

NBA Atlantic Division의 주요 플레이어들에 대한 정보를 보고, 북마크 기능을 제공하는 어플리케이션입니다.

## 1. DB Application에 대한 설명

기본 페이지

### nba athlentic division & key players

#### basic stats of players in 2019-2020 Athlentic Division

#### major players of season

season:

#### player bookmark

player:   
mytag:

기능 1

모든 선수들의 2019~2020시즌 평균 1차스탯을 보여줍니다.

#### basic stats of players in 2019-2020 Athlentic Division

| Player          | PPG  | APG  | RPG  |
|-----------------|------|------|------|
| Al Horford      | 11.9 | 4.0  | 6.8  |
| Seth Curry      | 12.5 | 2.7  | 2.4  |
| Kevin Durant    | 26.9 | 5.6  | 7.1  |
| OG Anunoby      | 13.3 | 1.9  | 5.4  |
| Derrick Rose    | 14.9 | 4.2  | 2.9  |
| Marcus Smart    | 19.9 | 4.9  | 4.0  |
| Gordon Hayward  | 17.5 | 4.1  | 6.7  |
| Jayson Tatum    | 23.4 | 3.0  | 7.0  |
| Kyle Lowry      | 18.3 | 7.4  | 5.2  |
| Ben Simmons     | 15.4 | 7.4  | 7.6  |
| Joel Embiid     | 25.8 | 2.9  | 11.1 |
| Jarrett Allen   | 13.2 | 10.4 | 1.7  |
| JayLen Brown    | 22.5 | 2.4  | 5.1  |
| Robert Williams | 6.6  | 1.4  | 5.7  |
| Blake Griffin   | 10.0 | 2.4  | 4.7  |
| Kyrie Irving    | 30.4 | 6.2  | 5.0  |
| Taj Gibson      | 5.8  | 0.8  | 5.0  |
| RJ Barrett      | 16.0 | 2.8  | 5.4  |
| Fred VanVleet   | 18.6 | 6.5  | 4.0  |
| Kemba Walker    | 11.7 | 5.3  | 3.7  |
| Evan Fournier   | 13.0 | 3.1  | 3.3  |
| Pascal Siakam   | 22.2 | 4.0  | 7.3  |
| Kevin Knox      | 5.2  | 0.7  | 2.2  |
| Julius Randle   | 21.8 | 4.6  | 10.0 |
| Josh Richardson | 12.9 | 2.8  | 3.3  |
| Tobias Harris   | 19.6 | 3.4  | 6.9  |
| Reggie Bullock  | 9.5  | 1.5  | 2.9  |
| Dennis Smith    | 5.5  | 2.9  | 2.3  |
| Joe Harris      | 14.3 | 2.0  | 4.0  |
| DeAndre Jordan  | 7.9  | 1.8  | 8.8  |
| James Harden    | 24.6 | 10.9 | 8.5  |
| Serge Ibaka     | 13.3 | 1.6  | 7.5  |

## 기능 2

입력한 시즌(연도)의 각 팀별 주요 선수 로스터를 출력합니다.

### major players of season

season:

[get result](#)

| Team               | Player_id | Player          |
|--------------------|-----------|-----------------|
| Boston Celtics     | 001       | Jayson Tatum    |
| Boston Celtics     | 002       | JayLen Brown    |
| Boston Celtics     | 004       | Marcus Smart    |
| Boston Celtics     | 005       | Kemba Walker    |
| Boston Celtics     | 006       | Robert Williams |
| Boston Celtics     | 007       | Evan Fournier   |
| Brooklyn Nets      | 009       | Kyrie Irving    |
| Brooklyn Nets      | 010       | DeAndre Jordan  |
| Brooklyn Nets      | 011       | Joe Harris      |
| Brooklyn Nets      | 012       | James Harden    |
| Brooklyn Nets      | 013       | Kevin Durant    |
| Brooklyn Nets      | 014       | Blake Griffin   |
| New York Knicks    | 021       | RJ Barrett      |
| New York Knicks    | 022       | Julius Randle   |
| New York Knicks    | 024       | Taj Gibson      |
| New York Knicks    | 025       | Kevin Knox      |
| New York Knicks    | 026       | Derrick Rose    |
| New York Knicks    | 023       | Reggie Bullock  |
| Philadelphia 76ers | 015       | Joel Embiid     |
| Philadelphia 76ers | 016       | Ben Simmons     |
| Philadelphia 76ers | 017       | Tobias Harris   |
| Philadelphia 76ers | 019       | Josh Richardson |
| Philadelphia 76ers | 020       | Seth Curry      |
| Toronto Raptors    | 028       | Serge Ibaka     |
| Toronto Raptors    | 029       | OG Anunoby      |
| Toronto Raptors    | 030       | Kyle Lowry      |
| Toronto Raptors    | 031       | Pascal Siakam   |
| Toronto Raptors    | 032       | Fred VanVleet   |

잘못된 숫자를 입력하면 에러 페이지로 안내됩니다.

### major players of season

season:

[get result](#)

failed to get roster. please check season again(have to be 2019~2020)

\*에러 페이지

기능3

플레이어 id와 자신이 추가할 태그를 입력하고 북마크에 추가할 수 있습니다. 성공적으로 추가됐으면 player inserted라는 문구와 함께 업데이트된 북마크가 보여집니다.

## player bookmark

player:

mytag:

player inserted

| player id: | player name:    | player tag:          |
|------------|-----------------|----------------------|
| 001        | Jayson Tatum    | king of the boston   |
| 013        | Kevin Durant    | mid range specialist |
| 002        | JayLen Brown    | injured              |
| 006        | Robert Williams | great wingspan       |
| 021        | RJ Barrett      | Great Rookie         |

edit player tag

player id:

new tag:

## 기능 4

북마크 페이지는 북마크 추가 직후 혹은 my bookmark 버튼을 통해 들어갈 수 있습니다. 여기서 edit tag를 통해 선수의 태그를 수정할 수 있습니다. 태그 수정 후엔 초기 화면으로 돌아가며, my bookmark 버튼으로 수정된 결과를 확인할 수 있습니다.

player inserted

| player id: | player name:    | player tag:          |
|------------|-----------------|----------------------|
| 001        | Jayson Tatum    | king of the boston   |
| 013        | Kevin Durant    | mid range specialist |
| 002        | JayLen Brown    | injured              |
| 006        | Robert Williams | great wingspan       |
| 021        | RJ Barrett      | Great Rookie         |

edit player tag

player id:

new tag:

## nba athlentic division & key players

### basic stats of players in 2019-2020 Athlentic Division

### major players of season

season:

### player bookmark

player:

mytag:

my bookmark

| player id: | player name:    | player tag:          |
|------------|-----------------|----------------------|
| 001        | Jayson Tatum    | king of the boston   |
| 013        | Kevin Durant    | mid range specialist |
| 006        | Robert Williams | great wingspan       |
| 021        | RJ Barrett      | Great Rookie         |
| 002        | JayLen Brown    | healed               |

edit player tag

player id:

new tag:

Jaylen Brown의 tag가 injured->healed로 변경되었습니다.

기능 5

선수의 아이디를 입력하고 Delete from bookmark 버튼을 통해 북마크에서 선수를 제거할 수 있습니다.

## player bookmark

player:

mytag:

player deleted

| player id: | player name:    | player tag:          |
|------------|-----------------|----------------------|
| 001        | Jayson Tatum    | king of the boston   |
| 013        | Kevin Durant    | mid range specialist |
| 006        | Robert Williams | great wingspan       |
| 021        | RJ Barrett      | Great Rookie         |

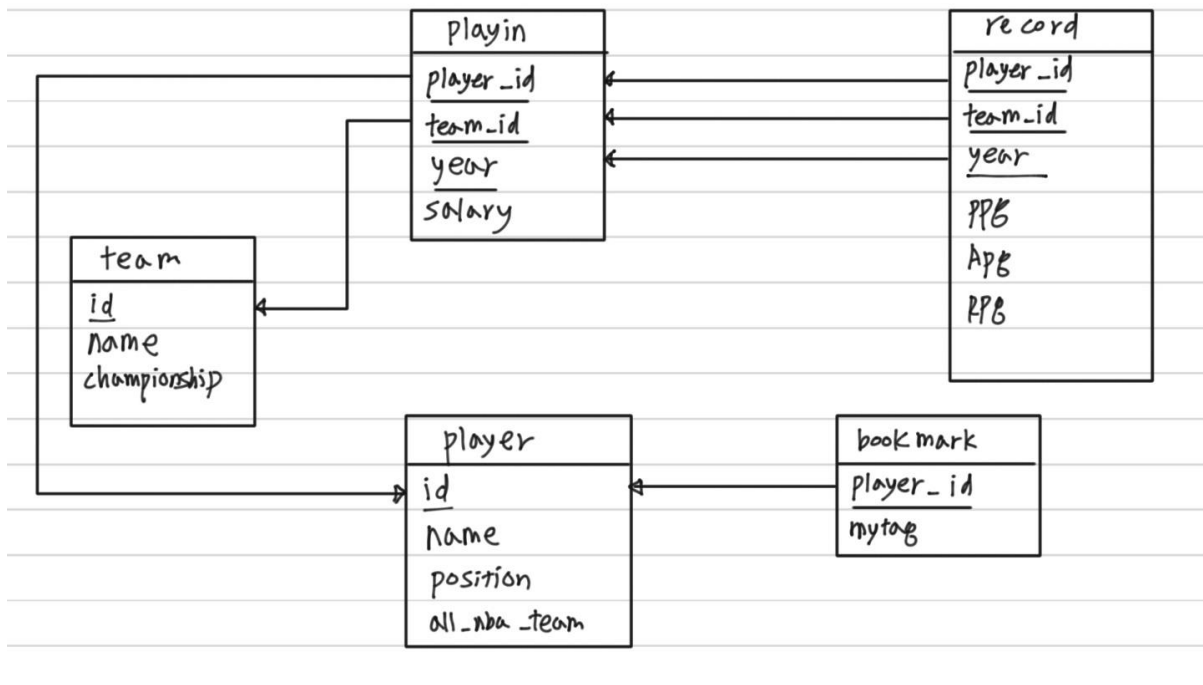
edit player tag

player id:

new tag:

북마크에서 Jaylen Brown이 사라진 것을 볼 수 있습니다.

## 2. DB Application에 사용한 schema diagram



### 3. 구현한 함수들에 대한 설명

```
@app.route('/')
def firstPage(): #라우팅하고 바로 실행하게 됨
    # Use a breakpoint in the code line below to debug your script.
    return render_template('main.html')
```

firstPage: main page를 보여줍니다.

```
@app.route('/stat', methods=['GET'])
def print_stats():
    cur.execute("select player.name, ROUND(avg(record.ppg), 1) as points_per_game, ROUND(
    result = cur.fetchall()
    return render_template('stat.html', infos=result)
```

Print\_stats: 선수들의 2019-2020시즌 주요 스탯 평균을 소수점 첫째 자리까지 보여줍니다. 이는 stat.html에 전달되어 테이블로 출력됩니다.

다음과 같은 sql 문장이 사용되었습니다.

```
select player.name, ROUND(avg(record.ppg), 1) as points_per_game, ROUND(avg(record.apg), 1) as
assists_per_game, ROUND(avg(record.rpg), 1) as rebounds_per_game from player, playin natural join
record where player.id=playin.player_id group by player.name;
```

```
def print_players():
    year = request.form["year"]
    button = request.form["send"]
    if button:
        cur.execute("SELECT T.name, P.id, P.name from team as T, player as P where P.id
        result = cur.fetchall()
        if result: return render_template('print.html', rosters=result)
    return render_template('failed.html')
```

Print\_players

year값을 입력받아 버튼이 입력되면 해당 연도에 Atlantic Division에서 플레이했던 선수들과 팀을 출력합니다. 성공 시 print.html에 결과를 전달하고 실패 시 failed페이지를 출력합니다.

다음과 같은 sql 문장이 사용되었습니다.

```
SELECT T.name, P.id, P.name from team as T, player as P where P.id in (select player_id from playin
where T.id=team_id and year={}) //year변수는 .format(int(year)) 를 통해 전달되었습니다.
```



```

@app.route('/add_myplayer', methods=['POST'])
def add_bookmark():
    playerid = request.form["player"]
    mytag = request.form["mytag"]
    send = request.form["send"]
    if send:
        message="my bookmark"
        if send == "add to bookmark":
            cur.execute("INSERT INTO bookmark VALUES('{}', '{}');".format(playerid, mytag))
            connect.commit()
            message="player inserted"
        if send == "delete from bookmark":
            if playerid:
                cur.execute("DELETE FROM bookmark WHERE player_id='{}';".format(playerid))
                connect.commit()
                message="player deleted"
            else: message="no info to delete"
        cur.execute("select player_id, name, mytag from bookmark as B, player as P where B.player_id = P.id;")
        result = cur.fetchall()
        if result: return render_template('mybookmark.html', sorf=message, bookmark=result)
    return render_template('mybookmark.html', sorf="something excepted")

```

Add\_bookmark: 플레이어 아이디와 태그를 받습니다. Add to bookmark버튼이 입력된 경우에는 INSERT INTO bookmark VALUES('{}', '{}')를 통해 해당 값을 bookmark 테이블에 삽입합니다. 변수는 .format(playerid, mytag)) 를 통해 전달됩니다.

Delete from bookmark버튼이 입력된 경우에는 player\_id 변수를 sql문장에 전달합니다. 이 때 DELETE FROM bookmark WHERE player\_id='{}' 가 실행됩니다.

입력이 처리된 후에는 mybookmark.html에 처리된 프로세스에 따라 메시지를 전달합니다. 이 때 mybookmark.html에서는 select player\_id, name, mytag from bookmark as B, player as P where B.player\_id = P.id; 의 결과를 보여줍니다. 이에 따라 insert 혹은 delete의 결과를 볼 수 있습니다. Insert 나 delete 가 아닌 mybookmark버튼을 통해 bookmark의 결과만을 요청받은 경우 message는 'my bookmark'로 처리되며, 그 외의 경우에는 insert나 delete 여부에 따라 message가 정해집니다. 전달된 message는 mybookmark.html의 상단에 출력됩니다. 버튼이 정상적으로 처리되지 않았을 경우에는 something excepted가 sorf를 통해 전달됩니다. 이 매개변수는 mybookmark.html의 body태그 내에 출력됩니다.

```

@app.route('/editTag', methods=['POST'])
def edit_tag():
    id = request.form["id"]
    newtag = request.form["newtag"]
    send = request.form["edit"]
    if send:
        cur.execute("update bookmark set mytag='{}' where player_id='{}';".format(newtag, id))
        connect.commit()
    return firstPage()

```

Edit\_tag: mybookmark.html에서 tag를 편집 시 발생하는 함수입니다. Player의 id와 갱신될 태그를

입력받아 edit tag버튼이 입력되면 update bookmark set mytag='{}' where player\_id='{};' 쿼리를 통해 bookmark테이블을 갱신합니다. 변수는 format(newtag, id)를 통해 전달됩니다.