

به نام خدا
تکلیف سوم درس پایگاه داده ها ۱
ترم دوم ۹۷-۹۸

توجه مهم: این فایل را یک بار دیگر در لحظه آخر و هنگام ارسال بخوانید. عدم رعایت شرایط ذکر شده در نام گذاری یا ارسال فایل‌های پراکنده یا هر چیزی غیر از یک فایل PDF، یک فایل SQL و یک فایل Tar در قالب یک فایل فشرده zip باعث کاهش تمام یا بخشی از نمره شما می شود.

فرمت ارسال: کل پاسخ ها را در قالب یک فایل Zip روی سامانه قرار دهید. پاسخ برخی از سوالات باید فقط شامل قسمت هایی باشد که تحت عنوان «خروجی» در زیر هر سوال مشخص شده است؛ برای باقی سوالات توضیحات لازم را در پاسخنامه متنی خود وارد کنید. نام فایل با شماره دانشجویی شروع شود و سپس فامیل (انگلیسی). مثال: 9511113-Irani.zip

نحوه ارسال روی سامانه: وارد سامانه lms.iut.ac.ir شوید و فایل پاسخنامه خود را در قسمت تکلیف اول قرار دهید. ارسال مستقیم با ایمیل به هیچ وجه پذیرفته نمی شود. لطفاً چند روز قبل از پایان مهلت، از عدم وجود مشکل در قسمت آپلود تکالیف مربوط به خودتان روی سامانه مطمئن شوید و اگر مشکلی هست آن را رفع کنید.

توجه: برای انجام این تمرین باید PostgreSQL را از روی لینک‌هایی که در کانال تلگرام ارسال شده بود نصب نمایید. در بین گزینه‌های نصب یکی از بسته های pgAdmin3 LTS یا pgAdmin 4 را نیز انتخاب نمایید.

مهلت ارسال: پنجشنبه ۱۵ فروردین، ساعت ۱۵:۲۳

بخش اول

۱- درمورد Transaction ها تحقیق کنید و به سوالات زیر پاسخ دهید؛

أ- خواص ACID را برای Transaction ها نام برده و مختصراً توضیح دهید.

ب- گفته میشود که اجرای همزمان Transaction ها زمانی که قرار باشد داده ها از یک دیسک با سرعت پایین دریافت شوند یا در هنگام طولانی بودن Transaction ها، بسیار مهم تر از زمانی است که داده‌ها در حافظه اصلی هستند یا Transaction ها بسیار کوتاه هستند. علت را توضیح دهید.

ج- تفاوت بین دو عبارت Serial و Serializable را برای زمانبندی اجرای Transaction ها در یک DBMS توضیح دهید.

د- در دو Transaction زیر فرض کنید حالت اولیه $A=0$ و $B=0$ برقرار است و شرط Consistency برای پایگاه داده نیز $A = 0 \vee B = 0$ باشد؛

| | |
|---------------------------|---------------------------|
| T_1 : read(A); | T_2 : read(B); |
| read(B); | read(A); |
| if A = 0 then B := B + 1; | if B = 0 then A := A + 1; |
| write(B). | write(A). |

آیا اجرای Serial این دو Transaction شرط Consistency را در پایگاه داده حفظ خواهد کرد؟ توضیح دهید.

۲- چهار تفاوت Function و Stored Procedure را در SQL بیان کنید.

۳- فرض کنید می خواهیم یک سامانه نرم افزاری برای فروش بلیط هواپیما طراحی کنیم؛ این سامانه دارای پایگاه داده ای با جداول زیر است:

ا- **فرودگاه** (کد فرودگاه، نام فرودگاه، کد شهر، طول جغرافیایی، عرض جغرافیایی)

ب- **شهر** (کد شهر، نام شهر، کد استان)

ج- **استان** (کد استان، نام استان، کد کشور)

د- **کشور** (کد کشور، نام کشور)

ه- **ایرلاین** (کد ایرلاین، نام ایرلاین)

و- **برنامه پروازی** (شناسه پرواز، تاریخ پرواز، کد فرودگاه مبدأ، کد فرودگاه مقصد، ساعت پرواز، کد ایرلاین، شماره پرواز، آخرین قیمت، آخرین ظرفیت)

ز- **قیمت - ظرفیت** (شناسه پرواز، تاریخ و ساعت اعلام قیمت - ظرفیت، قیمت، ظرفیت، کانال اعلام اطلاعات)

ح- **مسافر** (کد ملی، نام، نام خانوادگی، سن، جنسیت)

ط- **بلیط** (شماره بلیط، شناسه پرواز، کد ملی، شماره صندلی، قیمت کل)

در این پایگاه داده شرایط زیر باید رعایت شود؛

- کد مسافر، کد شهر، استان و کشور همگی عدد صحیح هستند
- کد فرودگاه ها همواره ۳ کاراکتر است. برای مثال IFN مربوط به فرودگاه شهید بهشتی اصفهان است.
- کد ایرلاین همواره دو کاراکتر است. مانند IR برای ایران ایر یا EP برای هواپیمایی آسمان و ...
- شماره پرواز الزاماً منحصر به فرد نیست. لذا برای هر پرواز یک شناسه پرواز یکتا در نظر گرفته می شود. لذا این امر طبیعی است یک پرواز تنها یکبار انجام می شود!
- قیمت و ظرفیت در برنامه پروازی شناور است و در هر لحظه می تواند عوض شود
- تغییر قیمت و ظرفیت بر اساس اطلاعات دریافتی از کانال های اعلام اطلاعات خواهد بود که تنها شامل وب سرویس، چک کردن سیستم های رزرواسیون ایرلاین ها و تماس تلفنی است. این تغییر منجر به درج یک رکورد در جدول قیمت - ظرفیت خواهد بود.

حال به سوالات زیر طبق خروجی خواسته شده پاسخ دهید.

ا- جداول فوق را با اسامی دلخواه ساخته و ضمن رعایت محدودیت های خواسته شده و محدودیت های منطقی (کلید خارجی و...)، رکورد هایی در آن ها ثبت کنید. برای ثبت استاندارد داده ها می توانید از این لینک ها نیز استفاده کنید؛

• اطلاعات فرودگاه ها: <https://openflights.org/html/apsearch>

• اطلاعات ایرلاین ها: <http://openflight.org/html/alsearch>

ب- یک view بنویسد که تعداد پرواز های بین استان ها را به تفکیک تاریخ مشخص نشان دهد که شامل نام استان مبدأ، نام استان مقصد، تاریخ، تعداد پرواز باشد.

ج- یک view از آخرین قیمت پرواز ها بر اساس جدول قیمت - ظرفیت ایجاد کنید. سپس بر اساس این view، یک view دیگر ایجاد کنید که قیمت های مغایر (رکورد هایی که دارای اختلاف قیمت هستند به همراه مقدار این اختلاف) بین جداول

برنامه پروازی و قیمت-ظرفیت را نمایش دهد. به نظر شما چرا از یک توسعه دهنده خواسته می شود تا اطلاعات این چینی را در دو جدول نگه دارد که باعث چنین مغایرتهای احتمالی شود؟

د- در یک view لیست کد و نام فرودگاه هایی که به عنوان مبدأ پرواز در سامانه ثبت شده اند را نمایش دهید با توجه به اینکه اگر برای یک شهر بیش از یک فرودگاه در لیست باشد، باید کد شهر و نام شهر نیز در رکورد های مذکور نمایش داده شوند. این امر نباید برای شهرهایی که تنها یک فرودگاه دارند انجام شود (برای مثال برای اصفهان که تنها یک فرودگاه دارد فقط کد و نام فرودگاه باید نمایش داده شود اما برای شهری نظیر تهران که دو فرودگاه دارد باید کد و نام شهر نیز علاوه بر کد و نام فرودگاه نمایش داده شود)

ه- نحوه ساخت Trigger، Stored Procedure و Function را در PostgreSQL مطالعه کنید. سپس یک Trigger بنویسید که هر زمان تغییری در اطلاعات برنامه های پروازی نظیر ظرفیت یا قیمت پرواز ایجاد شد، نسخه قدیمی اطلاعات را در یک جدول دیگر به همراه تاریخ و زمان تغییر توسط یک Stored Procedure به ثبت برساند.

و- یک Trigger بنویسید که به محض درج رکورد در جدول قیمت-ظرفیت، آخرین قیمت و ظرفیت یک پرواز را با استفاده از یک Stored Procedure، در جدول برنامه پروازی به روزرسانی کند.

ز- برخی توسعه دهندگان، به جای استفاده از Trigger، در یک Transaction کارهایی که میخواهند در Trigger انجام دهند را انجام می دهند و سپس عملی را که Trigger به آن واکنش نشان می دهد را اجرا می کنند. ضمن انجام روند بیان شده برای سوال ۳-الف، بیان کنید این امر چه ایرادی می تواند داشته باشد؟

ح- یک Function بنویسید که مشخص کند یک پرواز انجام شده است یا خیر (برای انجام شدن پرواز کفایت از تاریخ و ساعت پرواز گذشته باشد). سپس با استفاده از این Function، یک Procedure طراحی کنید که به محض اجرا شدن، پرواز های انجام شده را از سیستم حذف و حداقل اطلاعات مورد نیاز آن ها را به جدولی با عنوان تاریخچه سفر ها منتقل کند.

سپس راهکاری برای اجرای دوره ای این Procedure پیشنهاد دهید تا پرواز ها در سیستم همواره بروز بماند.

ط- برای پایگاه داده ای که در این سوال طراحی کرده اید، با استفاده از دستورات ایجاد نقش و اعطای مجوز در PostgreSQL یک نقش به عنوان اپراتور فروش بلیط تعریف کرده، سپس مجوز های لازم را جهت انجام وظایف به این نقش اعطا کنید. (توجه داشته باشید که در طراحی نقش ها و مجوز ها در یک پایگاه داده همواره اصل بر این است که برای دادن مجوز حداقل دسترسی های مورد نیاز را در نظر بگیریم. چرا؟)

خروجی: برای پاسخ های سوال ۳ ضمن وارد کردن پاسخ های تشریحی و Screenshot ها در پاسخنامه کتبی (با وضوح کافی، لطفاً ابتدا تصویر را در بالاترین وضوح بگیرید و پس از Paste لطفاً سایز تصویر را تغییر ندهید) و کد های SQL مورد نیاز در فایل HW3.SQL، از طریق لینک زیر، نحوه Backup گرفتن از پایگاه داده را در PostgreSQL مطالعه نموده و از پایگاه داده خود یک پشتیبان با فرمت Tar ایجاد نمایید. به این ترتیب شما باید به همراه پاسخنامه و فایل HW3.SQL خود یک فایل Tar با عنوان شماره دانشجویی و نام خود به صورت 9511113-Irani-Q3.tar نیز تحویل دهید.

https://www.pgadmin.org/docs/pgadmin4/dev/backup_dialog.html

۴- دستورات مربوط به Recursive View را در PostgreSQL مطالعه کنید و یک Recursive View بنویسید که اعداد سری فیبوناچی کمتر از ۶۰ را در هر رکورد نمایش دهد.

خروجی: کد Recursive View مذکور در فایل HW3.sql و تصویر خروجی کد در پاسخنامه کتبی

بخش دوم

برای حل سوال ۵، فایل dvdrental.zip را که در تکالیف قبل ارائه شد مورد استفاده قرار دهید
کد های SQL خود را درون فایل HW3.sql در قسمت مشخص شده برای سوال که به همراه تکلیف در سامانه بارگذاری شده است
وارد کنید و تنها برای قسمت الف این سوال screenshot واضح و خوانا از خروجی پرس و جوی خود را در پاسخنامه متنی خود کپی
کنید. پاسخ به این سوالات به هر نحو دیگری منجر به از دست دادن نمره خواهد بود.

۵- یکی از امکاناتی که PostgreSQL در اختیار توسعه دهنده ها قرار می دهد، دستورات EXPLAIN و EXPLAIN ANALYZE است. در مورد کاربرد این امکان و نحوه استفاده از آن تحقیق کنید و به سوالات زیر پاسخ دهید؛
أ- پرس و جوی زیر را عیناً کپی و روی دیتابیس dvdrental اجرا نموده و خروجی آن را تشریح کنید.

EXPLAIN ANALYZE

SELECT * FROM payment WHERE staff_id = 2

ب- دستور زیر را یک بار عیناً و بار دیگر با استفاده از Nested Query بدون استفاده از عبارت WITH باز نویسی و سپس توسط EXPLAIN ANALYZE روی dvdrental اجرا کنید. خروجی ها را از نظر Query Plan و میانگین زمان اجرا بررسی کنید. چه نتیجه ای می گیرید؟

WITH italian_customers AS (

SELECT cu.customer_id cid, cu.first_name, Cu.last_name ,ci.city , co.country

FROM customer cu

INNER JOIN address USING (address_id)

INNER JOIN city ci USING (city_id)

INNER JOIN country co ON co.country_id = ci.country_id

WHERE co.country = 'Italy'

)

SELECT st.staff_id, ic.first_name, ic.last_name

FROM staff st

INNER JOIN rental rt USING (staff_id)

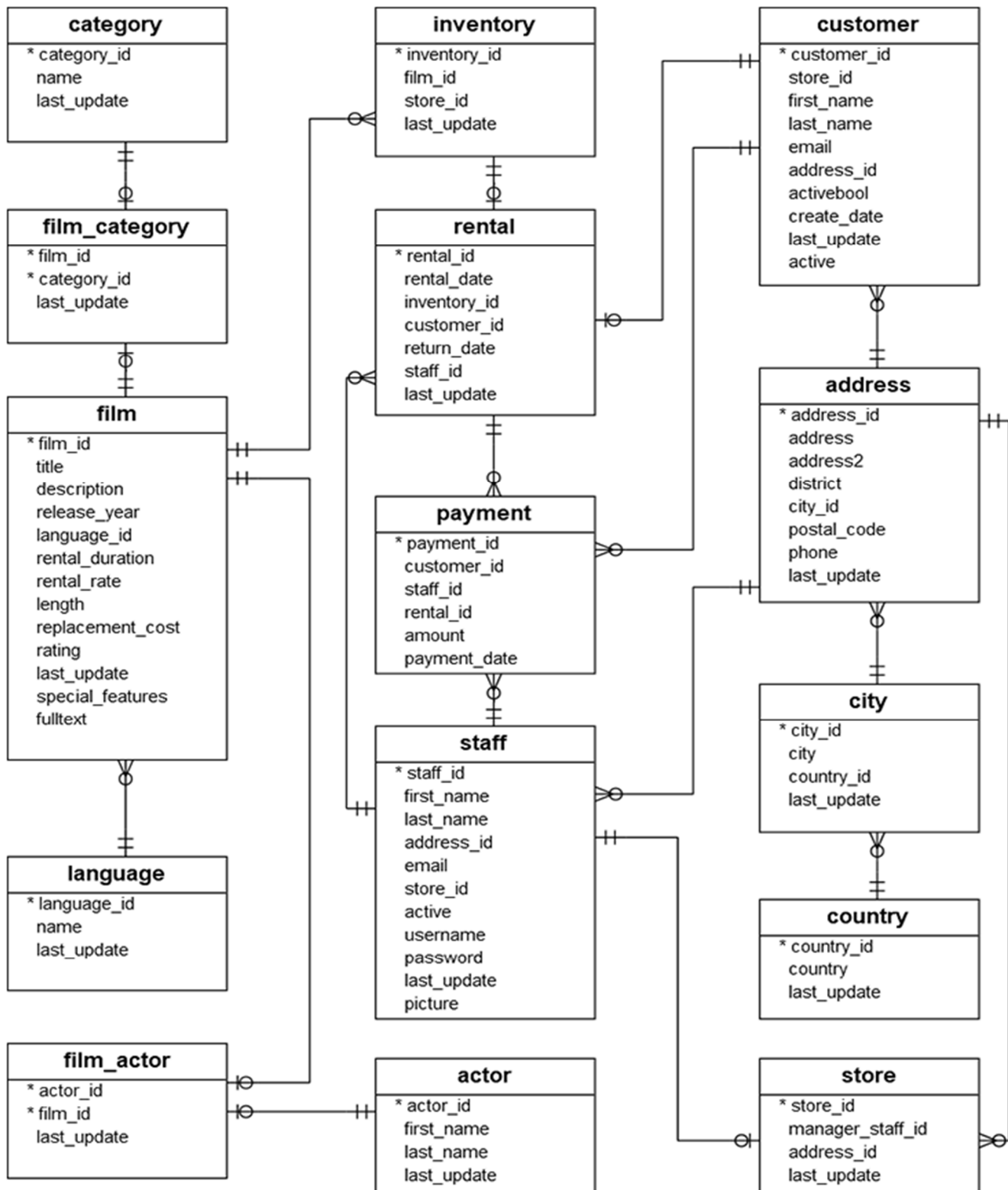
INNER JOIN italian_customers ic ON ic.cid = rt.customer_id;

ج- با توجه به Query plan در خروجی سوال (۵- ب) عملگر Hash به چه منظور توسط DBMS مورد استفاده قرار

میگیرد؟

خروجی ۵: کد SQL پرس و جوها را در قسمت Q5-b در فایل HW3.sql اضافه کنید. تصویر خروجی کد قسمت های الف و ب را

در پاسخنامه متنی خود وارد کنید.



شکل ۱ - نمودار ER پایگاه داده dvdrental