

بسمه تعالی



پایگاه داده ۱

دانشکده برق و کامپیوتر

بهار ۱۴۰۳

استاد:

دکتر شیرین بقولی زاده

زمان تحویل:

۱۴۰۳/۰۲/۲۳

## نکات تحویل تکلیف

- فایل‌های خود را با فرم HW3\_Name\_NumStudent.zip ارسال کنید. توجه شود که به فایل‌هایی که نام گذاری آنها بصورت فرمت گفته شده نباشد، نمره‌ای تعلق نخواهد گرفت.
- به پاسخ‌های بدون توضیح امتیاز و نمره‌ای تعلق نخواهد گرفت.
- در صورت وجود هرگونه ابهام می‌توانید از طریق تلگرام سؤالات خود را با دستیارآموزشی مربوط به این تکلیف مطرح کنید. آیدی تلگرام دستیارآموزشی مربوط به این تکلیف: [@mrymfakhari](#)
- در صورت وجود شباهت واضح، نمره‌ای به پاسخ تعلق نمی‌گیرد.
- پاسخ تکلیف را حتماً در سامانه یکتا آپلود کنید و از ارسال تکلیف به ایمیل یا تلگرام اکیداً خودداری کنید.
- توصیه می‌شود به سؤالات مربوط به هر قسمت با ترتیب موجود در خود سؤالات پاسخ دهید (سعی شده هر سوال راهنمای سوال بعد از خودش باشد).
- پاسخ‌های شما باید بصورت فایل کد بصورت جداگانه برای هر سوال باشد یعنی باید ۵ فایل SQL در فایل zip موجود باشد.
- برای کد زنی می‌توانید از ابزارهای مختلفی استفاده کنید ولی توصیه ما به شما این است که از SQL Server Management Studio (SSMS) استفاده کنید. برای نصب SSMS هم می‌توانید از این [لینک](#) استفاده کنید.
- اگر به هر دلیلی نمی‌توانید ابزار مناسب را بر روی سیستم خود استفاده کنید، می‌توانید از این [سایت](#) استفاده کنید.
- با توجه به دیتامدل زیر برای هر یک از موارد خواسته شده در سؤالات دستور sql مناسب بنویسید.

```
CREATE TABLE "Students" (  
  "id" INT PRIMARY KEY,  
  "name" VARCHAR(100),  
  "family" VARCHAR(100),  
  "mobile" VARCHAR(50),  
  "isGraduated" BOOLEAN,  
  "majorId" VARCHAR(20),  
  "major_name" CHAR  
);
```



	id	name	family	mobile	isGraduated	majorId	major_name
1	3779898	بزرگ	قدیمی	10203040506	true	1	نرم افزار
2	3921478	دلدار	خجسته	<null>	false	1	نرم افزار
3	3981122	نادر	ابراهیمی	09515253	true	2	هوش مصنوعی
4	4001234	محمدرضا	گلزار	09223344556	false	1	نرم افزار
5	4004321	مجید	سمیعی	<null>	false	2	هوش مصنوعی
6	4012222	علی	صادقی	09131234567	false	3	علم داده
7	4015555	عادل	فردوسی پور	<null>	false	3	علم داده
8	4025678	مریم	میرزاخانی	<null>	false	1	نرم افزار

```
create table "Courses"  
(  
  "id" integer not null constraint courses_pkey primary key,  
  "name" varchar(20),  
  "department" varchar(20)  
);
```



	id	name	department
1	1	پایگاه داده	مهندسی کامپیوتر
2	3	گسسته	ریاضی
3	2	طراحی الگوریتم	مهندسی کامپیوتر
4	4	قانون اساسی	معارف

برای بررسی راحت تر کد ایجاد جداول در اختیار شما قرار گرفته است. انتظار می‌رود شما داده‌های بیشتری برای تست ایجاد کنید:

```
INSERT INTO "Students" (id, name, family, mobile, "isGraduated", "majorId", major_name) VALUES (3779898, 'بزرگ', 'قدیمی', '10203040506', true, '1', 'نرم افزار');  
INSERT INTO "Students" (id, name, family, mobile, "isGraduated", "majorId", major_name) VALUES (3921478, 'دلدار', 'خجسته', null, false, '1', 'نرم افزار');  
INSERT INTO "Students" (id, name, family, mobile, "isGraduated", "majorId", major_name) VALUES (3981122, 'نادر', 'ابراهیمی', '09515253', true, '2', 'هوش مصنوعی');  
INSERT INTO "Students" (id, name, family, mobile, "isGraduated", "majorId", major_name) VALUES (4001234, 'محمدرضا', 'گلزار', '09223344556', false, '1', 'نرم افزار');  
INSERT INTO "Students" (id, name, family, mobile, "isGraduated", "majorId", major_name) VALUES (4004321, 'مجید', 'سمیعی', null, false, '2', 'هوش مصنوعی');  
INSERT INTO "Students" (id, name, family, mobile, "isGraduated", "majorId", major_name) VALUES (4012222, 'علی', 'صادقی', '09131234567', false, '3', 'علم داده');  
INSERT INTO "Students" (id, name, family, mobile, "isGraduated", "majorId", major_name) VALUES (4015555, 'عادل', 'فردوسی پور', null, false, '3', 'علم داده');  
INSERT INTO "Students" (id, name, family, mobile, "isGraduated", "majorId", major_name) VALUES (4025678, 'مریم', 'میرزاخانی', null, false, '1', 'نرم افزار');
```

```
INSERT INTO "Courses" (id, name, department) VALUES (1, 'پایگاه داده', 'مهندسی کامپیوتر');  
INSERT INTO "Courses" (id, name, department) VALUES (2, 'الگوریتم طراحی', 'مهندسی کامپیوتر');  
INSERT INTO "Courses" (id, name, department) VALUES (3, 'گسسته', 'ریاضی');  
INSERT INTO "Courses" (id, name, department) VALUES (4, 'قانون اساسی', 'معارف');
```

## سوال ۱ (۲۵)

هدف از این قسمت یادگیری ایجاد محدودیت روی سطرهای جداول برای تهیه گزارش است:

الف) دستوری بنویسید که جدولی به نام "StudentCourseRelations" مانند جدول زیر ایجاد کند. توجه شود که خروجی جدول شما دقیقا باید مانند زیر باشد:

	id	student_id	course_id	year	grade
1	05	4015555	1	1401	11
2	08	4015555	3	<null>	18.5
3	11	4025678	3	1403	20
4	04	4025678	3	1402	19
5	02	3779898	1	1380	13
6	01	4001234	1	1401	18
7	07	4015555	2	1400	16
8	09	4004321	1	1401	15
9	10	4001234	3	<null>	17
10	03	4004321	2	1403	<null>

جدول ۱: StudentCourseRelations

ب) دستوری بنویسید که نام، نام خانوادگی و شماره دانشجویی دانشجویانی که نام آنها "علی" و "محمد" نباشد و ورودی ۱۳۹۹ و بعد از آن باشند و شماره موبایل آنها در صورت وجود معتبر باشد (شماره موبایل معتبر باشد یا اصلا وارد نشده باشد) را برگرداند. (برای اعتبار شماره موبایل میتوان صحت تعداد ارقام و شروع شماره با ۰۹ را بررسی کرد).

پ) دستوری بنویسید که نام، نام خانوادگی و شماره دانشجویی همه دانشجویانی که نام آنها "علی" و "محمد" نباشد، بعلاوه همه دانشجویانی که ورودی ۱۳۹۹ و بعد از آن هستند و هم چنین همه دانشجویانی که شماره موبایل آنها معتبر باشد یا اصلا وارد نشده است را برگرداند (خروجی سطر تکراری نداشته باشد).

ت) دستوری بنویسید که شماره دانشجویی و شناسه درس و نمره درس را نمایش دهید به طوری که نمرات با  $course\_id = 1$ ، از عدد دو کم شود و نمرات با  $course\_id = 2$ ، صفر و نمرات با  $course\_id = 3$ ، با ۲ درصد افزایش نمره باشد (اسلاید ۱۵ و ۱۶).

ث) دستوری بنویسید که نام، نام خانوادگی و رشته تحصیلی دانشجو و نام درس و نمراتی که بعد از سال ۱۳۰۰ برای دانشجو ثبت شده است را برای دانشجویان که هنوز فارق التحصیل نشده اند برگرداند. خروجی بر اساس شماره دانشجویی مرتب شده باشد (اسلاید ۱۹ و ۱۷).

## سوال ۲ (۳۵)

هدف از حل این سوالات با استفاده از توابع تجمعی و دستور group by آشنا خواهید شد

الف) کمترین نمره ای که بین سالهای ۱۳۹۹ و ۱۵۰۰ برای درسهایی با  $course\_id = 1$  و  $course\_id = 3$  ثبت شده است.

ب) دو کمترین نمره ای که تاکنون ثبت شده را به ترتیب از بزرگ به کوچک چاپ کنید (خروجی نباید تکراری باشد).

پ) کمترین نمره ثبت شده برای هر درس را همراه با شناسه درس به ترتیب کمترین نمره چاپ کنید.

ت) مجموع نمرات هر دانشجو که در حداقل دو درس ثبت نام کرده را همراه با شماره دانشجویی دانشجو چاپ کنید.

ث) برای هر درسی که حداقل یکبار اخذ شده و حداقل یک نمره بیشتر از ۵ برای آن وجود دارد، بیشترین نمره ثبت شده برای این درس و شناسه درس را چاپ کنید.

ج) برای تمام دانشجویانی که یک درس را دوبار اخذ کرده اند، شماره دانشجویی و شناسه درس را چاپ کنید.

خ) شناسه تمام درسهایی که هم در سال ۱۴۰۰ و هم در سال ۱۴۰۱ ارائه شدند ولی در هیچ یک از سالهای ۱۴۰۲ و ۱۴۰۳ ارائه نشده اند (اسلاید ۲۸).

## سوال ۳ (۲۰)

در این قسمت با نوشتن کوئری به صورت تودرتو و ... به مباحثی که در بالا گفته شد ان شاءالله: مسلط خواهید شد.

الف) در این قسمت با نوشتن کوئری به صورت تودرتو و ... به مباحثی که در بالا گفته شد ان شاءالله: مسلط خواهید شد.

ب) گزارشی حاوی شماره دانشجویی و تعداد دروسی که توسط دانشجو اخذ شده و نمره آن درس بیشتر از ۱۴ باشد.

پ) گزارشی حاوی شماره دانشجویی دانشجویانی که میانگین نمرات آنها بیشتر از ۱۴ باشد همراه با تعداد دروسی که دانشجو اخذ کرده است.

ت) برای دانشجویانی که میانگین نمراتشان بیشتر از میانگین کل نمراتی که تاکنون ثبت شده باشد، گزارشی حاوی شماره دانشجویی و تعداد دروسی که توسط دانشجو اخذ شده و میانگین نمرات آن دانشجو ارائه دهید.

## سوال ۴ (۳۰)

در این قسمت باعث ایجاد تغییر در دیتابیس خواهیم شد. برای پاسخ گویی به این سوالات باید از دستورات insert و update و delete استفاده کنید.

الف) دستور ساخت جدول StudentCourseRelations را بنویسید. با توجه به محتوای جداول Students و Courses یک سطر داده به جدول StudentCourseRelations اضافه کنید.(به صحت انتخاب تایپ ستون‌ها و کلید اصلی و کلید خارجی توجه کنید)

ب) ابتدا جدول StudentDetails را با ۲ ستون student\_id و description بسازید. سپس محتوای جدول را به شکلی تکمیل کنید که ستون student\_id حاوی شماره دانشجویی دانشجویانی که قبل از سال 1395 وارد دانشگاه شده و هنوز فارق التحصیل نشده‌اند باشد و ستون description به طور کامل با کلمه "با تجربه" پر شود (اسلاید ۱۵ و ۵۹).

پ) سطرهای از جدول StudentCourseRelations که قسمت نمره برای آن سطر خالی است، با میانگین کل نمراتی که تاکنون در StudentCourseRelations وارد شده بروزرسانی شود (اسلاید ۶۰).

ث) دستوری بنویسید که سطرهایی از جدول StudentCourseRelations که نمره‌ای در آن وارد نشده است را با میانگین کل نمراتی که تاکنون برای همان درس (اسلاید ۲۰) در StudentCourseRelations وارد شده جایگزین کند.

ج) با شرایط زیر جدول StudentCourseRelations را بروز کنید (اسلاید ۶۲).

- نمرات وارد نشده را با میانگین کل نمرات همان درس جایگزین کنید.
- نمرات کمتر از ۱۰ را به ۱۰ تغییر دهید.
- همه نمرات درس پایگاه داده را به اندازه ۱۰ درصد نمره افزایش دهید.
- در نمراتی که هیچ یک از شرایط بالا را ندارند تغییری ایجاد نکنید.

## سوالات اختیاری (۳۰)

۱) بعنوان یک مهندس داده اگر پروژه اصلاح دیتامدل ارائه شده به شما محول شود، نظر شما در رابطه با این ساختار چیست؟ معایب یا مزایای این طراحی را بیان کنید؟ در صورت وجود معایب به نظر شما چه تغییر یا تغییراتی در ساختار این طراحی لازم است؟ (این سوال اختیاری است و صرف نظر از کیفیت جواب، به شما نمره تعلق خواهد گرفت.)

۲) دستوری بنویسید که نام، نام خانوادگی، وضعیت فارق التحصیلی و رشته تحصیلی همه دانشجویانی که تاکنون هیچ نمره ای برای آنها ثبت نشده است یا نمره آنها همه اعداد بین ۰ تا ۱۵ بجز ۱۱ باشد را برگرداند (اسلاید ۱۹ و ۱۷ و ۲۷ و ۳۰).

۳) شماره دانشجویی و نام دانشجویانی که همه دروس بجز درسی با course\_id = 4 را اخذ کردند (اسلاید ۴۸).

۴) تعداد کل دانشجویانی که میانگین نمرات آنها بیشتر از میانگین کل نمراتی که تاکنون ثبت شده است را نمایش دهید.

۵) گزارشی تهیه کنید که در یک جدول تعداد کل دانشجویان با عنوان "total"، تعداد دانشجویان مشغول به تحصیل با عنوان "notGraduated" و تعداد دانشجویانی که میانگین نمرات آنها بیشتر از میانگین کل نمراتی که تاکنون ثبت شده است با عنوان "top" را نمایش دهد.

۶) اگر سال ورود به دانشگاه برای یک دانشجو، بعد از (بزرگتر از) سال وارد شدن اولین نمره برای همان دانشجو در دیتابیس باشد، به این معنی است که دانشجو قبل از وارد شدن به دانشگاه در درسی ثبتنام کرده و این داده نامعتبر است. با دستوری مناسب این داده‌های نامعتبر را از جدول حذف کنید (اسلاید ۵۶ و ۲۳ و ۲۴).  
راهنمایی:

```
where substring(student_id::varchar, 1, 3) > substring(year:: varchar, 2, 4);
```