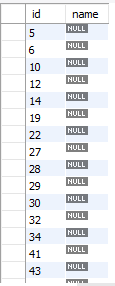
**گزارش**

در هر بخش در مورد خروجی query توضیحاتی داده شده است.

نکته: با توجه به اینکه جدول یا رابطه ای موجود نیست که به صورت self-referential باشد، برای همین از self-join استفاده نمی شود.

SelectionQuery1:



معلم هایی که درسی را تاکنون بر نداشته اند و باید درسی بردارند.

SelectionQuery2:

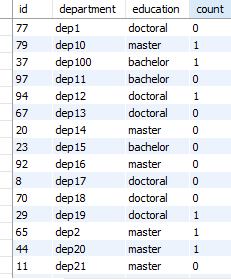


درس ها یا کورس هایی که معلم ندارند و باید معلمی داشته باشند که برای این query این داده را به جدول course اضافه کردیم:



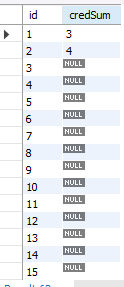
SelectionQuery3:

نشان می دهد که برای هر یک از کارشناسان آموزشی چند دانشجو با توجه به سطح تحصیلی کارشناسان موجود است و به وسیله این می فهمیم که کدام یک از کارشناسان آموزشی را باید اخراج کنیم چرا که به آن ها نیاز نداریم.

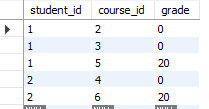


SelectionQuery4:

شناسه و مجموع واحد دانشجویانی که کمتر از ۵ واحد دارند یا اصلا واحدی را برنداشته اند(که در این صورت مجموع واحد ها برابر null خواهد بود) و باید واحد هایی را بردارند.

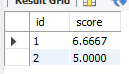


توجه شود که برای این query جدول رابطه و جدول takes ساخته شده است و داده های زیر به آن اضافه شده است:



SelectionQuery5:

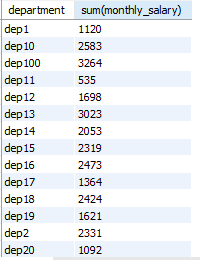
شناسه و معدل دانشجویان تاکنون که کمتر از ۱۰ می باشد و برای همین باید اخراج شوند.



توجه شود که نمره هر درس ضرب در تعداد واحد ها می شود و بعد درون جدول takes قرار می گیرد.

SelectionQuery6:

مجموع حقوق ماهانه کارکنان هر یک از دانشکده ها و می توان به کمک آن بودجه سالانه را تعیین کرد.



SelectionQuery7:

وسایل آزمایشگاهی که در آزمایشگاهی قرار ندارند و باید در آزمایشگاهی قرار بگیرند.



توجه شود که برای این query داده های زیر به LabInstrumentاضافه شده است:



SelectionQuery8:

مسئولین آزمایشگاهی که در آزمایشگاهی قرار ندارند و باید در آزمایشگاهی قرار بگیرند.



توجه شود که برای این query داده های زیر به LabDirectorاضافه شده است:



SelectionQuery9:

آزمایشگاه هایی که مسئول آزمایشگاه ندارند و باید مسئولی برای آن ها در نظر گرفته شود.



توجه شود که برای این query داده های زیر به Labاضافه شده است



SelectionQuery10:

زمان هایی که در آن ها دو یا چندین کلاس در یک مکان، سال و ترم برگزار می شوند و باید کلاس هایی را از این زمان ها جا به جا کرد.



توجه شود که برای این query داده های زیر به Coursesاضافه شده است:



SelectionQuery11:

زمان هایی که در آن ها دو یا چندین امتحان در یک مکان، سال و ترم برگزار می شوند و باید امتحان هایی را از این زمان ها جا به جا کرد.



توجه شود که برای این query داده های زیر به Courseاضافه شده است:



SelectionQuery12:

دانشجویانی که استاد راهنما ندارند و باید برای آن ها استاد راهنما مشخص کرد.



توجه شود که برای این query داده های زیر به studentاضافه شده است:



SelectionQuery13:

معلم هایی که بیش از چهار دانشجو را راهنمایی می کنند و باید یک یا چندین دانشجو را به معلم های دیگر داد.



SelectionQuery14:

دانشجویانی که پروژه ندارند و باید پروژه ای بردارند.



توجه شود که برای این query داده های زیر به studentاضافه شده است:



SelectionQuery15:

پروژه هایی که برای آن ها استاد ناظر مشخص نشده است و باید استاد ناظری برای آن ها مشخص شود.



توجه شود که برای این query داده های زیر به projectاضافه شده است:



Triggers

نکته: توجه شود رابطه های registers/ deregistersعملا همان رابطه Takes می باشند و در واقع نیازی به جدول هایی برای آن ها نمی باشد زیرا رابطه به صورت عملی می باشد و در واقع کارشناسان آموزشی یک ردیفی در جدول Takes قرار می دهند.

نکته: deleteTrigger1 و deleteTrigger2 برای رابطه takes می باشند.

نکته: برای پیاده سازی کاربران و فعال شدن trigger ای به نام UsernameTrigger از رابطه ای به نام users استفاده می شود که هنگامی که کاربری username موجود در این رابطه را تغییر می دهد، به وسیله trigger ردیفی در UsernameChanges تولید شده و به وسیله‌ آن کنترل می شود که کاربر در هر روز پیش از دو بار نمی تواند username خود را تغییر دهد.

نکته: توجه شود که فرض شده است که در سیستم trigger دیگری موجود می باشد که جلوگیری از update ای می کند که id کاربر یا users را تغییر می دهد.

نکته: نام کاربری هر یک از کاربران پایگاه داده به صورت اتوماتیک می باشد و در واقع همان id آن ها می باشد و برای همین امکان ندارد که آن را تغییر دهند. برای مثال نام کاربری پورتال همان شماره دانشجویی یا id می باشد که قابل تغییر نمی باشد. همچنین id هر یک از کاربران در یکی از جداول مربوط به خودشان و نه در همه جدول ها unique می باشد. برای حل این مشکل یک entity به نام  users ساخته می شود که هر ردیف آن به یکی از کاربران map می شود و برای یکی از کاربران می باشد و شامل username ای می باشد که قابل تغییر هست. همچنین تغییرات در جدول `UsernameChanges` ذخیره می شوند. توجه شود که field یا ویژگی entity مشخص می کنید که ردیف مربوط به کدام entity است.