

**دانشکده ریاضی و علوم کامپیوتر**

**برنامه‌سازی پیشرفته و کارگاه**

**دیتابیس مقدماتی 2**

استاد درس

دکتر مهدی قطعی

استاد دوم

بهنام یوسفی مهر

نگارش

محمدحسین هاشمی، امین رضایی‌مهر و آرمان حسینی

بهار ۱۴۰4

**فهرست**

[update 4](#_Toc198572337)

[order by 5](#_Toc198572338)

[مرتبش کن 5](#_Toc198572339)

[desc/asc 5](#_Toc198572340)

[multiple column ordering 6](#_Toc198572341)

[با این همه ردیف چیکار کنم ؟ 7](#_Toc198572342)

[aliases 7](#_Toc198572343)

[group by 8](#_Toc198572344)

[group by on multiple columns 9](#_Toc198572345)

[having 10](#_Toc198572346)

یه دستور جدید از طرف رئیس اومده. اون وسواس داره و دوست نداره که هیچ دوتا تسکی دستمزد برابر داشته باشن. چجوری می‌تونیم مقدار دستمزد هر تسک رو آپدیت کنیم ؟

update

اگه یادتون باشه یکم قبل تر برای اضافه کردن دستمزد ها دستور update رو دیدید. خیلی خلاصه این دستور کار تغییر مقدار record هارو برامون انجام می‌ده. کافیه اسم جدولت رو بهش بدی و بگی که کجا چه مقداری رو بذار.

بیاید با کوئری زیر کار رئیس رو راه بندازیم بعد یکم بیشتر در مورد این دستور صحبت می‌کنیم:

update tasks  
set cost = cost + (id - 1)\* 5  
where id <= 3;  
update tasks  
set cost = cost - id \* 5  
where id > 2 and id <= 5;  
update tasks  
set cost = cost - (id - 6) \* 5  
where id > 5;

الان اگه یه نگاه به جدولمون بندازیم می‌بینیم که رکورد های دستمزد با مقادیر دلخواه ما پر شدن.

A screen shot of a computer

AI-generated content may be incorrect.

اگه توی یکی از این آپدیت ها یادمون میرفت که از where استفاده کنیم چی می‌شد؟ دقیقا، کل جدول رو خراب می‌کردیم. جدول الانمون یه جدول کوچیکه و شاید خراب شدنش مسئله خیلی بزرگی نباشه ولی اگه شما یه روز توی یه شرکتی مثل اسنپ کار کنید اصلا دلتون نمیخواد که موجودی همه کاربر ها رو تغییر بدید (شایدم میخواد :D). البته که شرکت ها به این راحتی ها اجازه تغییر دیتابیسشون رو نمی‌دن ولی شما به عنوان برنامه‌نویس همیشه باید حواستون به اینجور خطاها باشه و به شرط هایی که برای دستوراتی مثل delete یا update می‌ذارید دقت کنید.

order by

تا این لحظه احتمالا موقع استفاده از دستور select به هر شکلی کاری به ترتیب row های خروجی مون نداشتیم. ولی خب اگه یکم بهش فکر کرده باشید براتون سوال شده که چجوری می‌تونیم ترتیب خروجی کوئریمون رو عوض کنیم یا اصلا ترتیب دیفالتی که دستور seletبهمون خروجی میده به چه صورته.

مرتبش کن

کوئری زیر رو روی دیتابیستون اجرا کنید :

select \* from tasks order by due\_date;

می‌بینید که خروجی بر حسب due\_date و به صورت نزولی مرتب شده:

A screen shot of a computer

AI-generated content may be incorrect.

حالا اگه بخوایم نزولی یا صعودی بودن خروجی رو مشخص کنیم چی ؟

desc/asc

دیتابیس شما باهوشه، کافیه بهش ترتیب مورد نظرتون رو بدید تا انجامش بده. بیاید برای صعودی کردن ترتیب دستور زیر رو اجرا کنیم :

select \* from tasks order by due\_date desc;

همینطور که دیدید خروجی دقیقا چیزی شد که می‌خواستیم :

A screenshot of a computer

AI-generated content may be incorrect.

ولی هنوز یه مشکلی هست، ما نمی‌دونیم فیلد هایی که due\_date یکسانی دارن چجوری مرتب شدن. جواب این سوال خیلی سادست، اگه ما چیزی رو مشخص نکنیم دیتابیس اون مقادیر رو به ترتیب اضافه شدنشون به دیتابیس نشون میده. حالا بیاید به این مسئله فکر کنید، این سناریو رو در نظر بگیرید که ما نیاز داریم بین تسک هایی که ددلاینشون مشترکه تسک با بیشترین دستمزد رو ببینیم برای این باید چیکار کنیم ؟

multiple column ordering

select \* from tasks  
order by due\_date desc, cost desc;

اگه کوئری بالا رو ارسال کنید می‌بینید که جدول دقیقا طبق خواسته ما مرتب شده :

A screenshot of a computer

AI-generated content may be incorrect.

شما می‌تونید هر چند تا ستون که لازم دارید رو برای معیار مرتب سازی قرار بدی و دیتابیس ما با اولویت از چپ به راست تسک های مارو طبق معیار هایی که دادیم مرتب میکنه. برای درک بهتر این ترتیب بیاید یه کوئری دیگه رو هم اجرا کنیم:

select \* from tasks  
order by cost desc, due\_date desc;

همین طور که تو عکس پایین میبینید خروجی این کوئری حتی شبیه به چیزی که میخواستیم هم نیست و تسک ها برخلاف خواسته ما طبق دستمزد مرتب شدن.

A screenshot of a computer program

AI-generated content may be incorrect.

با این همه ردیف چیکار کنم ؟

ایده todo-list ما گرفته و مشتری ها دارن ازش برای تسک های شرکتشون استفاده می‌کنن. ولی الان یه مشکلی هست، مشتری ها میگن که تعداد row های هر جدول خیلی زیاد شده و نمیتونن چشمی تفسیرشون کنن. اونا نیاز به یه سری ابزار دارن که باهاش بتونن اطلاعات مفید رو خلاصه تر ببینن. بیاید بگردیم تا راه حل مشکلشون رو پیدا کنیم.

select employee\_id, *count*(\*) as 'tasks\_num'  
from tasks group by employee\_id;

کوئری بالا رو اجرا کنید و بیاید خروجیش رو باهم بررسی کنیم :

A screenshot of a computer

AI-generated content may be incorrect.

قبل از اینکه ببینیم چه اتفاقی برای row هامون افتاده به column ها دقت کنید. ما که چیزی به اسم task\_num نداشتیم پس این column از کجا اومده ؟

aliases

دستور as (مخفف aliases) یه اسم موقت به ستونی که مشخص کردیم می‌ده. بعضی اوقات ستون های ما اسمی دارن که دوسش نداریم (طولانی یا نامفهومه) یا که مثل همین مثال با aggregate function ها ساخته شده و اسم نداره. با استفاده از این دستور ما می‌تونیم خروجی های قشنگ تری ایجاد کنیم که نه کاربر و نه خودمون رو گیج نکنه.

این دستور یه کاربرد دیگه هم داره. با استفاده از as می‌تونید با دادن یه اسم موقتی به یه ستون یا جدول کد های تمیز تر و خوانا تری بنویسید. برای بهتر فهمیدن این کاربرد یه نگاه به کد زیر بندازید :

select t.name, t.due\_date, s.name as 'step\_name', s.is\_completed as 'is\_done'  
from tasks as t, steps as s  
where s.task\_id = t.id;

این کوئری یه کوئری نسبتا کوتاه محسوب میشه و توی همین کوئری هم میبینید که دستور as چقدر از طولانی شدن کوئری جلوگیری می‌کنه و کار رو براتون راحت تر می‌کنه. البته یادتون نره که SQL هم قواعد کلین کد خودشو داره، یه وقت خرابکاری نکنید.

راستی دستور asرو به این شکل هم میشه نوشت اگه یه وقت به همچین چیزی برخوردید تعجب نکنید:

select t.name, t.due\_date, s.name 'step\_name', s.is\_completed 'is\_done'  
from tasks t, steps s  
where s.task\_id = t.id;

خیلی جاها می‌تونیم خود دستور as رو ننویسیم.

کاربرد این دستور هرچقدر که برید جلوتر بیشتر می‌شه و خودتون باهاش دوست می‌شید پس بیاید بیشتر از این معطل نشیم و ببینیم چه بلایی سر row هامون اومده بود.

A screenshot of a computer

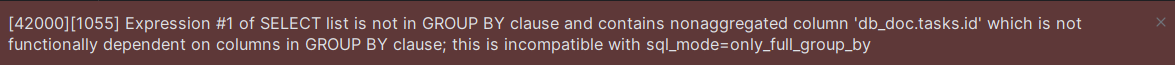
AI-generated content may be incorrect.

group by

این دستور خیلی ساده row های مارو برمی‌داره و اونایی که توی ستونی که مشخص کردیم مقدار یکسان دارنو توی یه ردیف خلاصه می‌کنه. بیاید یه کوئری دیگه بزنید تا بیشتر با group by آشنا بشید:

select \* from tasks group by employee\_id;

اوپس! کوئری ارسال نشد ولی چرا ؟



شاید خودتون هم بهش فکر کرده باشید که SQLچجوری می‌خواد بقیه ستون هارو که مقدار های متفاوتی دارن نشون بده و مشکل دقیقا همین جاست. ستون هایی مثل id، name و... هرکدوم برای خودشون یه سری مقدار دارن، شاید توی یه سناریو خاص بشه یه چیزی رو نشون داد که جای همه این مقادیرو بگیره ولی یه فرمول کلی وجود نداره که برای هرکدوم از گروه ها که با employee\_id مشخص شدن یه اسم یا ایدی بده که همشون رو رپرزنت کنه اروری که DataGripداده هم تقریبا داره همینو میگه.

الان شاید بپرسید اگه ما نمی‌تونیم ستون های دیگه رو ببینیم دقیقا چی می‌تونیم ببینیم؟ یکم قبل تر شما با aggregate function ها آشنا شدید. دستور group by بدون این توابع بیشتر کاربرد خودش رو از دست می‌ده. شما می‌تونید همونطور که توی کوئری اول دیدید تعداد اعضای هر گروه رو با تابع countببینید یا با تابع هایی مثل sum یا avg مجموع یا میانگین هر فیلد ستون عددی ای که خواستید رو نشون بدید یا حتی مقادیر max و min هر فیلدی که این توابع ساپورت می‌کنن رو نشون بدید. در نهایت شما می‌تونید با استفاده از این توابع و حتی توابع بیشتری که خودتون بعدا باهاشون آشنا می‌شید خلاصه کل دیتاتون رو داشته باشید و روش تحلیل انجام بدید بدون اینکه لازم باشه کل ردیف های جدولتون رو بررسی کنید.

حالا که با این دستور آشنا شدید بیاید تا باهم توی این دستور و کاربرد هاش بیشتر عمیق بشیم. اگه ما بخوایم در آمد روزانه هر کارمند رو داشته باشیم چی ؟ چجوری می‌تونیم از این دستور استفاده کنیم تا جمع در آمد هر روز رو برای همه کارمند هامون ببنیم ؟

group by on multiple columns

برای رفع این نیازمون باید تسک هامون رو بر اساس کارمند و تاریخ گروه بندی کنیم و بعد جمع هر گروه رو نشون بدیم. یکم بهش فکر کنید، ایده ای دارید ؟

select employee\_id,due\_date,*count*(\*) as 'tasks\_num',  
 *sum*(cost) as 'total income' from tasks  
group by employee\_id,due\_date  
order by `total income` desc;

همین طور که توی خروجی میبینید با ارسال این کوئری همون چیزی که خواستیم رو به ترتیب بیشترین درآمد روزانه از دیتابیس گرفتیم.

A screenshot of a computer

AI-generated content may be incorrect.

به نظرتون ترتیب گروه بندی روی این دستور هم تاثیر داره؟ بیاید باهم چکش کنیم:

select employee\_id,due\_date,*count*(\*) as 'tasks\_num',  
 *sum*(cost) as 'total income' from tasks  
group by due\_date,employee\_id  
order by `total income` desc;

خودتون دیدید که خروجی از نظر محتوا فرقی نکرد. احتمالا حدستون هم همین بوده چون در نهایت قرار بود که مقادیر یونیک گروه بندی بشن. ولی اشتباه می‌کنید. درسته که خروجی از نظر محتوای داده یکیه ولی ترتیب غلط ممکنه توی یه سری از دستور ها که از سطح کلاس AP خارجه به دردسر بندازتتون برای همین حداقل تا وقتی که با این دستورات و کاربرد هاشون آشنا نشدید دقت کنید که ترتیبی که می‌نویسید دقیقا همون چیزی باشه که مد نظر دارید.

راستی ما قبل تر برای فیلتر کردن دیتامون از where استفاده میکردیم. بیاید همین دستور رو استفاده کنیم و فقط حالتی رو نشون بدیم که اون کارمند درآمد روزانه بیشتر از 300 داره :

select employee\_id,due\_date,*count*(\*) as 'tasks\_num',  
 *sum*(cost) as 'total income' from tasks  
group by due\_date,employee\_id  
where `total income` > 300  
order by `total income` desc;

اوپس بازم ارور! این سری دیتاگریپ حتی کوئری مارو نمی‌فرسته و خودش داره بهمون میگه که کد ایراد داره. بیاید متن ارور رو بررسی کنیم:



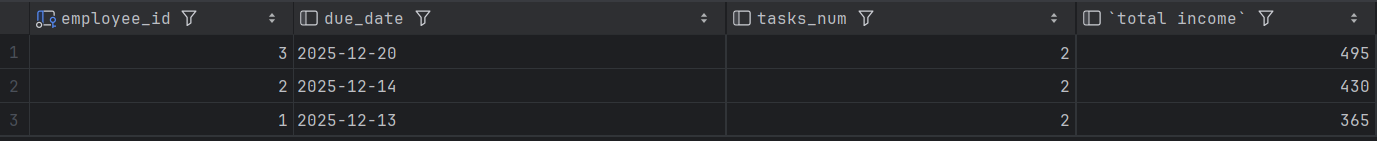
کلید واژه where توی این بخش از کوئری اصلا شناخته نمیشه. پس باید چیکار کنیم؟ درست حدس زدید، یه کلید واژه جایگزین وجود داره.

having

این دستور به ما اجازه میده که بعد از گروه کردن دیتامون بتونیم فیلترش کنیم. کارکرد having تقریبا مشابه where هستش و اکثرا وقتی استفاده میشه که ما بخوایم روی ستون های تولید شده توسط aggregate function ها فیلتر انجام بدیم.

select employee\_id,due\_date,*count*(\*) as 'tasks\_num',  
 *sum*(cost) as 'total income' from tasks  
group by due\_date,employee\_id  
having `total income` > 300  
order by `total income` desc;

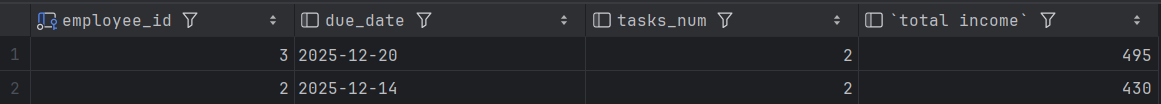
می‌بینید که با ارسال این کوئری خروجی مدنظرتون رو بدون هیچ اروری دریافت میکنید.



البته بهتره براتون روشن کنم این دستور جایگزین دستور where نیست. در اصل group by به SQL اضافه شده چون where، aggregate function ها رو نمی‌شناخت. ما هنوزم برای فیلتر کردن دیتامون قبل از گروه کردن **باید** از دستور where استفاده کنیم.

مثلا اگه لازم داشته باشیم که فقط کارمند هایی با ایدی بزرگتر از 1 رو نشون بدیم کوئری ارسالیمون باید به این شکل باشه :

select employee\_id,due\_date,*count*(\*) as 'tasks\_num',  
 *sum*(cost) as 'total income' from tasks  
where employee\_id > 1  
group by due\_date,employee\_id  
having `total income` > 300  
order by `total income` desc;



البته که برای این کار هم می‌تونید از having استفاده کنید ولی همونطوری که دیتاگریپ موقع ارسال کوئری زیر بهتون میگه این کار در صورتی که به aggregate function ها نیاز نداشته باشید بهینه نیست.

select employee\_id,due\_date,*count*(\*) as 'tasks\_num',  
 *sum*(cost) as 'total income' from tasks  
group by due\_date,employee\_id  
having employee\_id > 1  
order by `total income` desc;

کار ما تمومه ولی رئیس به نظر ثبات روانی خیلی خوبی نداره و میخواد به روند قبلی با حقوق ثابت برگرده.فعلا با کوئری زیر ستون cost رو حذف کنید تا کار رئیس راه بیوفته بعد تر درمورد تغییرات این مدلی روی دیتابیس بیشتر صحبت می‌کنیم.

alter table tasks  
drop column cost

قبل از این بیاید با یه دستور دیگه هم آشنا بشیم.