



## به نام هستی بخش

"و سلام بر مهدی که انتظارش را نه فقط دل عاشق،  
که ترنم هر باران بهاری و هر روزنه‌ی امید می‌کشد..."

### تمرین ۴ (فصل ۵)

موعد تحویل: پنج‌شنبه ۲۶ فروردین

درس پایگاه داده‌ها، بهار ۱۴۰۰

## مقدمه

در این تمرین قرار است که پرس‌وجوهایی که در ادامه آمده است را به زبان MySQL بنویسید. دقت داشته باشید که تنها زمانی به پرس‌وجوی شما نمره‌ای تعلق می‌گیرد که کد شما به صورت صحیح اجرا شود. این تمرین باید به طور انفرادی انجام گردد، بنابراین این با هر گونه تقلب در نوشتن کدها برخورد می‌شود. پس لطفا سعی کنید تمرین را خودتان حل کنید!

## نحوه آپلود

شما در نهایت برای این تکلیف باید یک فایل تکست با پسوند sql آپلود کنید که تمام پرس‌وجوها در آن فایل موجود باشد. برای تمیزی فایل حتما در آخر هر پرس‌وجو نقطه‌ویرگول (semicolon) ";" بزنید و همچنین قبل هر پرس‌وجو با کامنت گذاشتن (با استفاده از "#") شماره سوال را مشخص کنید. به مثال زیر دقت کنید:

```
1  # 1
2  SELECT * FROM Suppliers;
3
4  # 2
5  SELECT s.SupplierID
6  FROM Suppliers s;
7
```

پیشنهاد می‌شود برای تمیزی بیشتر پرس‌وجو خود را در چند خط با اینتر جدا کنید (مانند پرس‌وجوی شماره دو) همچنین پیشنهاد می‌شود که برای تیبل‌ها اسم مستعار (alias) بگذارید که کویری ساده‌تر شود. (مانند "s" برای "Suppliers" در خط ۶).

## راه اندازی MySQL

برای راه اندازی MySQL بسته به سیستم عامل خود راه حل های متنوع و زیادی وجود دارد که می توانید با سرچ کردن آن ها را پیدا کنید. به عنوان یک راه عمومی برای همه سیستم عامل ها می توانید از فایل docker-compose زیر استفاده کنید:

```
https://gist.github.com/mohalisad/5de8efbb15b5ea533f2104241e6c25e3
```

برای راه اندازی کافی است که فایل را با نام "docker-compose.yml" در پوشه ای خالی قرار دهید و با دستور زیر MySQL را راه اندازی کنید. (قبل از آن نیاز دارید که علاوه بر داکر آخرین نسخه docker-compose را نصب داشته باشید که آموزش این کار در سایت داکر موجود است)

```
docker-compose up -d
```

دستور بالا در صورت لزوم ایمپج مورد نظر را دانلود کرده (ترجیحا VPN فعال داشته باشید که موقع دانلود مشکلی به وجود نیاید) و سپس آن را اجرا می کند. مشخصات آن نیز در زیر آمده است:

Port	3306
Username	root
Password	example

## اتصال به MySQL

برای اتصال به MySQL کلاینت های متنوع و زیادی وجود دارند و شما می توانید از هر کدام از آن ها استفاده کنید. چند تا از کلاینت های موجود که در همه سیستم عامل ها قابلیت اجرا شدن را دارند:

- MySQL Workbench
- Navicat
- DBeaver
- phpMyAdmin (تحت مرورگر)

## وارد کردن داده ها به MySQL

شما برای انجام این پروژه ابتدا باید فایل northwind.sql را در پایگاه داده خود وارد کنید (import کنید). با این کار پایگاه داده ای با نام northwind ایجاد می شود.

<sup>1</sup> <https://docs.docker.com/compose/install/>

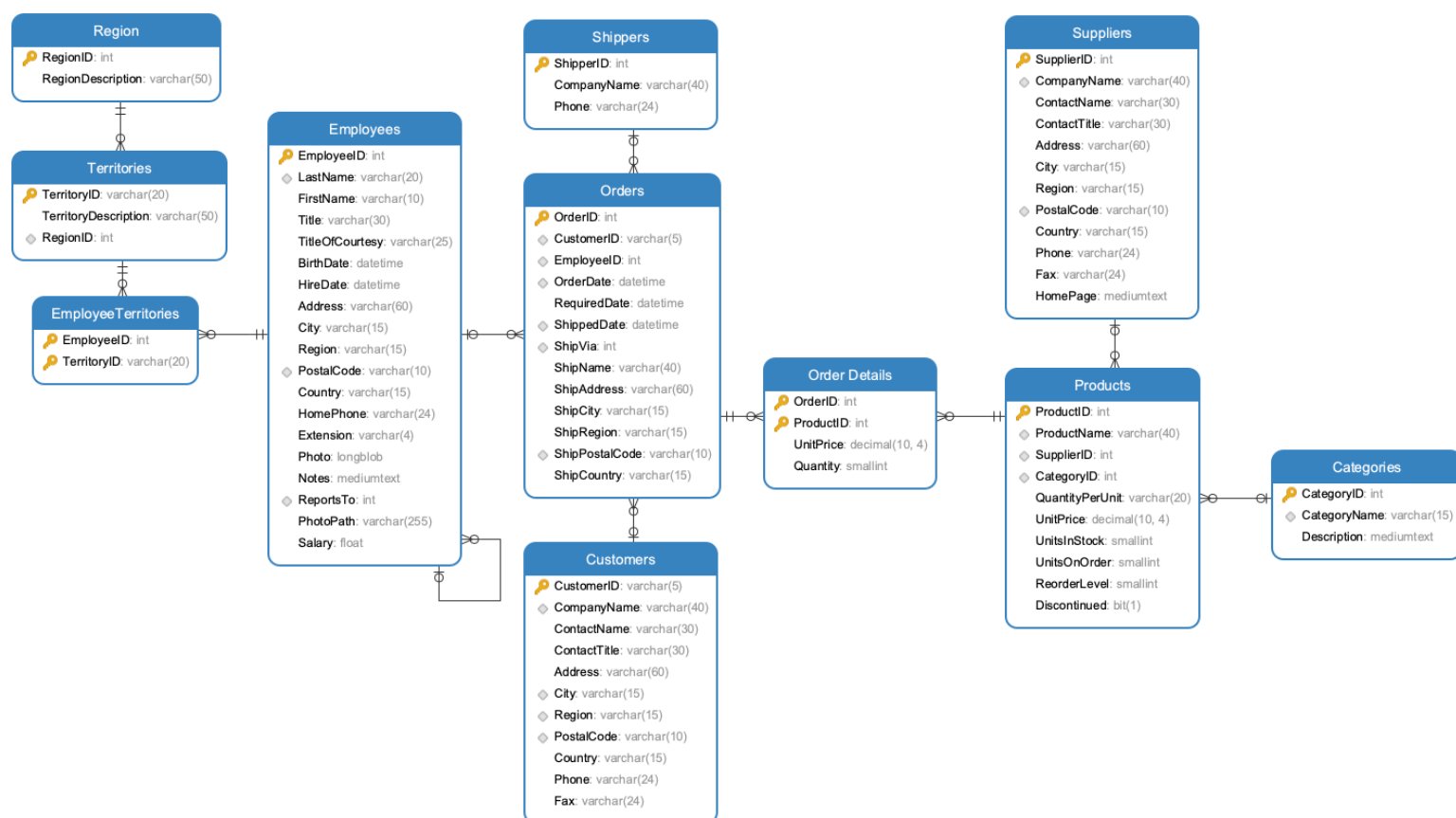
## شمای پایگاه داده

طراحی پایگاه داده را در شکل زیر می‌توانید مشاهده کنید. در این پایگاه داده تعدادی کالا وجود دارند (Products) هر کالا دارای یک دسته‌بندی (Categories) می‌باشد همچنین برای هر کالا تامین کننده (Suppliers) آن نیز مشخص است.

یک رابطه many2many به نام "Order Details" بین سفارشات (Orders) و کالاها موجود است که در آن رابطه مشخصاتی اعم از قیمت هر واحد و همچنین تعداد آن‌ها در فاکتور آمده است. هر سفارش دارای یک فرستنده (Shippers) می‌باشد. همچنین هر سفارش دارای یک مشتری (Customers) و یک کارمند (Employees) مسئول نیز می‌باشد.

یک سری از کارمندان باید به یک کارمند دیگر گزارش دهند (ReportsTo). همچنین هر کارمند مسئول تعدادی ناحیه (territories) می‌باشد که این موضوع با رابطه many2many به نام (EmployeeTerritories) پیاده سازی شده است.

همچنین هر ناحیه متعلق به یک منطقه (Region) می‌باشد.



## نکات مهم در پرس وجوها

- حتما اگر می خواهید عنصری را با NULL مقایسه کنید از عبارت "x IS NULL" استفاده کنید.
- ساختار (Syntax) گفته شده برای Trigger ها در اسلایدهای درس متعلق به System R است و این ساختار در MySQL اعتبار ندارد عکس زیر شامل مثالی برای ایجاد Trigger در MySQL می باشد.

```
DELIMITER //
CREATE TRIGGER manager_inc
AFTER UPDATE ON dept
FOR EACH ROW
IF (OLD.manager != NEW.manager) THEN
    UPDATE employee e SET e.salary = e.salary * 1.1 WHERE e.number = NEW.manager;
END IF; //
DELIMITER ;
```

## پرس وجوهای SELECT

- پرس وجوهای زیر هر کدام ۵ نمره می باشند. (جمعا ۵۵ نمره)
- ۱- نام تمام محصولاتی که قیمتشان بین ۱۵ تا ۲۰ است.
  - ۲- نام و قیمت ارزان ترین محصول. (بدون استفاده از LIMIT)
  - ۳- نام و تعداد بیشترین محصولی که موجود (UnitsInStock) است. (با استفاده از LIMIT)
  - ۴- در یک ردیف تعداد کارمندانی که حقوق بالای ۲۰۰۰ دریافت میکنند و در ردیف دیگر تعداد کارمندانی که حقوق زیر ۱۸۰۰ دارند. (به کمک UNION)
  - ۵- قیمت میانگین هر دسته همراه با نام آن دسته برای دسته هایی که قیمت میانگین بالای ۲۵ دارند.
  - ۶- نام یکتا مشتریانی که اسمشان با "M" شروع می شود و سفارشی با فرستنده "Federal Shipping" داشته اند.
  - ۷- نام و نام خانوادگی یکتا افرادی که خود به هیچ کس گزارش (ReportsTo) نمی دهند ولی وجود دارد حداقل یک نفر که به آن فرد گزارش دهد.
  - ۸- شماره سفارش (OrderID) و مجموع قیمت هر سفارش. (مجموع حاصل ضرب UnitPrice و Quantity)
  - ۹- تعداد مشتریانی که سفارشی با مجموع قیمت بالای ۵۰۰۰ ثبت کرده باشند.
  - ۱۰- نام ناحیه هایی (TerritoryDescription) که کارمندی مسئولیت آن جا را به عهده ندارد.
  - ۱۱- نام مشتریانی که فقط دو کارمند (یکتا) مسئول سفارششان بوده اند.

## ایجاد VIEW

پرس وجوهای زیر هر کدام ۷,۵ نمره می باشند. (جمعا ۲۲,۵ نمره)

۱۲- دستور ایجاد VIEW ای را بنویسید که اسم هر مشتری همراه با مجموع خریدهایی کرده است را نمایش دهد.

۱۳- دستور ایجاد VIEW ای را بنویسید که جلوی اسم هر کارمند تعداد افرادی که به او گزارش می دهند را نمایش دهد و اگر این تعداد صفر بود در جدول این کارمند نمایش داده نشود. (راهنمایی: این ویژگی به راحتی با GROUP BY قابل پیاده سازی است)

۱۴- دستور ایجاد VIEW ای را بنویسید که جلوی اسم هر تامین کننده (Suppliers) تعداد کالاهایی که موجودیشان (UnitsInStock) صفر است و توسط آن تامین کننده تامین می شود را نمایش دهد. می خواهیم در صورت این که این تعداد صفر بود باز تامین کننده در جدول باشد. (راهنمایی: این ویژگی با کمک Nested Query در جلو SELECT قابل پیاده سازی است مانند زیر)

```
SELECT x, (SELECT COUNT(*) FROM ...) FROM ... ;
```

## ایجاد TRIGGER

پرس وجوهای زیر هر کدام ۷,۵ نمره می باشند. (جمعا ۲۲,۵ نمره)

۱۵- دستور ایجاد TRIGGER ای را بنویسید که در هنگام اضافه شدن یک رکورد به "Order Details" بررسی کند که آیا قیمت این رکورد با قیمت کالا برابر است یا خیر و اگر برابر نبود با نمایش پیغام زیر از اضافه شدن ردیف جلوگیری کند.

```
Invalid Price: Price is inconsistent
```

۱۶- دستور ایجاد TRIGGER ای را بنویسید که در هنگام اضافه شدن یک رکورد به "Order Details" بررسی کند که آیا تعداد درخواست شده از تعداد باقی مانده کالا (UnitsInStock) بیشتر است یا خیر. اگر کمتر یا مساوی بود باید تعداد باقی مانده کالا آپدیت شود (مقدار Quantity از آن کم شود) و اگر بیشتر بود از با پیغام زیر از اضافه شدن ردیف جلوگیری کند.

```
Invalid Quantity: Quantity is higher than UnitsInStock
```

۱۷- می خواهیم کارمندانی که دارای پیشوند "Dr." هستند در هنگام آپدیت حقوقشان دوبرابر شود همچنین اگر کسی این پیشوند را از دست داد حقوقش نصف شود. این کار را با بررسی "TitleOfCourtesy" انجام دهید. (راهنمایی: ابتدا چک کنید که آیا ردیفی که آپدیت شده پیشوندش تغییر کرده است، سپس بر اساس نوع تغییر حقوق را تغییر دهید)

سربلند و پیروز باشید