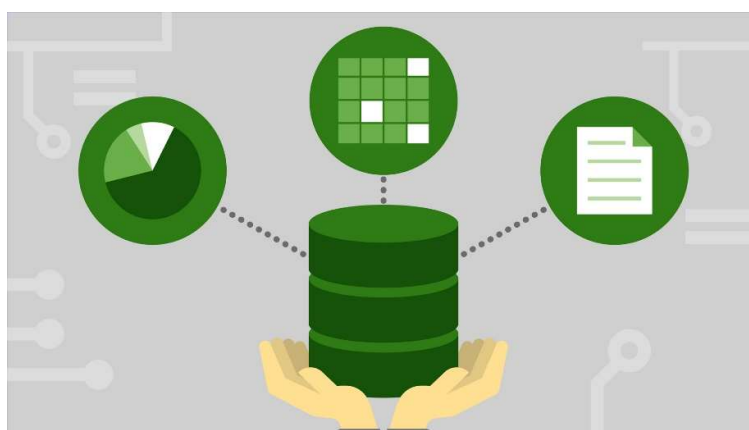


به نام خدا



دانشگاه تهران
پردیس دانشکده‌های فنی
دانشکده برق و کامپیوتر



آزمایشگاه پایگاه داده

دستورکار شماره ۴

مهلت تحویل :

۱۴۰۱/۰۸/۲۰

مجتبی بنائی

آنچه خواهید آموخت

هدف اصلی از این تمرین، آشنایی عملی با دستورات پایه SQL بخصوص انواع اتصالات، گروه‌بندی و مرتب‌سازی خواهد بود. اگر قصد کار تخصصی در حوزه نرم افزار را دارید، آشنایی با SQL به صورت تخصصی جزء ملزومات اصلی این کار خواهد بود و در این تمرین هم برآنیم که تا حد امکان به این هدف برسیم.

دیتابیس‌ای که برای این تمرین در نظر گرفته شده است، دیتابیس معروف آموزشی میکروسافت با نام Northwind¹ است که هر چند کمی قدیمی است اما برای اهداف آموزشی ما، بسیار مناسب است و نسخه پستگرس آن را در این تمرین استفاده خواهیم کرد.

این دیتابیس که یک فروشگاه بزرگ را مدلسازی می‌کند، از جداول اصلی زیر تشکیل شده است :

- Customer : مشتریان فروشگاه
- Order : سفارشات هر مشتری - اطلاعات کلی
- OrderDetails : جزئیات هر سفارش شامل آیتم‌های خریداری شده .
- Products : محصولات فروشگاه
- Suppliers: تامین کنندگان کالا که محصولات از آنها تهیه می‌شوند.
- Ctaegories : گروه بندی کالاها
- Shippers : شرکت های حمل و نقلی که وظیفه ارسال کالا به دست مشتری را بر عهده دارند.
- Employee: کارمندان فروشگاه که هر سفارش، توسط یک کارمند ثبت یا تایید می‌شود.
- Territory & Region: مناطق و نواحی ای که این فروشگاه در آنها شعبه دارد. هر منطقه شامل چندین ناحیه است و هر کارمند، می‌تواند به سفارشات چندین ناحیه رسیدگی کند.
- CustomerDemographics : برای رتبه بندی مشتریان استفاده می‌شود که در این تمرین با آن سروکار نخواهیم داشت.

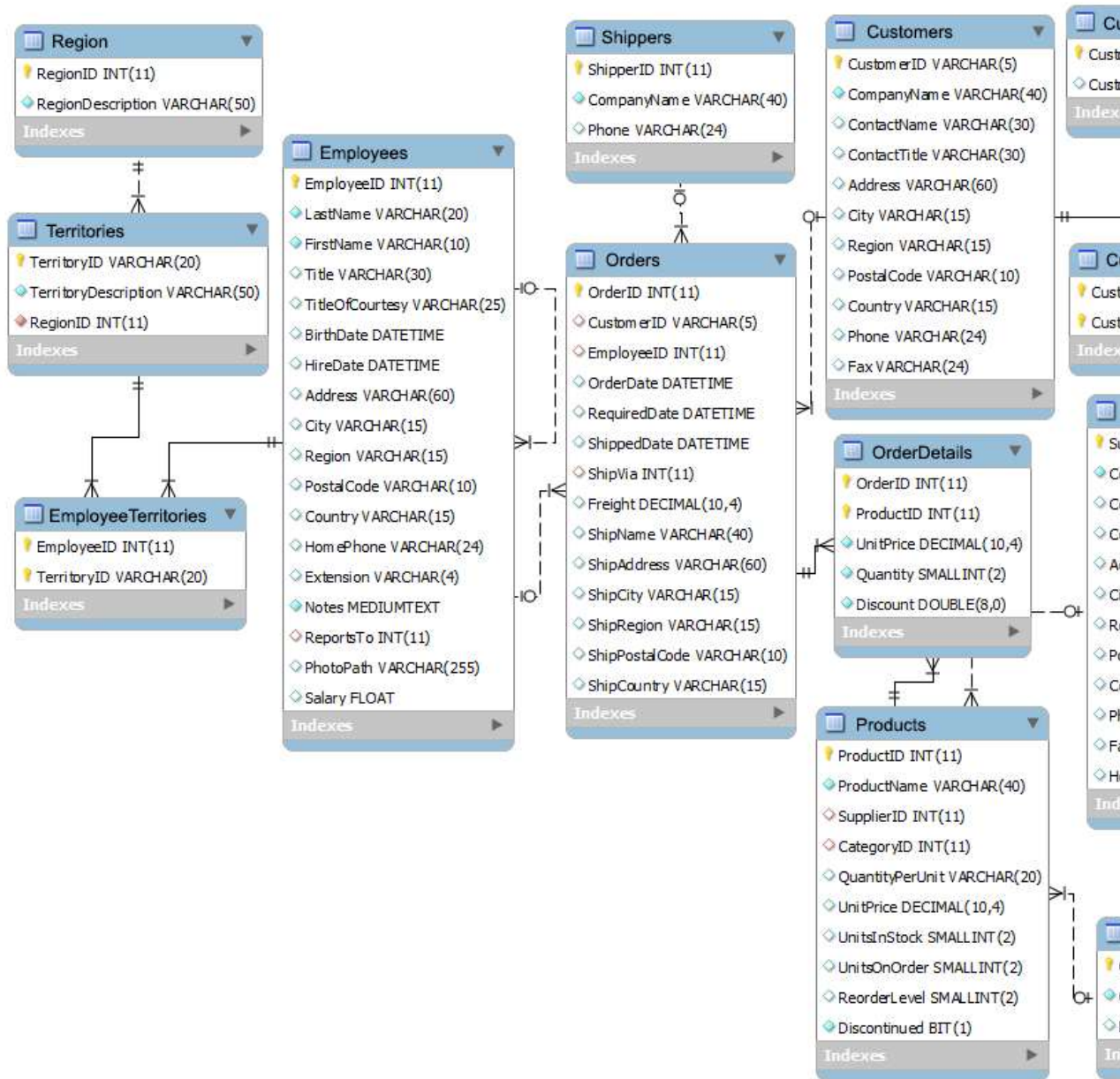
در صفحه بعد، نمودار ER این دیتابیس را مشاهده می‌کنید.

نسخه آنلاین آنرا می‌توانید از این آدرس دانلود کنید :

<https://www.w3resource.com/mysql-exercises/northwind/products-table-exercises/mysql-northwind-database.php>

¹ <https://github.com/Microsoft/sql-server-samples/tree/master/samples/databases/northwind-pubs>

دستورکار آزمایشگاه پایگاه داده

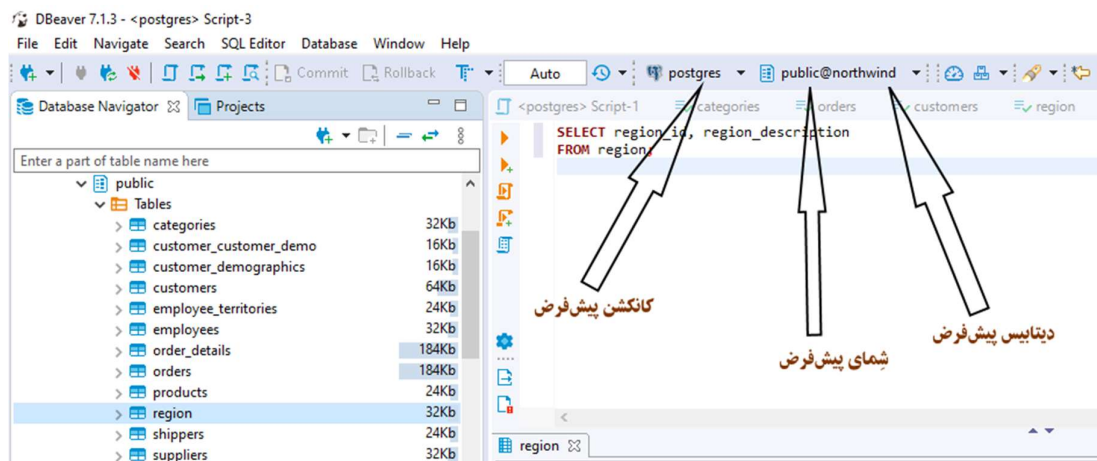


ایجاد و ایمپورت داده‌های دیتابیس Northwind

قبل از شروع به کار، باید خود دیتابیس Northwind را درون پستگرس بسازیم و سپس ساختار جداول و داده‌های آنرا با اجرای مجموعه دستورات SQL، ایجاد کنیم.

دیتابیسی با نام Northwind ایجاد کنید. (پستگرس به حروف بزرگ و کوچک حساس است بنابراین حواستان به این موضوع باشد). حال بر روی دیتابیس کلیک راست کرده، گزینه Set As Default را بزنید که دیتابیس پیش فرض شما در هنگام اجرای دستورات SQL، این دیتابیس باشد.

سپس گزینه F3 را بزنید و یا از منوی بالا، SQL Editor را انتخاب کنید. یک فایل اسکریپت جدید باز می شود که در مرحله بعد از منوی SQL Editor گزینه Import SQL Script را انتخاب کرده، فایلی که در این آدرس¹ قرار گرفته است را دانلود کرده، آنرا ایمپورت نمایید. دقت کنید که دیتابیس پیش فرض شما حتما Northwind باشد (مطابق تصویر زیر):



حال با زدن دکمه Ctrl+A همه دستورات را انتخاب کرده و مثلث کوچک نارنجی رنگ صفحه ویرایشگر SQL را برای اجرای تمام دستورات انتخاب شده، بزنید.

مطابق شکل بالا، درون شمای پایلیک باید جداول شما ساخته شده باشد. (البته نیاز خواهید داشت روی دیتابیس Northwind کلیک راست کرده و گزینه Refresh را بزنید). حال اگر جداول اصلی را باز کنید، در قسمت Data، داده‌های آنها را می توانید مشاهده کنید.

اگر تا اینجا همه چیز بدون مشکل پیش رفته باشد، آماده انجام تمرینات این دستورکار شده اید. فیلم آموزشی ایمپورت این دیتابیس، بر روی ایلرن قرار گرفته است.

نکته بسیار مهم: در تمامی سوالات این دستورکار، در سوالاتی که نیاز به گروه بندی دارید، به جای استفاده از group by آنها را طبق فیلم آپلود شده با توابع پنجره ای جایگزین کنید.

¹ https://github.com/ptthom/northwind_psql/blob/master/northwind.sql

دستورات پایه

برای انجام این تمرین، یک فایل SQL جدید ایجاد کنید و این فایلها را هم در انتها، کنار گزارش خود ارسال نمایید. (با زدن دکمه F3 و انتخاب new script)

احیاناً اگر ابهامی در یک سوال مشاهده می کنید، با در نظر گرفتن فرضی که این ابهام را برطرف کند آنرا حل کنید.

1. می خواهیم ببینیم به ازای هر منطقه (region)، چه نواحی ای تعریف شده است. دستور SQL متناظر را با استفاده از inner join جدول region و territories بنویسید. فقط نام منطقه و نام ناحیه در خروجی باید ظاهر شود.

دستور inner join مشابه این خواهد بود (r و t نام مستعار دو جدول هستند که برای ساده تر شدن نوشتن دستورات SQL معمولاً از این شیوه استفاده می کنیم):

```
from region r inner join territories t on r.region_id = t.region_id
```

2. تعداد محصولات هر کتگوری که دیگر تولید نمی شوند را (Discontinued) به دست آورید. لیست حاصل بر اساس کتگوری که بیشترین محصول از رده خارج شده را دارد باید مرتب باشد.

3. با در نظر گرفتن میزان تخفیف و تعداد هر محصول در سفارش، قیمت کل پرداختی مشتری به ازای هر سفارش را محاسبه کنید. از جدول OrderDetails استفاده کنید. سپس ده مشتری را پیدا کنید که بیشترین تخفیف را گرفته اند (مجموع تمام تخفیف های دریافت شده به ازای هر مشتری).

نکته: این گونه کوئری ها را با توابع پنجره ای، به راحتی می توانید انجام دهید که در دستور کار ششم به آنها پرداخته شده است. بهتر است از روش سنتی و بدون استفاده از توابع پنجره ای، این مساله را حل کنید.

4. ده محصولی که بیشترین تعداد خرید را به خود اختصاص داده اند را بیابید. منظور از تعداد خرید، تعداد فاکتورهای صادر شده به ازای آن محصول است و نه تعداد خرید هر مشتری در هر فاکتور. منظور از فاکتور همان جدول Orders است.

5. آیا محصولی وجود دارد که هنوز فروش نرفته باشد؟

6. می خواهیم گزارشی تهیه کنیم از کتگوری ها و تعداد سفارش هایی که روی کالاهای این کتگوری انجام شده است. (تعداد سفارشها یا فاکتورهایی که شامل کالاهای یک کتگوری هستند). دستور sql مربوطه را بنویسید. دقت کنید که برای این سوال، نیاز دارید از جویین های خارجی (چپ یا راست) استفاده کنید چون ممکن است یک کتگوری هنوز محصول فروش رفته ای نداشته باشد.

نکته: کافی است جدول کتگوری، محصول و جزییات سفارش را جویین کنید (یک جویین این کوئری از نوع داخلی و دومی از نوع خارجی خواهد بود) و سپس بر اساس کتگوری، گروه بندی کرده و تعداد کالاها را بشمرید.

7. می خواهیم به کارمندی که بیشترین فروش (از لحاظ مبلغ پرداختی مشتری) را در یک سال خاص داشته است (یک سال فرضی در نظر بگیرید)، جایزه بدهیم. این کارمند را چگونه به دست می آوریم؟ از توابع تاریخ در پستگرس برای استخراج سال از روی OrderDate و گروه بندی داده ها بر اساس آن، استفاده کنید. توصیه می کنیم برای راحتی کار از CTE در SQL استفاده کنید. یعنی جداول مورد نیاز را به صورت موقت در ابتدا بسازید (با with) - مثل جدولی

که خروجی دستور سوم است و اینجا به آن نیاز دارید- سپس از آن جدول در دستورات پایین تر استفاده کنید. به مثال زیر دقت کنید :

```
WITH regional_sales AS (
    SELECT region, SUM(amount) AS total_sales
    FROM orders
    GROUP BY region
), top_regions AS (
    SELECT region
    FROM regional_sales
    WHERE total_sales > (SELECT SUM(total_sales)/10 FROM regional_sales)
)
SELECT region,
    product,
    SUM(quantity) AS product_units,
    SUM(amount) AS product_sales
FROM orders
WHERE region IN (SELECT region FROM top_regions)
GROUP BY region, product;
```

8. می خواهیم به سفارشات که همان روز سفارش ارسال شده اند (یعنی **OrderDate** و **ShippedDate** آنها برابر است)، برچسب **عالی**، آنهایی که تا سه روز بعد از سفارش ارسال شده اند، برچسب **خوب** و به بقیه سفارشات برچسب **نامناسب** بزنیم . در گزارش نهایی، کد سفارش و برچسب آن باید ظاهر شود. از دستور **case** و **when** استفاده کنید.

9. (سوال امتیازی) برای دستور **WITH RECURSIVE** در این دیتاست، یک مثال بنویسید که خروجی مناسب تولید کند.

10. می خواهیم سفارش جدیدی برای محصولات که تعداد آنها (**UnitsInStock**) از حد مجاز (**ReorderLevel**) کمتر شده است، بدهیم. این محصولات را بیابید و آنها بر اساس میزان موجودی (به صورت صعودی) نمایش دهید. یک ویو برای ذخیره این دستور ایجاد کنید که همواره برای مشاهده محصولات که نیاز به خرید دارند، این ویو، استفاده شود.

11. از کدام دسته محصولات، بیشترین ارسال به آلمان را داشته ایم؟

12. مشتریانی را بیابید که شماره فکس آنها در سیستم موجود نیست . (مقایسه **fax** با **null**)

درباره توابع پنجره‌ای : در دنیای واقعی، ما دستور گروه‌بندی را حذف نمی‌کنیم و بسته به نیاز از یکی از این دو رهیافت و یا ترکیب این دو استفاده می‌شود. مثلاً اگر قصد تعیین رتبه یک محصول بر اساس فروش آنرا داشته باشید می‌توانید در یک دستور، ابتدا تمامی داده‌های فروش را با کد محصول گروه‌بندی کنید (این دستور قبل از توابع پنجره‌ای اجرا میشود) و سپس به کمک تابع پنجره‌ای **rank** بر اساس میزان فروش هر محصول که خروجی گروه‌بندی شماست، آنها را رتبه‌بندی کنید. در این دستور کار، برای اینکه با توابع پنجره‌ای آشنایی کافی پیدا کنید، تمام گروه‌بندی‌ها را به کمک این توابع پنجره‌ای انجام بدهید و مستقیماً از **group by** استفاده نکنید. (فیلم آموزشی مرتبط با توابع پنجره‌ای را حتماً نگاه کنید و برای آشنایی بیشتر به این آدرس¹ مراجعه کنید.)

¹ yun.ir/window