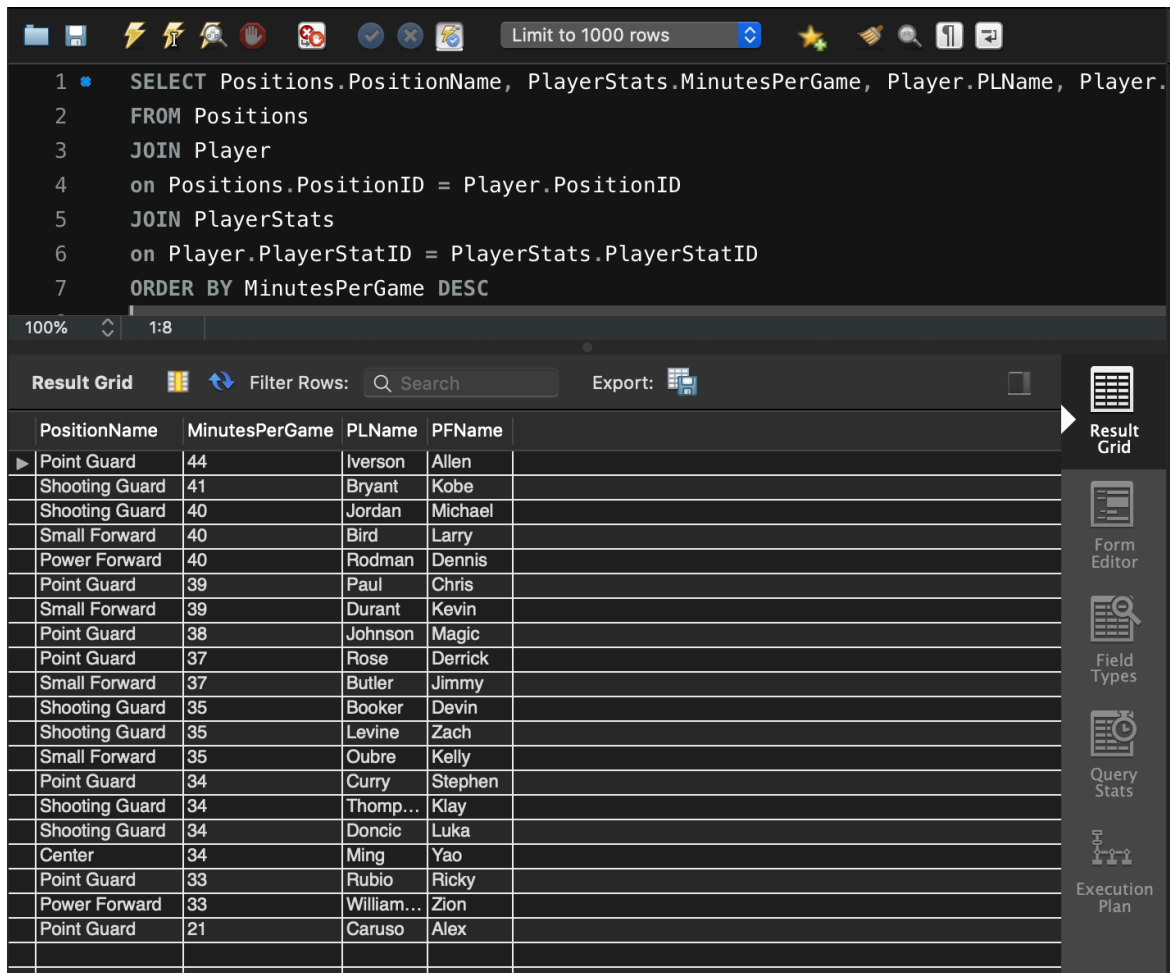
15. Show the list of players that won a championship in the given years from our data.
(Complex 11)

```
SELECT SeasonMetrics.ChampionshipFlag, Player.PLName, Player.PFName
FROM Team
JOIN Player
on Team.TeamID = Player.TeamID
JOIN SeasonMetrics
on Team.MetricsID = SeasonMetrics.MetricsID
WHERE SeasonMetrics.ChampionshipFlag = "Y"
```

[illegible]

16. Show all the players, their position name, and their minutes per game. (Complex 12)

```
SELECT Positions.PositionName, PlayerStats.MinutesPerGame, Player.PLName,
Player.PFName
FROM Positions
JOIN Player
on Positions.PositionID = Player.PositionID
JOIN PlayerStats
on Player.PlayerStatID = PlayerStats.PlayerStatID
ORDER BY MinutesPerGame DESC
```

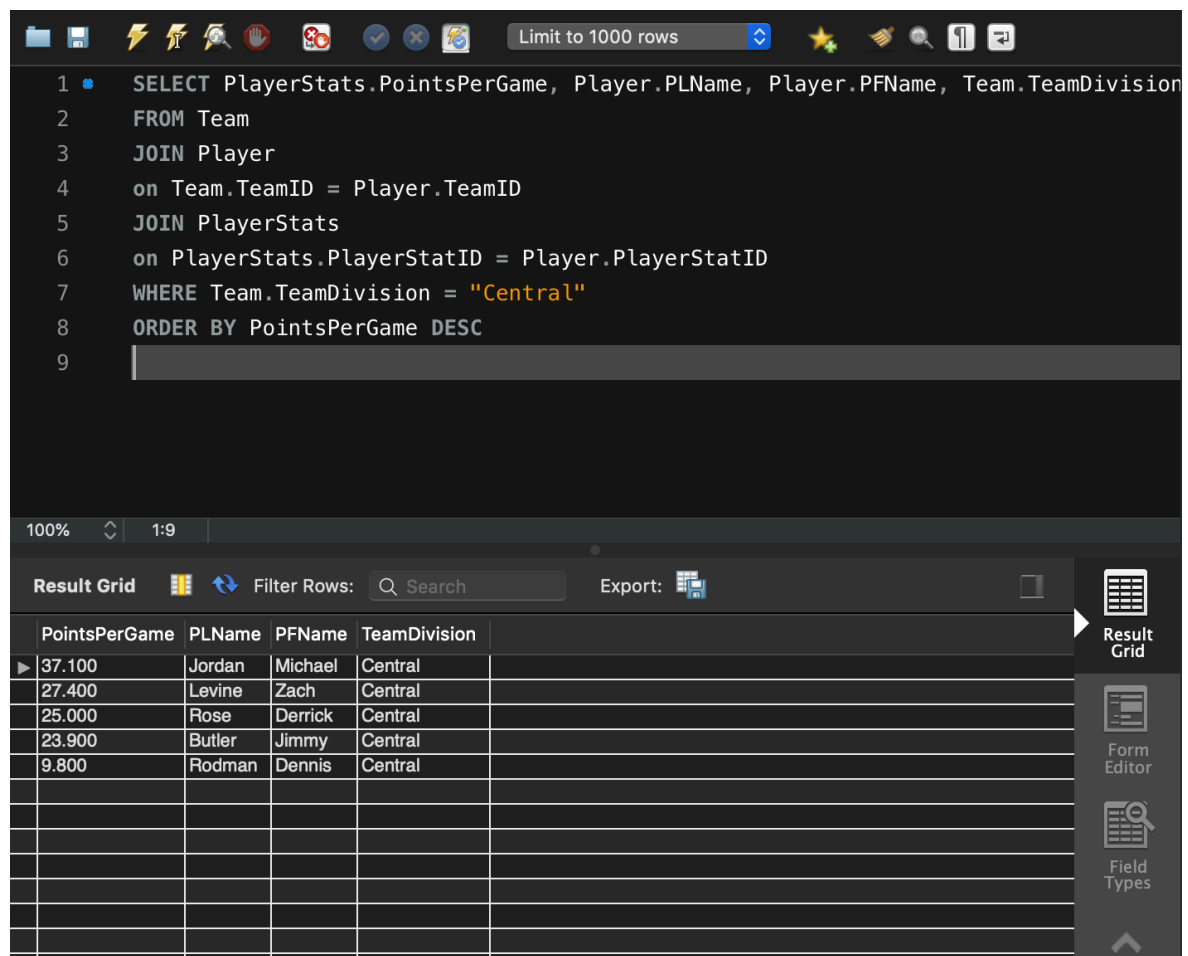


The screenshot shows a database query editor with a dark theme. The top toolbar includes icons for file operations, a 'Limit to 1000 rows' dropdown, and other utility icons. The SQL query is entered in the main text area. Below the query, the 'Result Grid' tab is active, displaying a table of results. The table has five columns: PositionName, MinutesPerGame, PLName, PFName, and an empty column. The results are sorted by MinutesPerGame in descending order. On the right side of the interface, there is a vertical toolbar with icons for 'Result Grid', 'Form Editor', 'Field Types', 'Query Stats', and 'Execution Plan'.

	PositionName	MinutesPerGame	PLName	PFName	
►	Point Guard	44	Iverson	Allen	
	Shooting Guard	41	Bryant	Kobe	
	Shooting Guard	40	Jordan	Michael	
	Small Forward	40	Bird	Larry	
	Power Forward	40	Rodman	Dennis	
	Point Guard	39	Paul	Chris	
	Small Forward	39	Durant	Kevin	
	Point Guard	38	Johnson	Magic	
	Point Guard	37	Rose	Derrick	
	Small Forward	37	Butler	Jimmy	
	Shooting Guard	35	Booker	Devin	
	Shooting Guard	35	Levine	Zach	
	Small Forward	35	Oubre	Kelly	
	Point Guard	34	Curry	Stephen	
	Shooting Guard	34	Thomp...	Klay	
	Shooting Guard	34	Doncic	Luka	
	Center	34	Ming	Yao	
	Point Guard	33	Rubio	Ricky	
	Power Forward	33	William...	Zion	
	Point Guard	21	Caruso	Alex	

17. Who has the highest points per game out of players in the central division? List all players in order by points per game. (Complex 13)

```
SELECT PlayerStats.PointsPerGame, Player.PLName, Player.PFName, Team.TeamDivision
FROM Team
JOIN Player
on Team.TeamID = Player.TeamID
JOIN PlayerStats
on PlayerStats.PlayerStatID = Player.PlayerStatID
WHERE Team.TeamDivision = "Central"
ORDER BY PointsPerGame DESC
```



The screenshot shows a database query editor with a dark theme. The top toolbar includes icons for file operations, execution, and search, along with a "Limit to 1000 rows" dropdown. The SQL query is entered in the main text area. Below the query, the "Result Grid" is displayed, showing the results of the query. The grid has five columns: PointsPerGame, PLName, PFName, TeamDivision, and an empty column. The results are sorted by PointsPerGame in descending order.

PointsPerGame	PLName	PFName	TeamDivision	
37.100	Jordan	Michael	Central	
27.400	Levine	Zach	Central	
25.000	Rose	Derrick	Central	
23.900	Butler	Jimmy	Central	
9.800	Rodman	Dennis	Central	

On the right side of the interface, there are icons for "Result Grid", "Form Editor", and "Field Types".

18. Who has the highest three-point attempts out of players in the southwest division? List all players in order by three-point attempts. (Complex 14)

```
SELECT PlayerStats.ThreePtAttemptPerGame, Player.PLName, Player.PFName,
Team.TeamDivision
FROM Team
JOIN Player
on Team.TeamID = Player.TeamID
JOIN PlayerStats
on PlayerStats.PlayerStatID = Player.PlayerStatID
WHERE Team.TeamDivision = "Southwest"
ORDER BY ThreePtAttemptPerGame DESC
```

The screenshot shows the SQL Editor interface. The top toolbar includes icons for file operations, execution, and search. A dropdown menu indicates "Limit to 1000 rows". The SQL query is as follows:

```
1 SELECT PlayerStats.ThreePtAttemptPerGame, Player.PLName, Player.PFName, Team.TeamID  
2 FROM Team  
3 JOIN Player  
4   ON Team.TeamID = Player.TeamID  
5 JOIN PlayerStats  
6   ON PlayerStats.PlayerStatID = Player.PlayerStatID  
7 WHERE Team.TeamDivision = "Southwest"  
8 ORDER BY ThreePtAttemptPerGame DESC  
9
```

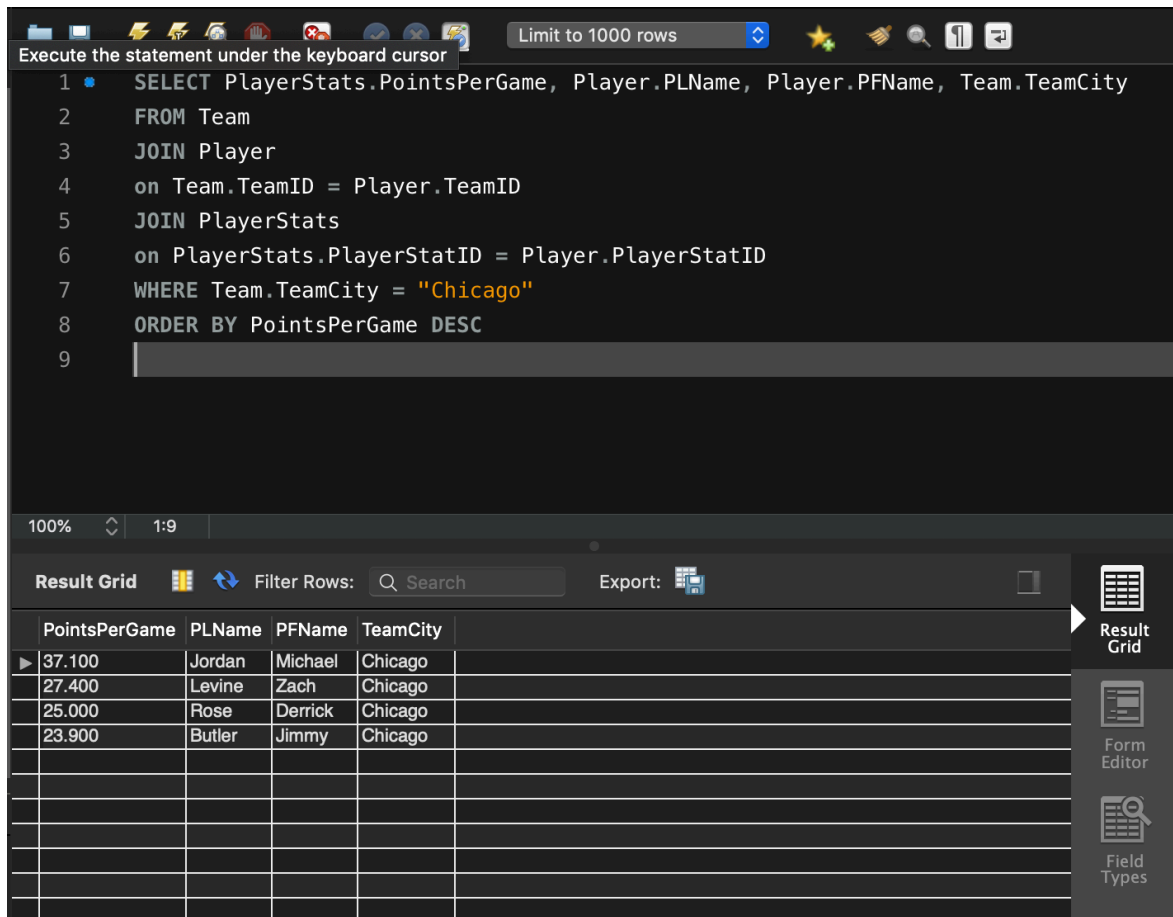
Below the editor, the "Result Grid" section displays the query results. It includes a search bar and an "Export" button. The results are shown in a table with columns: ThreePtAttemptPerGame, PLName, PFName, TeamDivision, and an unlabeled column.

	ThreePtAttemptPerGame	PLName	PFName	TeamDivision	
▶	8.900	Doncic	Luka	Southwest	
	2.300	Paul	Chris	Southwest	
	0.600	Williamson	Zion	Southwest	
	0.000	Ming	Yao	Southwest	

On the right side, there are three vertically stacked icons: "Result Grid" (selected), "Form Editor", and "Field Types".

19. Show the points per game of all players who played in Chicago. (Complex 15)

```
SELECT PlayerStats.PointsPerGame, Player.PLName, Player.PFName, Team.TeamCity
FROM Team
JOIN Player
on Team.TeamID = Player.TeamID
JOIN PlayerStats
on PlayerStats.PlayerStatID = Player.PlayerStatID
WHERE Team.TeamCity = "Chicago"
ORDER BY PointsPerGame DESC
```



The screenshot shows a database query editor interface. At the top, there's a toolbar with various icons and a "Limit to 1000 rows" dropdown. Below the toolbar, the SQL query is entered in a text area. The query is as follows:

```
1 SELECT PlayerStats.PointsPerGame, Player.PLName, Player.PFName, Team.TeamCity
2 FROM Team
3 JOIN Player
4 on Team.TeamID = Player.TeamID
5 JOIN PlayerStats
6 on PlayerStats.PlayerStatID = Player.PlayerStatID
7 WHERE Team.TeamCity = "Chicago"
8 ORDER BY PointsPerGame DESC
9
```

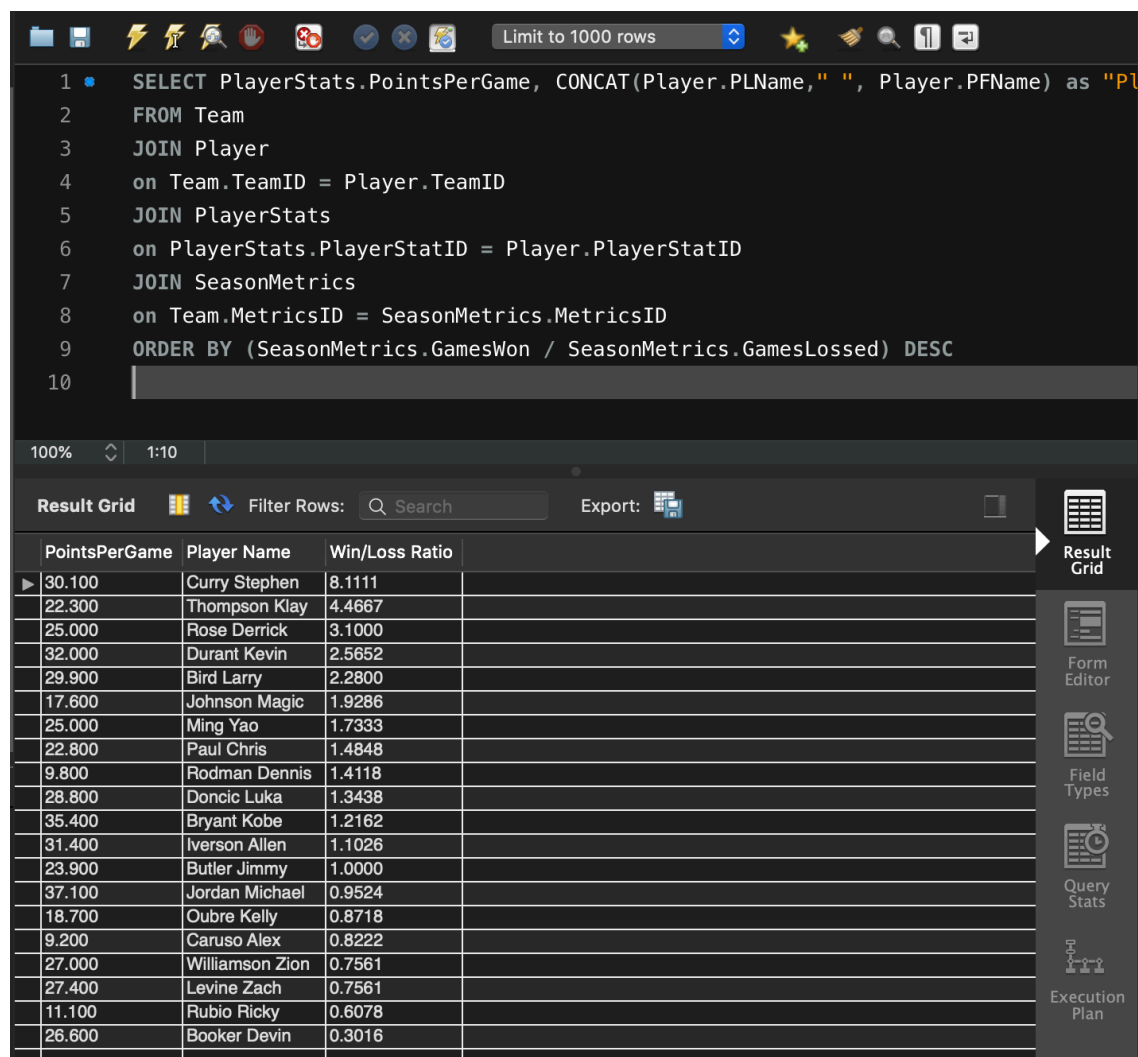
Below the query editor, there's a "Result Grid" section. It includes a "Filter Rows:" search bar and an "Export:" button. The result grid displays the following data:

PointsPerGame	PLName	PFName	TeamCity
37.100	Jordan	Michael	Chicago
27.400	Levine	Zach	Chicago
25.000	Rose	Derrick	Chicago
23.900	Butler	Jimmy	Chicago

On the right side of the interface, there are three icons: "Result Grid" (selected), "Form Editor", and "Field Types".

20. Show all player names, in one column, their win/loss ratio, and their points per game.
(Complex 16)

```
SELECT PlayerStats.PointsPerGame, CONCAT(Player.PLName," ", Player.PFName) as "Player Name", (SeasonMetrics.GamesWon / SeasonMetrics.GamesLost) as "Win/Loss Ratio"
FROM Team
JOIN Player
on Team.TeamID = Player.TeamID
JOIN PlayerStats
on PlayerStats.PlayerStatID = Player.PlayerStatID
JOIN SeasonMetrics
on Team.MetricsID = SeasonMetrics.MetricsID
ORDER BY (SeasonMetrics.GamesWon / SeasonMetrics.GamesLost) DESC
```



Limit to 1000 rows

```
1 SELECT PlayerStats.PointsPerGame, CONCAT(Player.PLName," ", Player.PFName) as "Player Name", (SeasonMetrics.GamesWon / SeasonMetrics.GamesLost) as "Win/Loss Ratio"
2 FROM Team
3 JOIN Player
4 on Team.TeamID = Player.TeamID
5 JOIN PlayerStats
6 on PlayerStats.PlayerStatID = Player.PlayerStatID
7 JOIN SeasonMetrics
8 on Team.MetricsID = SeasonMetrics.MetricsID
9 ORDER BY (SeasonMetrics.GamesWon / SeasonMetrics.GamesLost) DESC
10
```

100% 1:10

Result Grid Filter Rows: Search Export:

PointsPerGame	Player Name	Win/Loss Ratio	
30.100	Curry Stephen	8.1111	
22.300	Thompson Klay	4.4667	
25.000	Rose Derrick	3.1000	
32.000	Durant Kevin	2.5652	
29.900	Bird Larry	2.2800	
17.600	Johnson Magic	1.9286	
25.000	Ming Yao	1.7333	
22.800	Paul Chris	1.4848	
9.800	Rodman Dennis	1.4118	
28.800	Doncic Luka	1.3438	
35.400	Bryant Kobe	1.2162	
31.400	Iverson Allen	1.1026	
23.900	Butler Jimmy	1.0000	
37.100	Jordan Michael	0.9524	
18.700	Oubre Kelly	0.8718	
9.200	Caruso Alex	0.8222	
27.000	Williamson Zion	0.7561	
27.400	Levine Zach	0.7561	
11.100	Rubio Ricky	0.6078	
26.600	Booker Devin	0.3016	

Result Grid Form Editor Field Types Query Stats Execution Plan