

RDBMS homework - NBA Database

Quick start

1. clone the repository and enter the folder

```
$ git clone https://github.com/ChickenBenny/NBA-database-homework
$ cd NBA-database-homework
```

2. Use docker to build the database

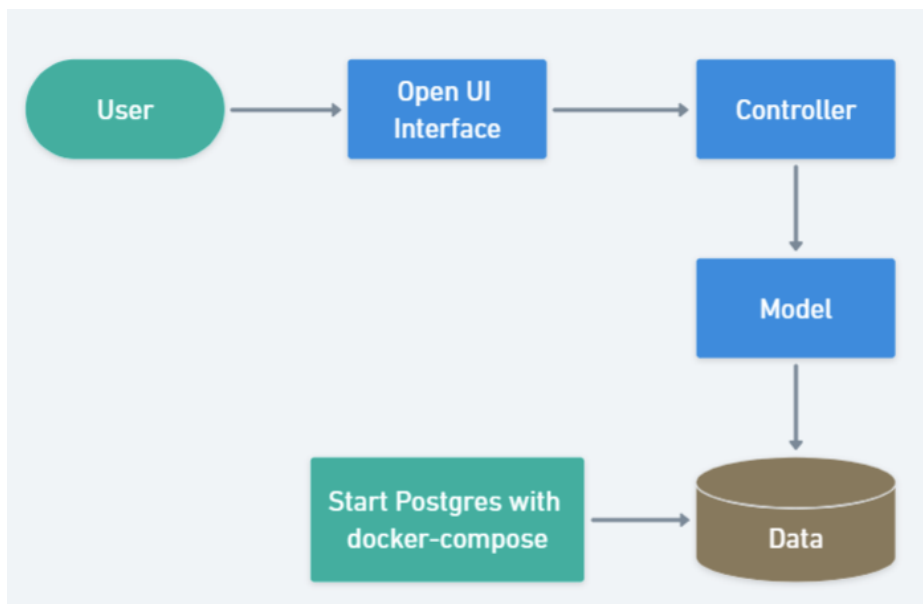
```
$ docker-compose up
```

3. Install the requirement and run the GUI app

```
$ pip install -r requirements.txt
$ python main.py
```

系統架構與環境

- 系統架構



- 系統環境

系統 : Ubuntu 20.04
容器 : Docker 20.10.17
資料庫 : PostgreSQL 13
前端 : PyQt5

- 使用說明

1. 使用 docker-compose up 架設 Postgres。

```
root@LAPTOP-6VS6J9A4:/mnt/d/project/NBA-database-Homework# docker-compose up
[+] Running 2/2
  # Network nba-database-homework_default Created
  # Container database Created
Attaching to database
database | The files belonging to this database system will be owned by user "postgres".
database | This user must also own the server process.
database |
database | The database cluster will be initialized with locale "en_US.utf8".
database | The default database encoding has accordingly been set to "UTF8".
database | The default text search configuration will be set to "english".
database |
database | Data page checksums are disabled.
database |
database | fixing permissions on existing directory /var/lib/postgresql/data ... ok
database | creating subdirectories ... ok
database | selecting dynamic shared memory implementation ... posix
database | selecting default max_connections ... 100
database | selecting default shared_buffers ... 128MB
database | selecting default time zone ... Etc/UTC
```

2. 確認 Postgres 是否成功架設。

```
root@LAPTOP-6VS6J9A4:/mnt/d/project/NBA-database-Homework# docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS          NAMES
ada318309e35   postgres:13   "docker-entrypoint.s..." About a minute ago Up About a minute 0.0.0.0:5432->5432/tcp database

root@LAPTOP-6VS6J9A4:/mnt/d/project/NBA-database-Homework# docker exec -it database sh
# psql -U postgres
psql (13.9 (Debian 13.9-1.pgdg110+1))
Type "help" for help.

postgres=# \dt
          List of relations
Schema | Name      | Type  | Owner
-----+-----+-----+-----
public | coach     | table | postgres
public | game      | table | postgres
public | nba_player | table | postgres
public | stadium   | table | postgres
public | team      | table | postgres
(5 rows)
```

3. 使用 python ./app/main.py 開啟前端 UI 頁面。

Simple DBMS User Interface

Button: information

SELECT-FROM-WHERE: 列出主隊球場在 California 的球隊

Show Clear

Query

Type some POSTGRES query up here ↑

Show Clear

Clear ALL Clear

4. 下拉選擇你要的 query，然後點選 show 便能觸發該 query。(點選 Clear 刪除頁面上的結果)

Simple DBMS User Interface

Button: information

SELECT-FROM-WHERE: 列出主隊球場在 California 的球隊

Show Clear

Query

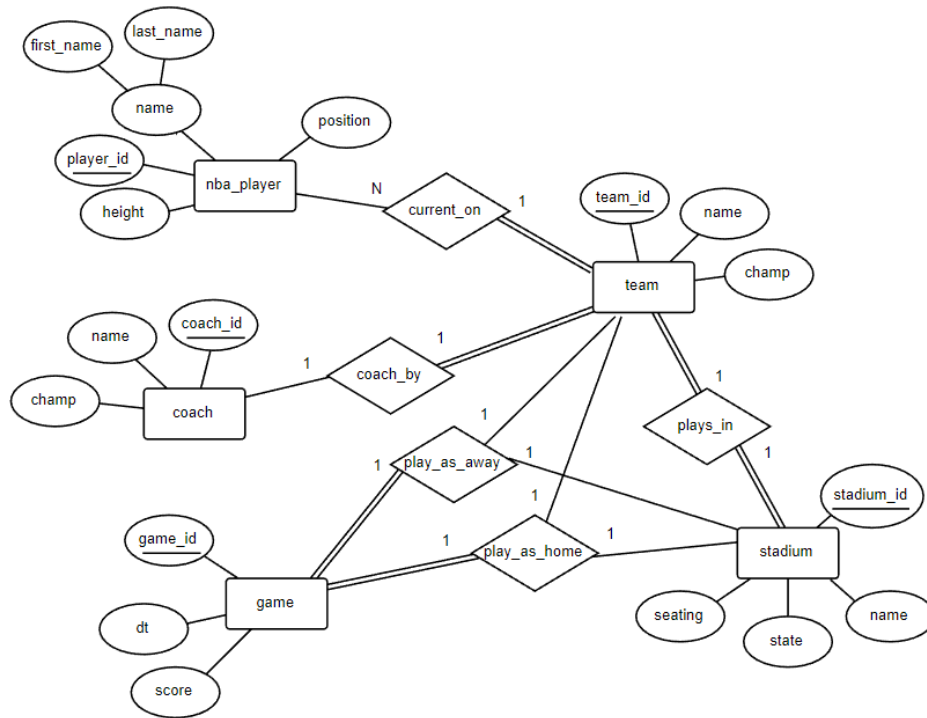
SELECT player_ssn, first_name, last_name FROM player AS P, team AS T WHERE T.team_name='New York Yankees' AND P.current_team=T.team_id;

Type some POSTGRES query up here ↑

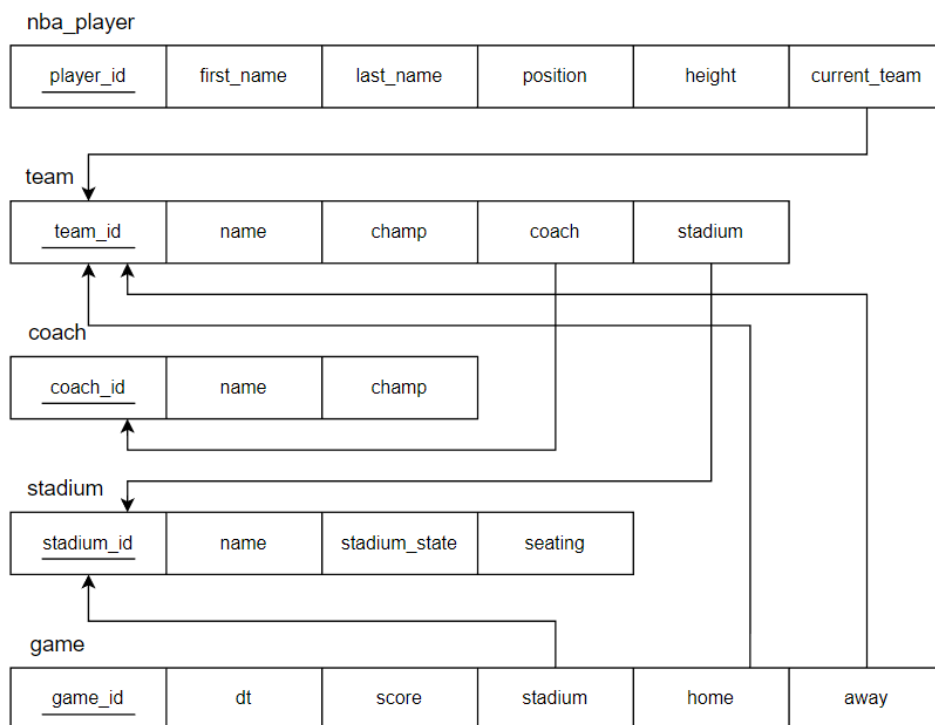
Show Clear

full_name_of_team	stadium_name	stadium_state
Golden State Warriors	Chase Center	California
Los Angeles Clippers	Crypto.com Arena	California
Los Angeles Lakers	Intuit Dome	California
Sacramento Kings	Golden 1 Center	California

ER Diagrams



Relation schema



資料庫基本需求

Entity (5個)

- **nba_player** (共43筆資料) :
 1. player_id (**primary key**) : 球員的 id , 為 key attribute 。

2. first_name : 球員的 first_name 。
 3. last_name : 球員的 last_name 。
 4. position : 球員在球隊中的打的位置 。
 5. height : 球員的身高 。
 6. current_team (**foreign key**) : 目前球員所屬的球隊 。
- **team** (共30筆資料) :
 1. team_id (**primary key**) : 球隊的 id 。
 2. name : 球隊的名稱 。
 3. team_state : 球隊所屬的州 。
 4. coach (**foreign key**) : 球隊的總教練 id 。
 5. stadium (**foreign key**) : 球隊得主隊球場 id 。
 - **coach** (共30筆資料) :
 1. coach_id (**primary key**) : 教練的 id 。
 2. name : 教練的名稱 。
 3. champ : 奪冠總數量 。
 - **stadium** (共30筆資料) :
 1. stadium_id (**primary key**) : 球場的 id 。
 2. name : 球場的名稱 。
 3. stadium_state : 球場所在的州 。
 4. seating : 球場的觀眾席次 。
 - **game** (共20筆資料) :
 1. game_id (**primary key**) : 比賽的 id 。
 2. dt : 比賽的日期 。
 3. score : 比賽結束的比分 。
 4. stadium (**foreign key**) : 比賽的球場 id 。
 5. home (**foreign key**) : 比賽的主隊 id 。
 6. away (**foreign key**) : 比賽的客隊 id 。

Relationship (需包含二元及三元)

- 二元關係
 1. current_team => nba_player and team
每一支球隊有 N 個球員目前為他效力，但並不是每個球隊都有所屬球隊，有些球員可能是自由球員 。
 2. coach_by => coach and team
每一支球隊必有一個總教練執教，但並不是每個總教

練都有球隊可以執教。

3. plays_in => team and stadium

每一支球隊都必有一個球隊主場。

- 三元關係

1. play_as_home => game, stadium and team

每一場比賽都必定有主隊球隊、比賽的球場。

2. play_as_away => game, stadium and team

每一場比賽都必定有客隊球隊、比賽的球場。