Software Distribuït

Pràctica 2
2023

Estructura de fitxers py:

main.py: Estaran els Endpoints.

database.py: Definició d'on es troba la BD

models.py: Definició del model ORM de la BD (taules i mappings)

schemas.py: Definició dels esquemes/models de PyDantic, per a la validació i transformació de dades des de i cap als endpoints

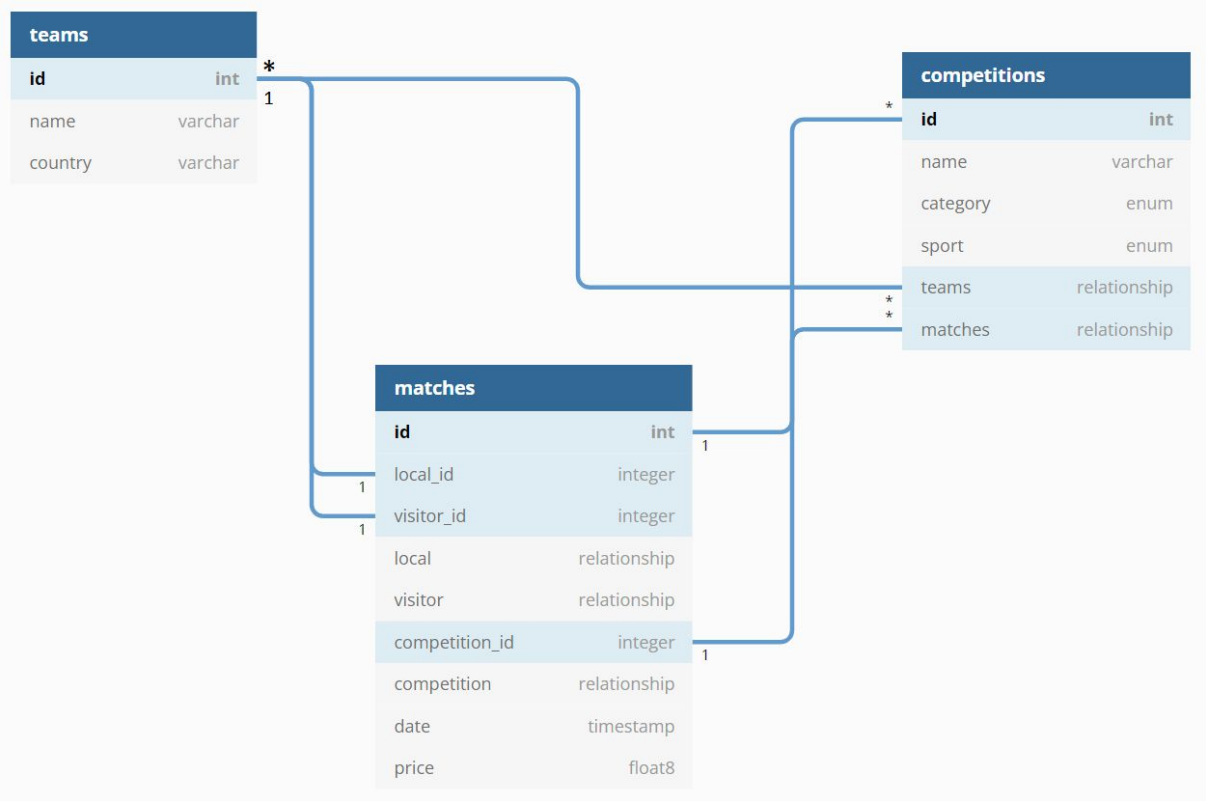
repository.py: Operacions de consulta i manipulació a la BD usant SQLAlchemy

test_main.py: Tests de l'aplicació.

OO/ ORM /ER

- Per definir un domini normalment ho fem Orientat a Objectes seguint un diagrama de classes
- Per definir una BD relacional normalment ho fem amb un diagrama ER, seguint les normes de l' àlgebra Relaciona.
- ORM ens permet fer un "Mapping" entre les classes i l'Orientat a Objectes, guardant l'estat dels objectes directament a bases de dades, sense usar SQL.

Diagrama de Classes / ER



SQL Alchemy

Per fer el Mapping entre Classes de Python i el Model de Bases de Dades (farem servir SQLlite que només és un fitxer) usarem SQL Alchemy.

A models.py hem de definir aquest Mapping, en que trobem tant taules, com camps, com claus primàries i foranes que representen els nostres objectes, com les relacions entre objectes que ens permeten usar els objectes com a objectes normals de python

Les relacions s'han de posar tant en format clau primària clau forana com en format de col·leccions d'ED (llistes).

SQL Alchemy ens ajuda a fer aquest mapping "màgic" gràcies a la seva "Alquímia".

Schemas

Podem fer servir els Schemas de Pydantic per passar els objectes directament dels endpoints als models de SQLAlchemy.

Definirem schemas base amb la informació comú tant per les operacions d'escriptura com de lectura de l'api. Els Schema Create seran per les operacions de post, que no necessiten ID per exemple, i els schema amb el nom igual que el model per les opereacions get.

Podeu ampliar aquests esquemes segons les necessitats dels endpoints.

Main

A main només podrem posar els endpoints,

Generarem una sessió de BD per cada petició.

Cridarem les funcions de repository que interactuen amb la base de daes.

Com a paràmetres i valors de retorns als endpoints, passarem els Schemas definits.

Repository

Aquí farem les crides a la BD tant per selects com per les altres operacions

Passarem per paràmetres Schemas de pydantic i treballarem amb els objectes de model de l'ORM

Per fer canvis a les classes del model ho fem de forma normal, quan vulguem que aquests canvis es desin a BD farem servir la sessió de DB, per exemple per afegir un nou registre fem add i a continuació commit. Amb refresh tornem a recuperar el valor de l'objecte de la BD, un cop fet el commit, així per exemple l'ID quedarà actualitzat.

Com que els Schemas i el Models de les operacions get coincideixen es pot retornar un Model resultant del repository i enviar-lo com a resultat de l'endpoint