

به نام خدا

کارگاه ۵

*از پایتون استفاده شده است.

*insert فقط جداول Team و Players دارند و سعی شده است که تمام مشکلات موقع insert رسیدگی و حل شود.

*delete فقط جدول Team دارد.

*تمام کوئری های کارگاه فاز ۴ را دارد.

*چندتا فایل به نام های connect.py و create_table.py و insert_test.py برای توضیح روند کار به چه شکل است وجود دارد.

*فایل اصلی پروژه به نام project.py است.

*خروجی های برنامه در آخر قرار گرفته است.

برای استفاده از sqlite3 در پایتون کافی است آن را import بکنیم و نیازی به نصب پکیج خاصی نیست.

در فایل (connect.py)

```
import sqlite3

def create_connection(db_file):
    conn = sqlite3.connect(db_file)

if __name__ == '__main__':
    create_connection(r"db_pro.db")
```

همان طور که در کد مشاهده می کنید ابتدا sqlite3 ، import شده است و سپس با دستور connect به دیتابیس وصل شده ایم حال اگر اسم آن دیتابیس که نوشته شده است "db_pro.db" موجود نباشد آن را میسازد.

*حال به سراغ ساختن جداول دیتابیس میرویم(در فایل (create table.py)):

یک تابع می نویسیم تا کدی مربوط `create table` را اجرا بکند و آن جدول را ایجاد بکند در دیتابیس.

```
def create_table(conn, create_table_sql):  
    c = conn.cursor()  
    c.execute(create_table_sql)
```

ورودی این تابع این تابع خروجی تابع `connect` و کد `sql` برای `create table` است و حال بعد از وصل شدن به دیتابیس ، `cursor` را صدا میزنیم تا به ما اجازه دهد که توابعی مانند `execute` را صدا بزنیم تا بتوانیم کد های `sql` را اجرا بکنیم بر روی دیتابیس.

*کد وصل شدن به دیتابیس:

```
conn = sqlite3.connect("db_pro.db")
```

حال باید کد مربوط به `create table` را بنویسیم، که فقط دو نمونه از آن آمده است که نسخه کامل آن در فایل `py` موجود است.

ابتدا جدول `team` خود را به صورت `string` به یک متغیر می دهیم.

```
team = """CREATE TABLE Team(  
    teamID INTEGER PRIMARY KEY AUTOINCREMENT,  
    teamName VARCHAR(255) NOT NULL,  
    LeagueName VARCHAR(255) NOT NULL,  
    teamWinNo INT NULL,  
    teamCupNo INT NULL,  
    teamLoseNo INT NULL,  
    PenaltyNo INT NULL,  
    Salary REAL NULL,  
    teamInjuryNo INT NULL,  
    UNIQUE(teamName, LeagueName)  
);"""
```

حال جدول `players`.

```
players = """CREATE TABLE Players(  
    playersID INTEGER PRIMARY KEY  
    AUTOINCREMENT,  
    playersName VARCHAR(255) NOT NULL ,  
    playersNationality VARCHAR(255) NOT  
    NULL ,  
    playersGoalNo INT NULL,  
    playersCupNo INT NULL,  
    AssistNo INT NULL,
```

```

        playersGameNo INT NULL,
        DurationOfInjury REAL NULL,
        playersInjuryNo INT NULL,
        playersCardNo INT NULL,
        StartTheContract TEXT NOT NULL,
        EndTheContract TEXT NOT NULL,
        TeamID INTEGER NOT NULL,
        UNIQUE(playersName, playersNationality,
TeamID) ,
        CONSTRAINT FK_Team_tID_tName_LeagueName
FOREIGN KEY(TeamID)
        REFERENCES Team(teamID)
    );"""

```

همانطور که مشاهده می کنید دقیقاً همان کد **Sql** را به آن داده ایم سپس این موارد را به تابع **create_table** که خودمان درست کرده ایم می دهیم.

```

if conn is not None:
    create_table(conn, team)
    create_table(conn, players)

```

دلیل شرط این است که اگر نتوانست به دیتابیس وصل شود دیگر اجرا نشود و به جای آن **error** چاپ نکند.

*حال به سراغ وارد کردن داده ها در جدول میرویم:

در فایل **insert_test.py** قرار دارند. با توابعی مانند **create_team** و ... ما داده خود را وارد میکنیم به عنوان مثال:

```

def create_team(connect, team):
    sql = ''' INSERT INTO
Team(teamID,teamName,LeagueName,teamWinNo,teamCupNo,tea
mLoseNo, PenaltyNo, Salary, teamInjuryNo)
        VALUES(?,?,?,?,?,?,?,?) '''
    cur = connect.cursor()
    cur.execute(sql, team)
    connect.commit()

```

این تابع یک تیم وارد میکند، به صورت پویا این کار را میکند کافی است ما مقادیر لازم را بدهیم و آن را در مقدار **sql** قرار می دهد و سپس آن را با **execute** به دیتابیس خود وارد میکنیم.

حال برای هندل کردن مشکلات در این جدول به صورت زیر عمل میکنیم:

به عنوان مثال teamID, teamName, LeagueName باید یونیک باشند پس نمی توان دو تا تیم با یک id وارد کرد. پس داریم:

```
try:
    team = (1, 'Barcelona', 'Laliga', 600, 74, 200,
100, 13799999.999999999999999, 10)
    create_team(connect, team)
    team = (2, 'Liverpool', 'Premier', 500, 40, 210,
30, 9000000.0, 10)
    create_team(connect, team)
    team = (3, 'Ajax', 'Eredivisie', 450, 35, 130, 40,
5000000.0, 20)
    create_team(connect, team)
except sqlite3.IntegrityError:
    print("TeamID,teamName,LeagueNameis UNIQUE")
```

حال به سراغ جدول `players` میرویم از آنجا که در این جدول باید `teamID` هر بازیکن را وارد بکنیم پس باید دقت شود `teamID` وارد شده در جدول `Team` وجود داشته باشد، پس داریم:

```
players = (1, 'Lionel
Messi', 'Argentina', 750, 40, 221, 850, 10.0, 2, 10, '2003/01/01
', '2022/01/01', 1)
create_players(connect, players)
players = (8, 'Mateo
Kovačić', 'Austria', 20, 12, 41, 250, 3.0, 1, 13, '2018/01/03', '
2024/01/09', 9)
teamID = ""SELECT teamID FROM Team""
c = connect.cursor()
temp = c.execute(teamID)
flag = False
for row in temp:
    if players[12] == row[0]:
        flag = True
if flag == True:
    create_players(connect, players)
else:
    print("TeamID don't exist")
```

بازیکن اول Lionel Messi چون teamID آن برابر یک است ساخته می شود اما بازیکن دوم چون تنها تا teamID شماره ۳ ساخته شده است (باتوجه به قطعه کد بالا) و teamID آن برابر ۹ است پس نباید این بازیکن ساخته شود، پس ابتدا همه teamID های موجود در جدول Team را پیدا میکنیم و بررسی میکنیم

که آیا teamID ، players برابر آن ها است یا خیر، اگر نبود که یک پیغام نشان می دهد و بازیکن را نمی سازد.

*بقیه داده ها را مانند بخش create table به صورت String می دهیم چون مراحل همین گونه است اما هنگامی که کاربر یک داده وارد میکند (که در جلوتر آمده است) این ها کامل بررسی می شود.

*با insert کردن همه داده ها دیتابیس ما ساخته می شود.

حال بعد از ساخته شدن دیتابیس به سراغ کوئری ها و وارد کردن و حذف کردن و سایر دستورات که ممکن است کاربر وارد بکند می رویم.(در فایل project.py)

*شامل تمام کوئری های فاز ۴ و insert برای دوتا جدول و حذف از یک جدول می شود.

ابتدا sqlite3 را import میکنیم سپس تابع های create_team و create_players و برای بقیه جداول هم نوشته شده است، به عنوان مثال تابع ساختن بازیکن به صورت زیر است:

```
def create_players(connect, players):
    sql = '''INSERT INTO
Players(playersID,playersName,playersNationality,player
sGoalNo,playersCupNo,AssistNo,playersGameNo,DurationOfI
njury,playersInjuryNo,playersCardNo,StartTheContract,En
dTheContract,TeamID)
VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)
'''
    cur = connect.cursor()
    cur.execute(sql, players)
    connect.commit()
```

که ورودی آن خروجی تابع connect برای صدا زدن cursor و سپس execute را برای اجرای کد صدا می زند.

در تابع main ، ابتدا به دیتابیس خود وصل می شویم. و از کاربر میخواهیم که کار مورد نیاز خود را (insert,delete,query) انتخاب بکند و سپس در صورت انتخاب insert و delete باید جدول خود را انتخاب کند(دقت کنید که برای insert جداول Team و Players این کار را انجام می دهند و برای delete هم جدول Team) حال ممکن است حالات زیر رخ دهد.

۱.گزینه insert و جدول Team انتخاب می شود.

ابتدا ما همه تیم ها را میگیریم و نشان می دهیم و سپس teamID را در یک آرایه ذخیره میکنیم برای چک کردن ID که کاربر وارد میکند که درست باشد و سپس teamName و LeagueName را با هم به صورت sting ذخیره میکنیم چون این دو با هم unique هستند. و باید ورودی کاربر چک شود که unique باشد.

که قطع کد به صورت زیر است:

```
select_team = """SELECT * FROM Team"""
list_TeamName_League = []
res = cursor.execute(select_team)
temp = 0
for row in res:
    print(row)
    temp = int(row[0])
    list_TeamName_League.append(row[1]+row[2])
```

حال به سراغ ورودی گرفتن از کاربر می رویم و سپس همه ورودی ها چک می شود برای نوع متغیر ها که درست وارد بکنند مثلا یک مقدار که باید در جدول int باشد کاربر نمی تواند string وارد بکند.

سپس در قطع کد زیر unique بودن آن را چک میکنیم.

```
for i in range(0,len(list_TeamName_League)):
    if (name + league) == list_TeamName_League[i]:
        unique_name = True
```

و اگر مشکلی وجود نداشت تیم جدید را می سازد.

۲.گزینه insert و جدول Players انتخاب شود.

مانند جدول Team است ولی مقدار unique و نوع مقادیر و تعداد خصوصیات جدول متفاوت است.

نکته ایی که باید رعایت شود ورودی start the contract و end the contract است که باید مثل "2020/10/05" باشد. در کد زیر نحوه چک شدن آمده است:

```
if len(StartTheContract) < 10:
    print("start the contract isn't correct.")
    check = True
elif StartTheContract[4] != "/" and StartTheContract[7]
!= "/":
    print("shape of start the contract isn't correocr.
it must be like 2020/01/01.")
    check = True
```

۳. گزینه delete و جدول Team انتخاب شود.

کاربر teamID موردنظرش را برای حذف وارد میکند. سپس چک می شود که آیا بازیکنی به این تیم وصل است یا خیر در صورت وصل بودن باید اروری به کاربر نشان می دهد.

۴. گزینه query :

کوئری های فاز ۴ به کاربر نشان داده می شود. یکی از آن ها را انتخاب می کند و خروجی را به کاربر نمایش می دهد.

حال به سراغ نمونه ایی از ورودی ها و خروجی ها می رویم.

```
C:\Users\Hojjat\Desktop\pro5_sqlite\Scripts\python.exe C:/Users/Hojjat/PycharmProjects/pro5_sqlite/project.py
Select One of them : Insert(1) , Delete(2) , Query(3):
1
Select a table: Team(1) , Players(2) , Position(3) , PlayersPosition(4) , Coach(5) , CoachLeadTeam(6) , TheManager(7) , Manager_Manage_Team(8) , Sponser(9) , TeamDealSponser(10)
1
Team Table:
teamID,teamName,LeagueName,teamWinNo,teamCupNo,teamLoseNo,PenaltyNo,Salary,teamInjuryNo
(1, 'Barcelona', 'Laliga', 600, 74, 200, 100, 13800000.0, 10)
(2, 'Liverpool', 'Premier', 500, 40, 210, 30, 9000000.0, 10)
(3, 'Ajax', 'Eredivisie', 450, 35, 130, 40, 5000000.0, 20)
(4, 'Real Madrid', 'Laliga', 620, 91, 178, 150, 20000000.0, 34)
(5, 'Juventus', 'Serie A', 537, 63, 230, 67, 11800000.0, 41)
(6, 'Manchester United', 'Premier', 591, 72, 201, 76, 10800000.0, 51)
(7, 'Paris Saint-Germain', 'Ligue 1', 478, 51, 183, 68, 20800000.0, 23)
(8, 'Manchester City', 'Premier', 501, 69, 209, 63, 12800000.0, 41)
(9, 'Chelsea', 'Premier', 596, 71, 203, 71, 8000000.0, 30)
(10, 'FC Bayern Munich', 'Bundesliga', 598, 74, 200, 86, 13000000.0, 47)
(11, 'Borussia Dortmund', 'Bundesliga', 421, 62, 217, 106, 9000000.0, 43)
(12, 'Atlético Madrid', 'Laliga', 432, 53, 267, 101, 10000000.0, 36)
(13, 'Arsenal', 'Premier', 569, 68, 278, 91, 11000000.0, 43)
(14, 'FC Porto', 'Primeira Liga', 378, 37, 307, 76, 3000000.0, 32)
(15, 'Olympique Lyonnais', 'Ligue 1', 392, 41, 301, 103, 4000000.0, 47)
(16, 'Test', 'Test', 0, 0, 0, 0, 0.0, 0)
Insert you're data:
```

در اینجا کاربر insert و Team را انتخاب کرده است حال باید داده مورد نظر خود را وارد کند:

```
Insert you're data:
17 Insert Insert 1 1 1 1 1 1
```

و حال ورودی خود را ثبت می کند ، که به شکل زیر است:

```

C:\Users\Hojjat\Desktop\pro5_sqlite\Scripts\python.exe C:/Users/Hojjat/PycharmProjects/pro5_sqlite/project.py
Select One of them : Insert(1) , Delete(2) , Query(3):
1
Select a table: Team(1) , Players(2) , Position(3) , PlayersPosition(4) , Coach(5) , CoachLeadTeam(6) , TheManager(7)
1
Team Table:
teamID,teamName,LeagueName,teamWinNo,teamCupNo,teamLoseNo,PenaltyNo,Salary,teamInjuryNo
(1, 'Barcelona', 'Laliga', 600, 74, 200, 100, 13800000.0, 10)
(2, 'Liverpool', 'Premier', 500, 40, 210, 30, 9000000.0, 10)
(3, 'Ajax', 'Eredivisie', 450, 35, 130, 40, 5000000.0, 20)
(4, 'Real Madrid', 'Laliga', 620, 91, 178, 150, 20000000.0, 34)
(5, 'Juventus', 'Serie A', 537, 63, 230, 67, 11800000.0, 41)
(6, 'Manchester United', 'Premier', 591, 72, 201, 76, 10800000.0, 51)
(7, 'Paris Saint-Germain', 'Ligue 1', 478, 51, 183, 68, 20800000.0, 23)
(8, 'Manchester City', 'Premier', 501, 69, 209, 63, 12800000.0, 41)
(9, 'Chelsea', 'Premier', 596, 71, 203, 71, 8000000.0, 30)
(10, 'FC Bayern Munich', 'Bundesliga', 598, 74, 200, 86, 13000000.0, 47)
(11, 'Borussia Dortmund', 'Bundesliga', 421, 62, 217, 106, 9000000.0, 43)
(12, 'Atlético Madrid', 'Laliga', 432, 53, 267, 101, 10000000.0, 36)
(13, 'Arsenal', 'Premier', 569, 68, 278, 91, 11000000.0, 43)
(14, 'FC Porto', 'Primeira Liga', 378, 37, 307, 76, 3000000.0, 32)
(15, 'Olympique Lyonnais', 'Ligue 1', 392, 41, 301, 103, 4000000.0, 47)
(16, 'Test', 'Test', 0, 0, 0, 0, 0.0, 0)
Insert you're data:
17 Insert Insert 1 1 1 1 1 1
insert is done.

Process finished with exit code 0

```

که با موفقیت وارد شده است.

حال یک ورودی دیگر را تست میکنیم:

```

C:\Users\Hojjat\Desktop\pro5_sqlite\Scripts\python.exe C:/Users/Hojjat/PycharmProjects/pro5_sqlite/project.py
Select One of them : Insert(1) , Delete(2) , Query(3):
1
Select a table: Team(1) , Players(2) , Position(3) , PlayersPosition(4) , Coach(5) , CoachLeadTeam(6) , TheManager(7)
1
Team Table:
teamID,teamName,LeagueName,teamWinNo,teamCupNo,teamLoseNo,PenaltyNo,Salary,teamInjuryNo
(1, 'Barcelona', 'Laliga', 600, 74, 200, 100, 13800000.0, 10)
(2, 'Liverpool', 'Premier', 500, 40, 210, 30, 9000000.0, 10)
(3, 'Ajax', 'Eredivisie', 450, 35, 130, 40, 5000000.0, 20)
(4, 'Real Madrid', 'Laliga', 620, 91, 178, 150, 20000000.0, 34)
(5, 'Juventus', 'Serie A', 537, 63, 230, 67, 11800000.0, 41)
(6, 'Manchester United', 'Premier', 591, 72, 201, 76, 10800000.0, 51)
(7, 'Paris Saint-Germain', 'Ligue 1', 478, 51, 183, 68, 20800000.0, 23)
(8, 'Manchester City', 'Premier', 501, 69, 209, 63, 12800000.0, 41)
(9, 'Chelsea', 'Premier', 596, 71, 203, 71, 8000000.0, 30)
(10, 'FC Bayern Munich', 'Bundesliga', 598, 74, 200, 86, 13000000.0, 47)
(11, 'Borussia Dortmund', 'Bundesliga', 421, 62, 217, 106, 9000000.0, 43)
(12, 'Atlético Madrid', 'Laliga', 432, 53, 267, 101, 10000000.0, 36)
(13, 'Arsenal', 'Premier', 569, 68, 278, 91, 11000000.0, 43)
(14, 'FC Porto', 'Primeira Liga', 378, 37, 307, 76, 3000000.0, 32)
(15, 'Olympique Lyonnais', 'Ligue 1', 392, 41, 301, 103, 4000000.0, 47)
(16, 'Test', 'Test', 0, 0, 0, 0, 0.0, 0)
(17, 'Insert', 'Insert', 1, 1, 1, 1, 1.0, 1)
Insert you're data:
18 Ajax Eredivisie 0 0 0 0 0 0
Team name must be unique.

Process finished with exit code 0
|

```


همانطور که مشاهده میکنید، چون اسم تیم و اسم لیگ آن با یکی از داده های قبلی برابر است ارور داده است و اینکه تیم Insert که اول وارد شده بود وارد شده است.

حال به سراغ delete می رویم.

```
C:\Users\Hojjat\Desktop\pro5_sqlite\Scripts\python.exe C:/Users/Hojjat/PycharmProjects/pro5_sqlite/project.py
Select One of them : Insert(1) , Delete(2) , Query(3):
2
Select a table: Team(1) , Players(2) , Position(3) , PlayersPosition(4) , Coach(5) , CoachLeadTeam(6) , TheManager
1
Team Table:
teamID,teamName,LeagueName,teamWinNo,teamCupNo,teamLoseNo,PenaltyNo,Salary,teamInjuryNo
(1, 'Barcelona', 'Laliga', 600, 74, 200, 100, 13800000.0, 10)
(2, 'Liverpool', 'Premier', 500, 40, 210, 30, 9000000.0, 10)
(3, 'Ajax', 'Eredivisie', 450, 35, 130, 40, 5000000.0, 20)
(4, 'Real Madrid', 'Laliga', 620, 91, 178, 150, 20000000.0, 34)
(5, 'Juventus', 'Serie A', 537, 63, 230, 67, 11800000.0, 41)
(6, 'Manchester United', 'Premier', 591, 72, 201, 76, 10800000.0, 51)
(7, 'Paris Saint-Germain', 'Ligue 1', 478, 51, 183, 68, 20800000.0, 23)
(8, 'Manchester City', 'Premier', 501, 69, 209, 63, 12800000.0, 41)
(9, 'Chelsea', 'Premier', 596, 71, 203, 71, 8000000.0, 30)
(10, 'FC Bayern Munich', 'Bundesliga', 598, 74, 200, 86, 13000000.0, 47)
(11, 'Borussia Dortmund', 'Bundesliga', 421, 62, 217, 106, 9000000.0, 43)
(12, 'Atlético Madrid', 'Laliga', 432, 53, 267, 101, 10000000.0, 36)
(13, 'Arsenal', 'Premier', 569, 68, 278, 91, 11000000.0, 43)
(14, 'FC Porto', 'Primeira Liga', 378, 37, 307, 76, 3000000.0, 32)
(15, 'Olympique Lyonnais', 'Ligue 1', 392, 41, 301, 103, 4000000.0, 47)
(16, 'Test', 'Test', 0, 0, 0, 0, 0.0, 0)
(17, 'Insert', 'Insert', 1, 1, 1, 1, 1.0, 1)
Enter TeamID to delete:
17
delete is done.

Process finished with exit code 0
```

تیمی که اول ایجاد کرده بودیم بدون مشکلی حذف شد. حالا به سراغ یک تیم دیگر می رویم.

```
C:\Users\Hojjat\Desktop\pro5_sqlite\Scripts\python.exe C:/Users/Hojjat/PycharmProjects/pro5_sqlite/project.py
Select One of them : Insert(1) , Delete(2) , Query(3):
2
Select a table: Team(1) , Players(2) , Position(3) , PlayersPosition(4) , Coach(5) , CoachLeadTeam(6) , TheMana
1
Team Table:
teamID,teamName,LeagueName,teamWinNo,teamCupNo,teamLoseNo,PenaltyNo,Salary,teamInjuryNo
(1, 'Barcelona', 'Laliga', 600, 74, 200, 100, 13800000.0, 10)
(2, 'Liverpool', 'Premier', 500, 40, 210, 30, 9000000.0, 10)
(3, 'Ajax', 'Eredivisie', 450, 35, 130, 40, 5000000.0, 20)
(4, 'Real Madrid', 'Laliga', 620, 91, 178, 150, 20000000.0, 34)
(5, 'Juventus', 'Serie A', 537, 63, 230, 67, 11800000.0, 41)
(6, 'Manchester United', 'Premier', 591, 72, 201, 76, 10800000.0, 51)
(7, 'Paris Saint-Germain', 'Ligue 1', 478, 51, 183, 68, 20800000.0, 23)
(8, 'Manchester City', 'Premier', 501, 69, 209, 63, 12800000.0, 41)
(9, 'Chelsea', 'Premier', 596, 71, 203, 71, 8000000.0, 30)
(10, 'FC Bayern Munich', 'Bundesliga', 598, 74, 200, 86, 13000000.0, 47)
(11, 'Borussia Dortmund', 'Bundesliga', 421, 62, 217, 106, 9000000.0, 43)
(12, 'Atlético Madrid', 'Laliga', 432, 53, 267, 101, 10000000.0, 36)
(13, 'Arsenal', 'Premier', 569, 68, 278, 91, 11000000.0, 43)
(14, 'FC Porto', 'Primeira Liga', 378, 37, 307, 76, 3000000.0, 32)
(15, 'Olympique Lyonnais', 'Ligue 1', 392, 41, 301, 103, 4000000.0, 47)
(16, 'Test', 'Test', 0, 0, 0, 0, 0.0, 0)
Enter TeamID to delete:
3
You can't delete this team because, this team have players.

Process finished with exit code 0
```

همانطور که مشاهده می کنید تیم 1 چون بازیکن دارد حذف نمی شود.

حال به سراغ کوئری های کارگاه فاز ۴ می رویم.

```
C:\Users\Hojjat\Desktop\pro5_sqlite\Scripts\python.exe C:/Users/Hojjat/PycharmProjects/pro5_sqlite/project.py
Select One of them : Insert(1) , Delete(2) , Query(3):
3
1. گلزن ترین بازیکن به همراه اسم و اسم باشگاه و تعداد گل او:
2. بازیکن هایی که صفر گل زده اند به همراه اسم و نام باشگاه و پوزیشنی که در آن بازی می کنند:
3. کدام اسپانسر تعداد بیشتری قرار داد با تیم ها بسته است:
4. با توجه به کوئری بالا، تیم هایی که با اسپانسری که بیشترین تعداد قرار داد را دارند به همراه اسم لیگ آن ها:
5. بازیکن هایی که کاربر حجت در یک بازی مشاهده کرده است
6. بازیکنانی که در تیمی که بزرگترین ورزشگاه را دارد را بازی می کنند به همراه ملیت آن ها:
7. بازیکنانی که از قبل سال 2009 با یک تیم قرار داد بسته اند به همراه اسم تیم و ملیت آن ها:
8. بازیکنانی که قبل از ساخت یا تعمیر ورزشگاه عضو باشگاه شده اند به همراه اسم تیم و ملیت:
9. بازیکنان و مربیانی که در یک تیم از یک کشور هستند:
10. بازیکنانی که در پر کارت ترین بازی شرکت کردند به همراه اسم تیم و اسم لیگ:
11. اسپانسر های تیم های لیگ Premier به همراه اسم تیم:
12. اسم استادیوم ها و تیم ها و اسپانسرهایی که باشگاه FC Bayern Munich از آن ها بازیکن خریداری کرده است:
13. اسم تیم و مدیران و تیم هایی که مقابل آن ها مهمان بوده است که اسم اسپانسر آن تیم Yokohama است:
Select one of them:
13
('Mr John', 'Chelsea', 'Paris Saint-Germain')

Process finished with exit code 0
|
```

کوئری مورد نظر خود را انتخاب میکنیم و سپس خروجی آن نمایش داده می شود.