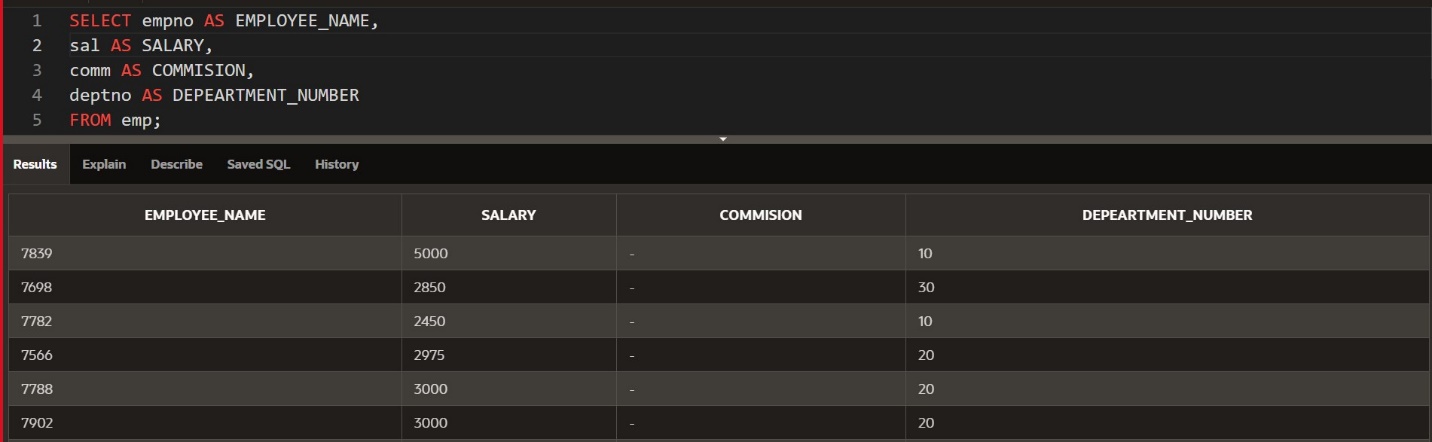
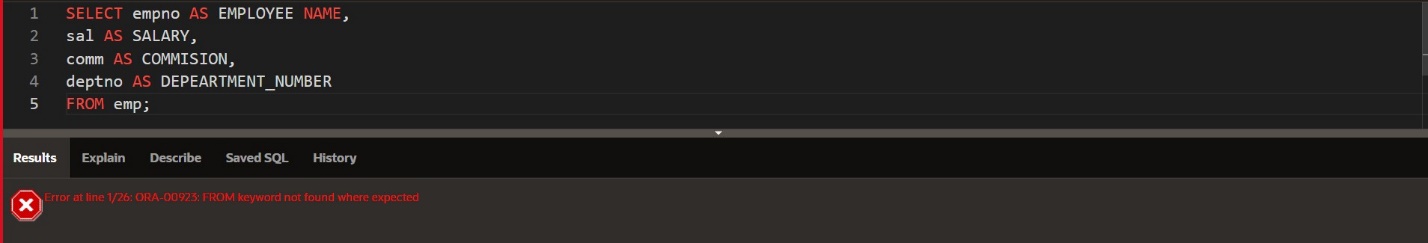
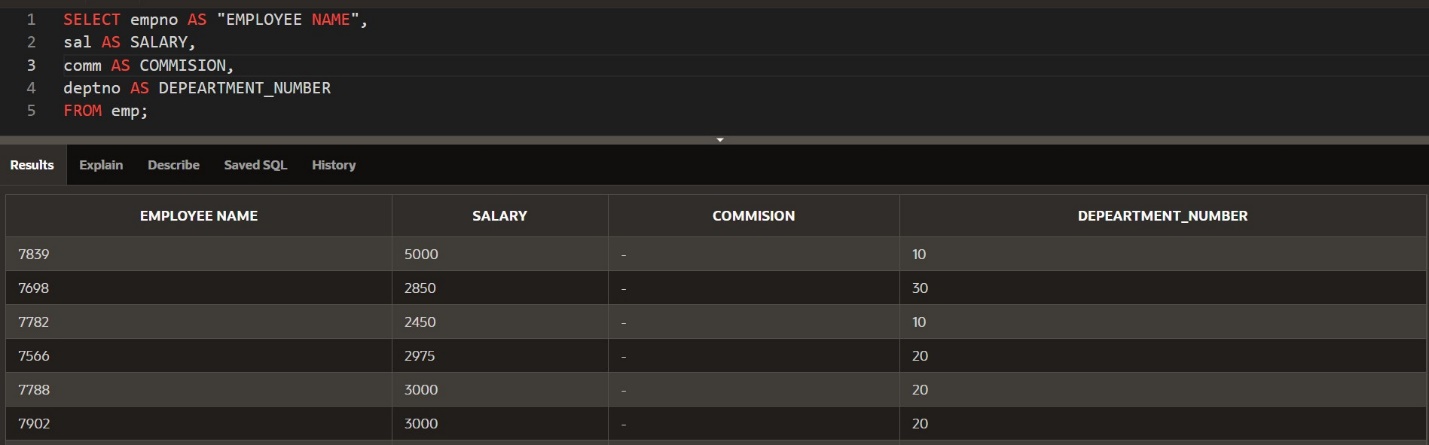
1. Bur dərslikdə biz **SQL** də column və yaxud table-lara özümüz tərəfindən **alislar** vasitəsi ilə adlar verməyi öyrənmiş olacayıq. Həmçinin bu dərslikdə concitanation əməliyyatını öyrənmiş olacayıq.
2. Bəzi hallar olur ki, table-da elə sütunlar olur ki, onların mənası olmur, yəni həmən column-u başa düşmək olmur ki, bu nə deməkdir?. Məhz bu kimi hallarda biz **aliaslardan** istifadə edərək həmən mənası olmayan column-ları mənalı hala gətiririk. Məsələn fərz edin ki, qarşı tərəfə biz elə bir report üçün **SQL** yazmağımız gərəkdir ki, orada olan column adlarını qarşı tərəf də rahat bir şəkildə anlaya bilsin. Məsələn aşağıdakı table-da görəcəyiniz olan **comm, sal, deptno** və **empno** kimi sütunlar çox mənasızdır.

Gördüyünüz kimi yuxarıdakı şəkildə altından qırmızı ilə qeyd olunan column adları başa düşülməsi çətindir. İndi isə gəlin bu mənası olmayan column-lara mənalı adları necə veririk aşağıdakı şəkildə olan query-ə baxaq.



Gördüyümüz kimi, az öncəki column adlarından fərqli olaraq indi column adlarımız rahat anlaşılır şəkildədir.

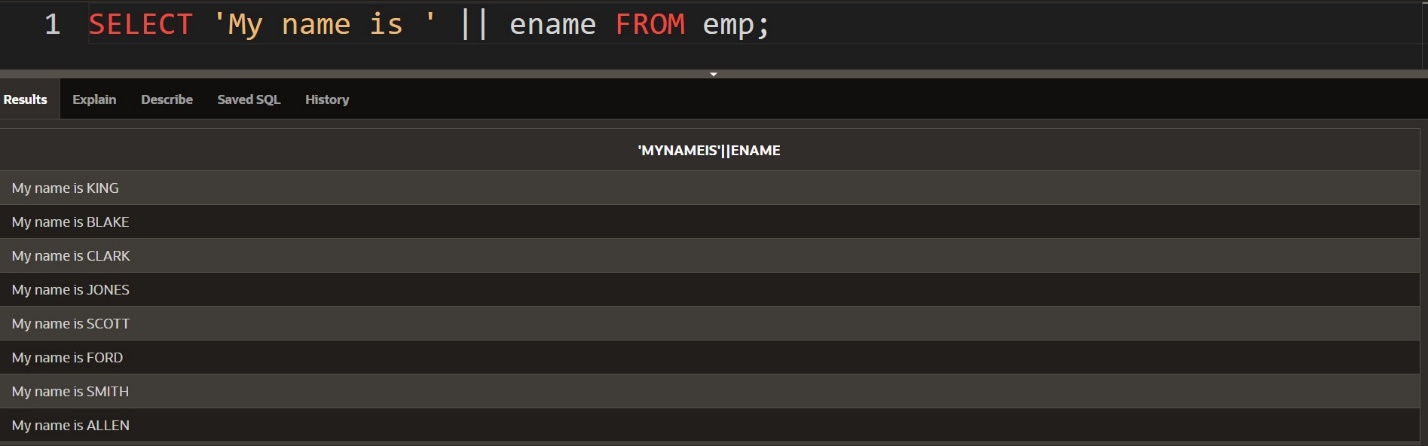
1. Daha sonra ola bilər ki, siz **aliaslar** təyin edərkən column adlarını aralarında boşluq qoyaraq etmək istəyərsiniz, bu zaman aşağıdakı kimi **alias** elan etsəniz siz xəta ilə qarşılaşacaqsınız, eynən aşağıdakı şəkildə göstərildiyi kimi.

Gördüyümüz kimi yuxarıdakı query-ni çalışdırdıqda burada xəta ilə qarşılaşdıq, buna səbəb **EMPLOYEE NAME** aralıq qoyaraq **alias** etmək istədik. Bu problemi aradan qaldırmaq üçün aşağıdakı şəkildə olan query-ə diqqət yetirmək lazımdır.

Gördüyünüz kimi, yuxarıdakı şəkildə olan **“EMPLOYEE NAME”** alias kimi elan olunduqda heç bir problem olmamamq şərti ilə aralarında boşluq olmaq ilə column adı qeyd olunur.

* **NOTE:** Əgər biz aliaslardan istifadə edib column adları vermək istəyirik isə bu zaman **“ “ (qoşa dırnaq)** istifadə edəcəyik, ancaq comparsion əməliyyatlarında isə fikir vermişdiniz isə **‘ ‘ (tək dırnaq)** istifadə etdik.
* Aliaslar istifadə edərkən **AS** keyword-ü optional-dır, yazsanızda olar yazmasanızda.

**CONCATİONATİON**

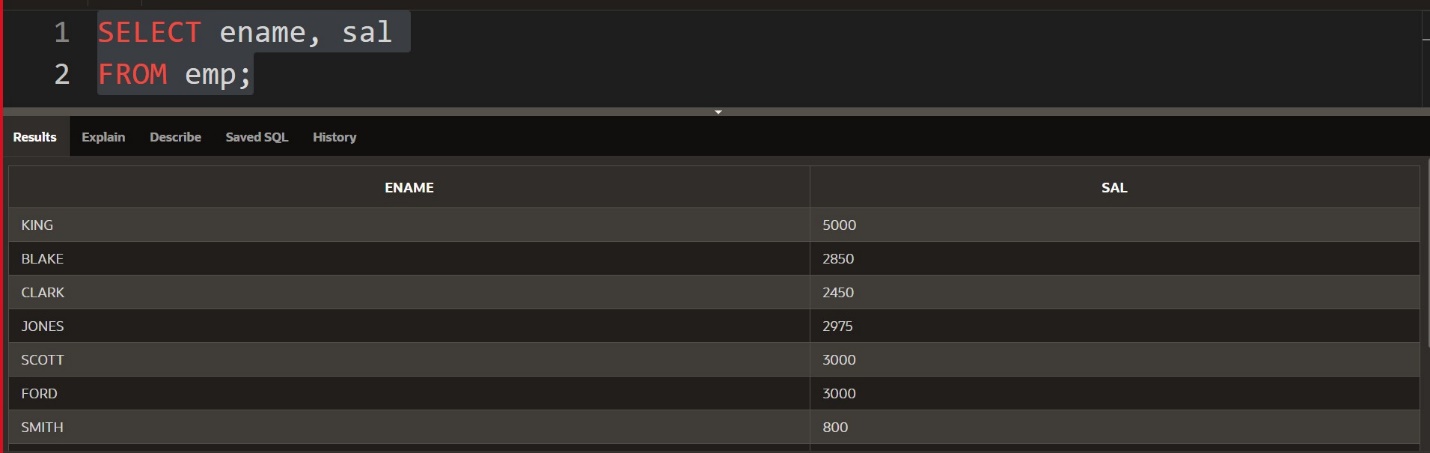
1. Əgər biz column dəyəri ilə birlikdə static bir dəyəri də biri birinə birləşdirib result-a çıxartmaq istəyirik isə bu zaman **concationation** əməliyyatından istifadə etməliyik. **SQL** də **concationation ||** (pipe) operatorunun vasitəsi ilə həyata keçirdilir. Məsələn, gəlin aşağıdakı şəkildə olan query-dən anlaya bilərik.

gördüyünüz kimi yuxarıdakı şəkildə olan query də biz static olan ‘My name is ‘ sözünü **ename** column da olan hər bir dəyərlə birləşdirib ortaya belə bir nəticə çıxartdıq.

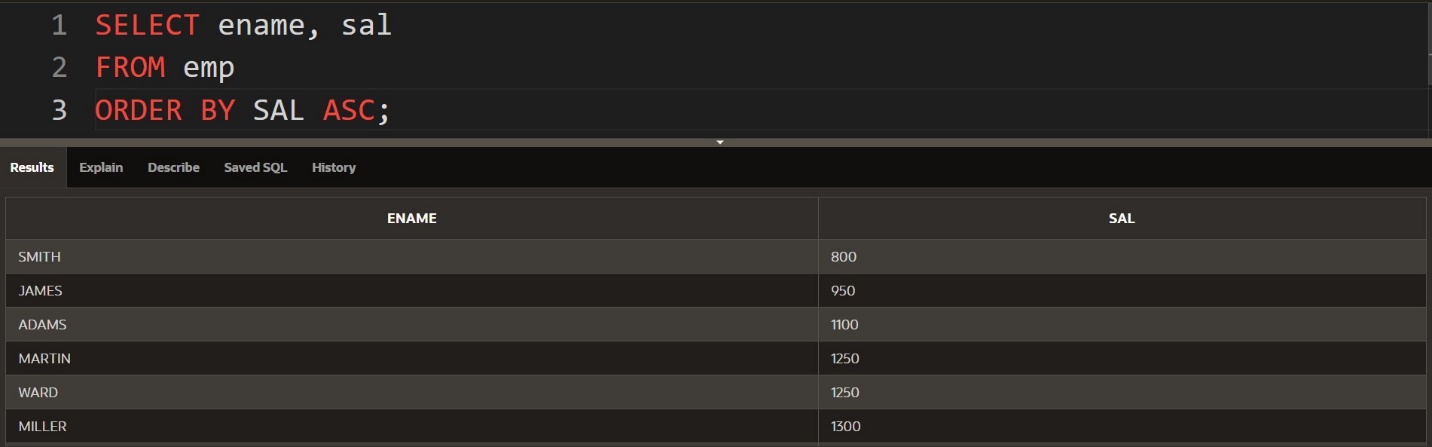
**ORDER BY nədir?**

1. **SQL** də ORDER BY clause-ı sətirləri verilmiş **sütuna** görə sıralamaq işinə baxır.

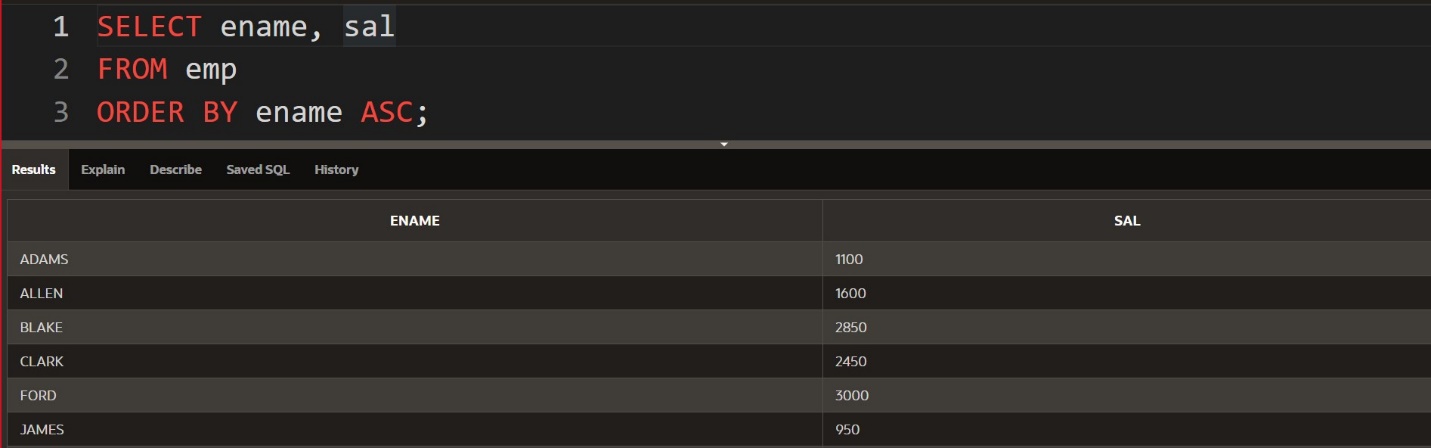
* **NOTE: SQL** default olaraq garanti vermir sizə dataların sıralı şəkildə çıxmasını, o defualt olaraq sırasız şəkildə sizin qarşınıza **resultset** çıxardır.

1. Məsələn fərz edin ki, bizim aşağıdakı şəkildə olan bir **resultset** vardır.

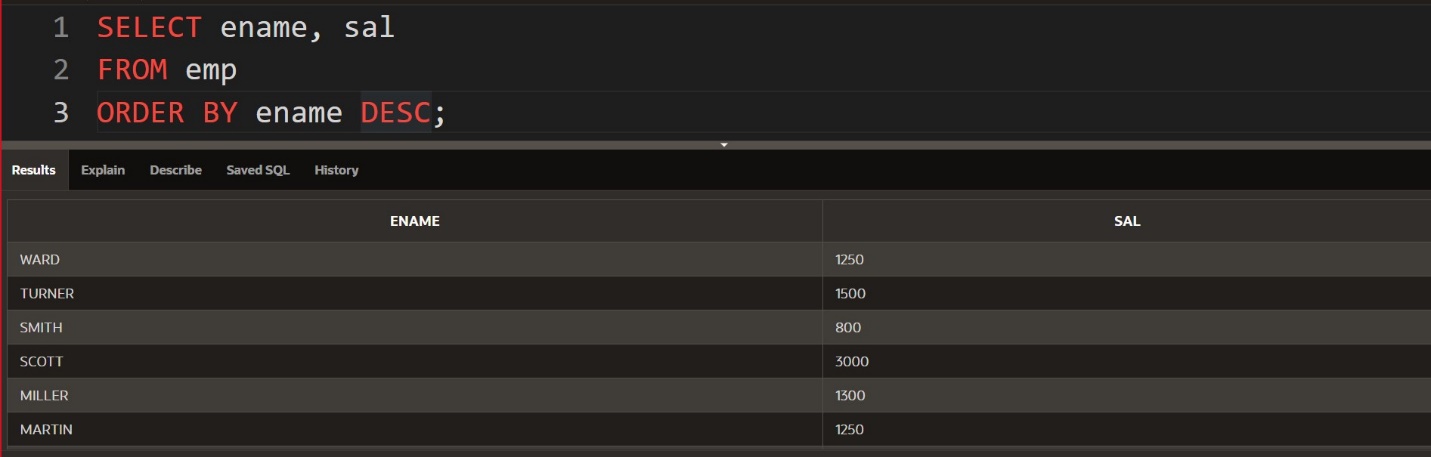
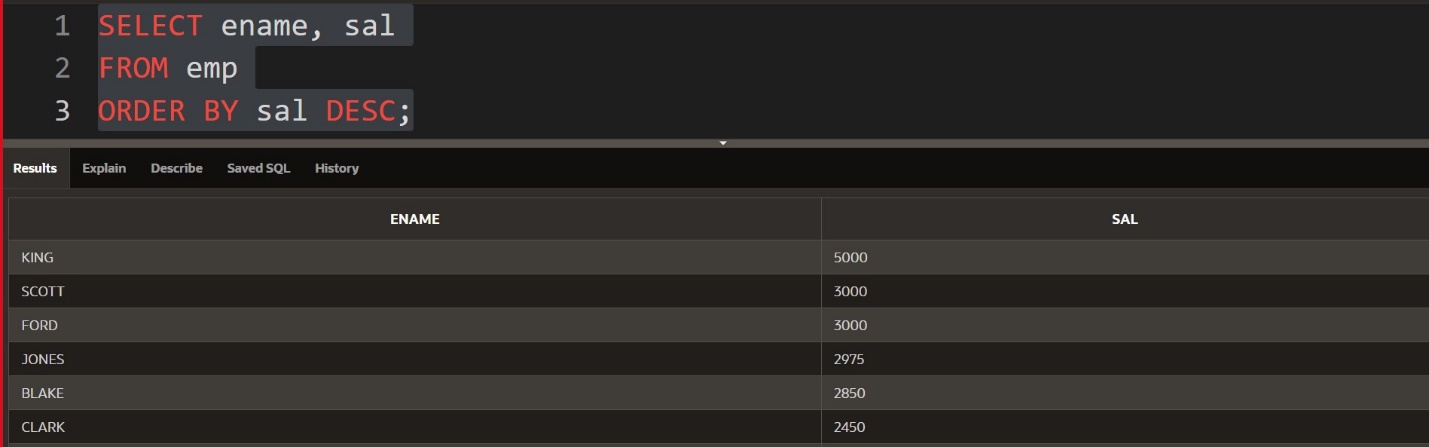
Gördüyünüz kimi yuxarıdakı şəkildə olan query-dən qayıdan **resultset** də həm **ename** və həmdə **sal** sütununda olan dəyərlər sırasız şəkildədir.

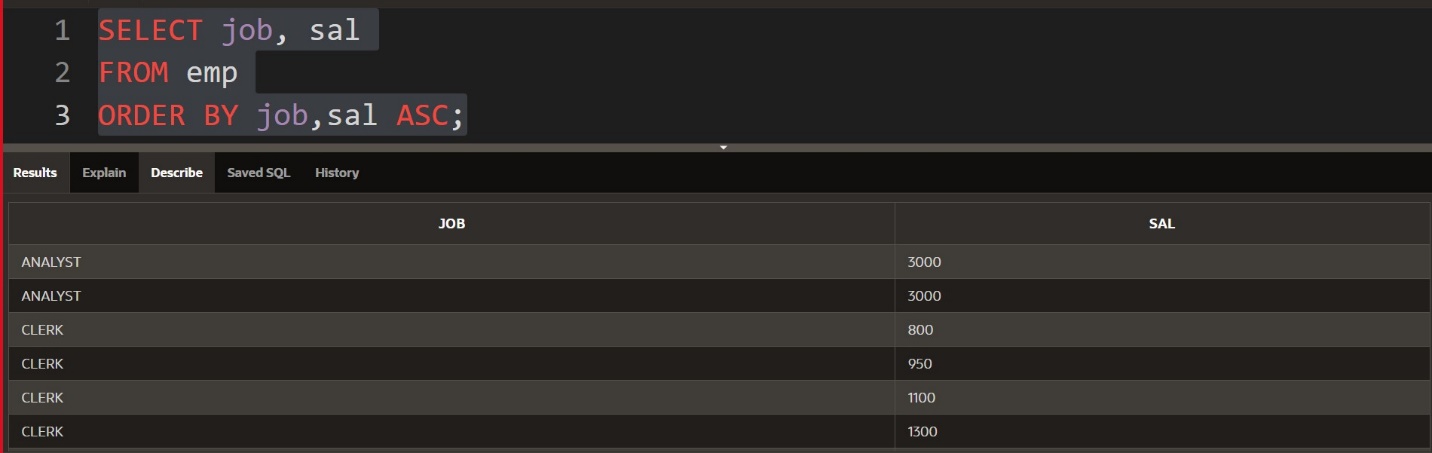
1. Gəlin indi isə yuxarıdakı şəkildə olan qarışıq sıralı şəkildə olan dataları sıralı şəkildə etməyə çalışaq aşağıdakı şəkildə olan query vasitəsi ilə.

Gördüyünüz kimi, yuxarıdaı şəkildə olan **resultset** də artıq **sal** column dakı dəyərlər sıralı şəkildə azdan çoxa doğru sıralanmışdır. Həmçinin biz **ename** sütununa görə də dataları sıralı şəkildə lakin budəfə rəqəmlərdən fərqli olaraq **A-Z-ə** doğru sıralayacaqdır aşağıdakı şəkildə olan query vasitəsi ilə.

gördüyünüz kimi bu dəfə isə **ename** sütunundakı dəyərlər **A-Z-ə** doğru sıralanmışdır.

* **NOTE: ORDER BY** clause-ı default olaraq **ASC** şəklində sıralayır dataları. Yazsanızda olar yazmasanızda.

1. Həmçinin biz dataları **çoxdan-aza** doğru da sıralaya bilərik və bunu **DESC** keyword-ünün vasitəsi ilə həyata keçirdirik , aşağıdakı şəkildən buna baxa bilərik.
2. Həmçinin biz bir çox sütuna görə **ORDER BY** clause-ından istifadə edə bilərik və bunu aşağıdakı şəkildə olan query-dən görə bilərik.



Gördüyünüz kimi, bir çox sütuna görə **ASC** şəklində **ORDER BY** əməliyyatını həyata keçirtmiş olduq. Lakin burada sizin üçün qaranlıq qalan məqam ola bilər ki, **ASC** şəklində **ORDER BY** etməyimizə baxmıyaraq **SAL** sütunun da yenədə **mix** formatda datalar çıxmışdır, bunun olmasına səbəb odur ki, **ORDER BY** clause-ı bir çox sütuna görə sıralama əməliyyatını həyata keçirtdikdə proses bu cürnə başa verir. O ilk öncə ilk sütuna görə **ORDER BY** əməliyyatını həyata keçirdir, daha sonra həyata keçirtdiyi ilk sütun üzərində olan dəyər digər sətirdədə olacaqdırsa, bu zaman növbəti sütuna görə sıralama əməliyyatını həyata keçirdəcəkdir.