



تکلیف سوم پایگاه داده یک

اردیبهشت ۱۴۰۳

مهلت ارسال پاسخ

۱۴۰۳/۰۳/۰۴

استاد درس

حمید روایی

نحوه تحویل:

پاسخ سوالات در یک فایل Zip که بصورت نام، شماره دانشجویی و شماره تکلیف، مانند:

HW3_Mahdi_Aghajanian_99876543

نامگذاری شده است در سامانه یکتا بارگذاری شود (کدهای هر سوال را در فایل SQL مجزا در کنار pdf

پاسخهایتان قرار دهید و تصویری از خروجی حاصل از کدتان در pdf پاسخها قرار دهید).

پاسخهای شما حتما باید به صورت تایپ شده و به زبان فارسی باشد و در صورت بارگذاری تکلیف به

صورت دستنویس، تکلیف شما تصحیح نخواهد شد.

در صورت وجود هر گونه سوال می توانید با مهدی آقاجانین از طریق تلگرام در ارتباط باشید

<https://t.me/MahdiAghajanian>



۱. جداول کتاب و کاربر از سوال آخر تکلیف اول را ایجاد کنید. (در صورت استفاده از foreign key در این جداول، جداول مربوطه را نیز ایجاد کنید).

۲. با استفاده از پایگاه داده university: Large version به سوالات زیر پاسخ دهید:

- a. به جدول Students یک فیلد اضافه کنید که تعداد واحدهای گذرانده شده توسط هر دانشجو در رکورد مربوط به آن دانشجو، ذخیره شود.
- b. یک کوئری بنویسید که نمره هر دانشجو در هر درس را یک نمره اضافه کند (به عنوان مثال دانشجو با نمره A- در یک درس باید نمره اش به A تغییر کند).
- c. ستون grade را بگونه‌ای تغییر دهید که غیر از مقادیر مجاز، مقدار دیگری در آن امکان insert نداشته باشد.
- d. یک ستون numeric_grade اضافه کنید و نوع آن را varchar(255) قرار دهید؛ سپس مقدار آن را مشابه با مقدار ستون grade پر کنید.
- e. یک کوئری Update بنویسید که در ستون numeric_grade نمره هر دانشجو را به مقدار عددی معادل آن تغییر دهد.
- f. یک کوئری بنویسید تمامی دانشجویانی که حداقل در یک درس مردود شده‌اند، از پایگاه داده حذف شوند. (برای پاسخ دهی به باقی سوالات دوباره پایگاه داده university: Large version را ایجاد کنید و با داده های اصلی به ادامه سوالات پاسخ دهید)

۳. دستورات زیر را با توجه به خواسته سوال در صورت امکان بازنویسی کنید:

a. بازنویسی بدون استفاده از دستور unique

```
WHERE unique ( SELECT name FROM student )
```

b. بازنویسی بدون استفاده از دستور Like

```
WHERE first_name LIKE 'me%' AND last_name LIKE '%avi'
```

c. بازنویسی بدون استفاده از outer join

```
select * from student natural left outer join takes
```

d. بازنویسی بدون استفاده از outer join

```
select * from student natural full outer join takes
```



۴. با استفاده از پایگاه داده university: Large version کوئری‌هایی بنویسید که به سوالات زیر پاسخ دهد:

a. فهرستی از همه Instructor ها را نمایش دهید که ID هر Instructor و تعداد Section های آموزش داده

شده را نشان می دهد. حتماً تعداد Section ها را برای Instructor هایی که هیچ Section ای تدریس

نکرده‌اند، ۰ نشان دهید.

b. شناسه و نام دانشجویانی که نام آنها با M شروع و با a پایان می‌یابد.

c. نام دروسی را بیابید که در دانشکده های مهندسی ارائه شده باشند (نام آنها با Eng. ختم شوند) و در ترم پاییز

۲۰۰۹ ارائه شده باشند.

d. شماره اتاق‌هایی را بدست بیاورید که در ترم بهار ۲۰۰۸ در طی هفته در مجموع ۲ ساعت یا بیشتر اشغال

هستند (فقط ساعت مد نظر است و دقیقه مدنظر نیست).

e. لیست دانشجویان کامپیوتر به همراه تعداد واحدهای اخذ شده آنها در ترم بهار ۲۰۰۵ که تعداد واحدهای

بیشتری نسبت به میانگین تعداد واحدهای اخذ شده توسط کل دانشجویان اخذ کرده‌اند (حتما با with نوشته

شود).

f. لیست دانشجویان کامپیوتر را پیدا کنید که درس‌هایی در این رشته وجود دارد که هنوز اخذ نکرده‌اند.

۵. فرض کنید در یک جدول تراکنش های مشتریان ذخیره شده است. در این جدول آیدی تراکنش، اسم

فرد و آدرس وی هنگام ثبت تراکنش ذخیره می‌شود (سه فیلد اول مثال زیر). می‌خواهیم اگر فردی در

یک تراکنش مقدار آدرس را وارد کرد تا زمانی که آدرس جدیدی وارد نشده است، در فیلدی بنام

Address2 مقدار آدرس قبلی وارد شده نمایش داده شود و در صورت ثبت آدرس جدید، آدرس جدید

جایگزین شود. کوئری بنویسید که این خروجی را بسازد.

id	person	Address	Address2
1	bob	null	null
2	bob	null	null
3	jim	null	null
4	bob	A	A
5	bob	null	A
6	bob	B	B
7	jim	X	X
8	jim	Y	Y
9	jim	null	Y



۶. جدول CTRN تراکنش های مشتریان یک فروشگاه زنجیره ای را نگه می دارد. در این جدول CID شماره مشتری، Time لحظه انجام تراکنش و Amount مبلغ تراکنش می باشد. جدول CUSTOMER نیز مشتریان فروشگاه را شامل می شود و فیلد CID شماره مشتری است. می خواهیم جمع تراکنش های هر مشتری، در شش ماه دوم سال ۱۴۰۲ را به تفکیک هر ماه را داشته باشیم و ماکزیمم آن ها را نیز در ستون دیگری در خروجی نمایش دهیم.

توجه کنید که برای هر مشتری فقط یک رکورد در جدول نهایی باید قرار داده شود.

پیشنهاد میشود در باره pivot function تحقیق کنید.

نمونه جدول:

CID	Month7	Month8	Month9	Month10	Month11	Month12	Max
1234	1,700,000	1,000,000	1,200,000	1,900,000	1,300,000	1,700,000	1,900,000
5678	1,800,000	1,100,000	1,300,000	1,800,000	1,400,000	1,600,000	1,800,000

۷. جداول زیر را ایجاد کرده و با استفاده از آن ها کوئری هایی بنویسید که به سوالات زیر پاسخ دهد:

- STD (S# ,S name ,MJR ,Degree ,AGE) دانشجو
- CRS (C# , C name , term , Room ,P#) درس
- REG (S# , C#) ثبت نام
- PRF (P# , P name , D#) استاد

- سن مسن ترین دانشجویی را بدهید که یا در رشته ICT تحصیل می کند یا در کلاس استاد P1 ثبت نام کرده است.
- نام کلاس هایی که در اتاق R128 برگزار شده یا دارای تعداد ۵ عدد یا بیشتر دانشجو باشد.
- نام دانشجویانی را بدهید که در دو کلاس که در یک ساعت برگزار می شوند ثبت نام کرده اند.
- بر هر مقطع تحصیلی به جز مقطع دکترا، متوسط سن دانشجویان آن مقطع را حساب کنید.
- نام دانشجویانی را بدهید که در کلاس هایی با بیشترین تعداد دانشجو شرکت کرده اند.
- نام دانشجویانی را بدهید که در کلاس هایی که شرکت کرده اند که تعداد شرکت کنندگان آن ها بیشتر از متوسط شرکت کنندگان در باقی کلاس ها می باشد
- نام دانشجویانی را بدهید که در هیچ کلاسی ثبت نام نکرده اند.
- نمره دانشجویانی که نمرشان از متوسط کل دانشجویان کمتر است را ۱۰٪ اضافه کنید و دانشجویانی که نمرشان بیشتر یا مساوی میانگین است را ۵٪ اضافه کنید.



۸. تحت چه شرایطی کوئری:

```
select *  
from student natural full outer join takes  
natural full outer join course
```

شامل تاپل‌هایی با مقادیر تهی برای ویژگی title می شود؟

۹. بر اساس نمرات زیر view هایی ایجاد کنید که به سوالات زیر پاسخ دهد:

A+ = 100 , A = 95, A- = 90

B+ = 85, B = 80, B- = 75

C+ = 70, C = 65, C- = 60

Else: 0

(a) معدل عددی هر دانشجو را نمایش دهد.

(b) نام هر دانشجو به همراه نام هر استاد و میانگین نمرات دروسی که آن دانشجو با آن استاد داشته است.

موفق باشید (:)