

به نام یکتا



طراحی پایگاه داده‌ها

تمرین ۲

استاد

مهدی آخی

تهیه و تدوین

تیم دستیاران درس - بخش تمارین

فروردین ۱۴۰۳

فهرست

2.....	ضوابط تمرین
2.....	سیاست‌های جزئی تمرین ۲
3.....	سوال اول: پیتزا فروشی
4.....	سوال دوم: قطعات
5.....	سوال سوم: توضیح جستارها
5.....	بخش اول
5.....	بخش دوم
6.....	سوال چهارم: بازی کامپیوتری
6.....	فرایندهای کسب‌وکار
6.....	Relational Model
7.....	جستارها

ضوابط تمرین

- پرسش و پاسخ

- برای هر تمرین یک پست در کوئرا ایجاد خواهد شد که می‌توانید سوالات و ابهامات احتمالی خود را در زیر همان پست بپرسید.

- سیاست تاخیر

- تاخیر به صورت ساعتی محاسبه می‌شود، هر تمرین را تا حداکثر ۲ روز پس از ددلاین می‌توانید ارسال کنید.
- در مجموع هر فرد 5 روز تاخیر مجاز دارد.
- به ازای هر ساعت تاخیر غیر مجاز ۲ درصد از نمره‌ی آن تمرین کم خواهد شد.

- سیاست تقلب

- در صورت مشاهده هرگونه مشابهت نامتعارف، بار اول نمره‌ی کل تمرین صفر شده و بار دوم ادامه‌ی درس میسر نخواهد بود. (برای هر دو طرف درگیر)
- مشورت و ایده گرفتن از یکدیگر در صورتی که راه حل دیگری را مشاهده نکنید ایرادی ندارد اما باید در پاسختان صراحتاً ذکر شود.

سیاست‌های جزئی تمرین ۲

- ددلاین ارسال پاسخ: 19 فروردین
- نمره تمرین: 1.5 نمره
- این تمرین مربوط به اسلایدهای Relational Algebra درس می‌باشد.

سوال اول: پیتزا فروشی

پایگاه داده‌ای از مجموعه پیتزافروشی‌های شهر و مشتریان ثابت آن‌ها و غذاهایی که سرو می‌کنند داریم که شامل روابط زیر است:

- Person(ID, name, age, gender)
- Frequents(personID, pizzeria)
- Eats(personID, pizza)
- Serves(pizzeria, pizza, price)

در این پایگاه داده، رابطه‌ی Person شامل افراد مختلف است و رابطه Frequents به ما می‌گوید که هر فرد معمولاً پیتزای خود را در چه پیتزافروشی(هایی) می‌خورد (مشتري ثابت آنجاست) و جدول Eats هر فرد و پیتزا(ها)ی که می‌خورد را به ما می‌دهد و در نهایت جدول Serves به ما می‌گوید هر پیتزافروشی چه پیتزایی را با چه قیمتی عرضه می‌کند.

حال با توجه به توضیحات بالا جبر رابطه‌ای مناسب برای جستار¹های زیر را بیابید؟

1. همه پیتزا فروشی‌هایی را پیدا کنید که حداقل یک مشتری بالای ۸۰ سال دارند.
2. همه پیتزا فروشی‌هایی را بیابید که حداقل یک پیتزا را سرو می‌کنند که قیمتی زیر ۲۵۰ هزار تومن دارد و فردی به نام عموحسن آن را می‌خورد.
3. تمام پیتزا فروشی‌هایی را بیابید که فقط یا مشتری‌های آقا دارند و یا فقط مشتری‌های خانم.
4. افرادی را بیابید که مشتری همه پیتزافروشی‌هایی هستند که حداقل یکی از پیتزاهایی که می‌خورند را سرو می‌کنند.
5. پیتزا فروشی(هایی) را بیابید که ارزانترین پیرونی شهر را دارند! (در صورتی که چند پیتزا فروشی ارزانترین پیتزا را می‌دادند باید همه آنها را برگردانید).

¹ Query

سوال دوم: قطعات

شمای رابطه‌ی زیر نشان دهنده کارخانه‌های تولید کننده قطعات در یک کشور، قطعات تولید شده، و قطعات تولید شده توسط هر کارخانه است.

- Factory(fID, fName, status, city, CEO, ...)
- Product(pID, color, price, ...)
- FP(sID, pID, quantity)

حال با توجه به توضیحات بالا جبر رابطه ای مناسب برای جستارهای زیر را بیابید.

الف) شماره کارخانه‌های شهر تهران که وضعیت آنها "Open" باشد.

ب) نام کارخانه‌های تولید کننده ی قطعه با شناسه ی "40678".

ج) نام کارخانه‌هایی که حداقل یک قطعه قرمز رنگ تولید می‌کنند.

د) نام کارخانه‌هایی که حداقل یک قطعه تهیه شده توسط کارخانه ی "Iran Sim" را تولید کرده باشند.

سوال سوم: توضیح جستارها

بخش اول

برای هر کدام از عبارات جبری زیر بگویید که چه چیزی می‌گویند؛ همچنین توضیح دهید تفاوت مورد شماره 1 و 2 در چیست؟

1. $\Pi_{sname}(\sigma_{color='red'}(Part) \bowtie \sigma_{cost < 100}(Catalog)) \bowtie Supplier$
2. $\Pi_{sname}(\Pi_{sid}(\sigma_{color='red'}(Part) \bowtie \sigma_{cost < 100}(Catalog)) \bowtie Supplier)$
3. $\Pi_{sname}(\sigma_{color='red'}(Part) \bowtie \sigma_{cost < 100}(Catalog) \bowtie Supplier) \cap \Pi_{sname}(\sigma_{color='green'}(Part) \bowtie \sigma_{cost < 100}(Catalog) \bowtie Supplier)$

بخش دوم

حال روابط زیر که مستتر در یک پایگاه داده می‌باشند را در نظر گرفته و مجدداً مانند سوال قبل پاسخ بدهید؛ s و b و r به ترتیب خلاصه‌ی Sailors و Boats و Reserves می‌باشند. (در مورد دوم و سوم p را در نظر بگیرید که نامگذاری کل جدول را عوض می‌کند).

s (sid, sname, rating, age)

b (bid, bname, color)

r (sid, bid, date)

1. $\Pi_{s.sname}[(\Pi_{sid,bid}(r) \div \Pi_{bid}(b))] \bowtie s$
2. $\Pi_{sid}(s) - \Pi_{s2.sid}(\sigma_{s2.rating \leq s.rating}[\rho_{s2}(s) \times \sigma_{sname='Bob'}(s)])$
3. $\Pi_{sname,age}[(\Pi_{sid}(s) - \Pi_{s2.sid}(\sigma_{s2.age < s.age}(\rho_{s2}(s) \times s))] \bowtie s$

سوال چهارم: بازی کامپیوتری

فرایندهای کسب و کار

برای یک بازی کامپیوتری Multiplayer پایگاه داده‌ای طراحی کرده‌ایم که اطلاعات کاربران، شخصیت‌ها، آیتم‌ها و اطلاعات دیگر بازی در آن ذخیره شده است. اطلاعات کاربران جدول Users شامل Username, Registration Date و UserID می‌باشد. هر User در بازی در واقع یک Player است که Stat های آن شامل Health یا میزان جان بازیکن ، XP یا امتیاز تجربه و Level یا سطح بازیکن می‌باشد.

هر بازیکن می‌تواند چندین شخصیت برای خود بسازد. همینطور هر بازیکن یک Inventory دارد که در آن آیتم‌های مورد نیازش را قرار داده و در صورت نیاز آن‌ها را استفاده یا حمل می‌کند. اطلاعات آیتم‌ها در جدول Items و اطلاعات شخصیت‌ها در جدول Characters ذخیره شده اند. از قابلیت‌های بازی می‌توان به معامله با بازیکنان دیگر اشاره کرد. هر بازیکن می‌تواند آیتم‌هایی که در اختیار دارد را بفروشد یا آیتم‌هایی را از بازیکنان دیگر بخرد. اطلاعات این معامله‌ها در جدول Transactions ذخیره می‌شود. مشخص است که هر دو بازیکن می‌توانند به تعداد دلخواه با یکدیگر معامله کنند. در نهایت نیز قابل ذکر است که بازیکنان می‌توانند به یکدیگر درخواست دوستی بفرستند. فرض کنید در درخواست‌های دوستی Accept شده Redundancy وجود ندارد.

Relational Model

Users: (UserID (Primary Key), Username, Email, Password, Registration Date, Last Login Date, Avatar/Image)

PlayerStats: (PlayerID (Primary Key, Foreign Key referencing Users.UserID), Level, XP, Health, Currency)

Characters: (CharacterID (Primary Key), PlayerID (Foreign Key referencing PlayerStats.PlayerID), Name, Class/Type, Attribute (strength, agility, intelligence))

Inventory: (InventoryID (Primary Key), PlayerID (Foreign Key referencing PlayerStats.PlayerID, Unique), Level)

Items: (ItemID (Primary Key), Name, Type (weapon, armor, consumable), Description)

InventoryHasItem((InventoryID (Primary Key, Foreign Key referencing Inventory.InventoryID), ItemID (Primary Key, Foreign Key referencing Items.ItemID) Quantity, Equipped (Boolean))

Transactions: (TransactionID (Primary Key), BuyerID (Foreign Key referencing PlayerStats.PlayerID), SellerID (Foreign Key referencing PlayerStats.PlayerID), ItemID (Foreign Key referencing Items.ItemID), Quantity, Price, Transaction Date)

Friends: (UserID1 (Foreign Key referencing Users.UserID), UserID2 (Foreign Key referencing Users.UserID), (Primary Key = (UserID1, UserID2)), Status (pending, accepted, declined))

دقت کنید که هر Inventory مربوط به یک Player بوده و نمی تواند بدون صاحب باشد. همینطور هر Player تنها یک Inventory دارد. جدول InventoryHasItem بدین منظور است که هر Item می تواند در چند Inventory باشد و همینطور هر Inventory می تواند انواع مختلف Item را داشته باشد.

جستارها

۱. حال با توجه به توضیحات داده شده با استفاده از عملیات های جبر رابطه ای ، جستارهایی برای موارد زیر طراحی کنید.

الف. لیست نام، نوع و تعداد آیتم های Equip شده ی تمام بازیکنانی که بعد از سال 2023 ثبت نام کرده اند.

ب. لیست نام بازیکنانی که همه آیتم‌های از نوع Weapon را حداقل یک بار خریداری کرده اند (راهنمایی: در مورد عملگر تقسیم در جبر رابطه ای مطالعه کنید. هر چند که بدون استفاده از تقسیم می‌توان به سوال پاسخ داد)

ج. لیست شناسه معاملاتی که با قیمت بیشتر از 100 دلار انجام شده و دو طرف معامله با یکدیگر دوست بوده اند.

۲. نتیجه عبارت جبر رابطه ای زیر را توضیح دهید.

$$\begin{aligned} & \Pi_{Characters.Name} (\sigma_{Characters.Attribute = 'strength'} (Characters \\ & \bowtie_{Characters.PlayerID = PlayerStats.PlayerID} (\sigma_{PlayerStats.XP \geq 250} (\\ & \Pi_{PlayerStats.PlayerID} (PlayerStats) - \Pi_{PlayerStats.PlayerID} (\\ & Transactions \bowtie_{Transactions.SellerID = PlayerStats.PlayerID} PlayerStats)))))) \end{aligned}$$

۳. کدام یک از عبارات زیر هزینه اجرای کمتری دارند؟ با دلیل توضیح دهید.

A.

$$\sigma_{Characters.PlayerID = Inventory.PlayerID \wedge Characters.Type = 'Healer' \wedge Inventory.Level \geq 2} (Characters \times Inventory)$$

B.

$$\sigma_{Characters.PlayerID = Inventory.PlayerID} ((\sigma_{Characters.Type = 'Healer'} (Character) \times \sigma_{Inventory.Level \geq 2} (Inventory))$$