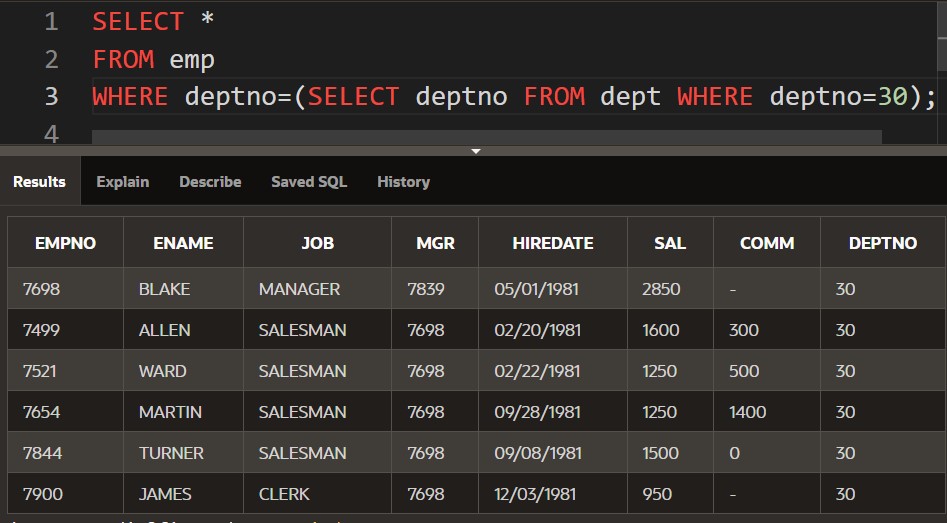
1. Əvvəlki dərsliklərdə **SQL** də basic query-lər yazaraq , orada biz **WHERE, ORDER BY, LIKE, IN GROUP BY, HAVING, MAX(),MIN(),COUNT(),SUM(),AVG(),NULLIF(),NVL(), CONCAT()** və s. **keyword**-lərdən istifadə edərək, basic bir şəkildə,yalnızca bir table üzərində query yazıb işlətmişdik. İndi isə artıq daha more complex query-lər yazaraq, yəni bir table-dan deyildə, birdən çox table üzərində **SELECT** query-si yazaraq onu öyrənmiş olacayıq.
2. Bu dərslikdə biz, əsasən **SQL** də **inner-query** nədir onu öyrənməyə çalışacayıq. Biz sizinlə birlikdə **SELECT** query-sini yazıb işə saldığımız example-lar və challange-lar olmuşdu və biz bilirdik ki, **select** query-si bizə table-dan dataların gətirilməsi üçün istifadə olunur. Burada isə artıq yeni bir query növü olan **SUB QUERY**-ni öyrənmiş olacayıq.

**SUB QUERY nədir?**

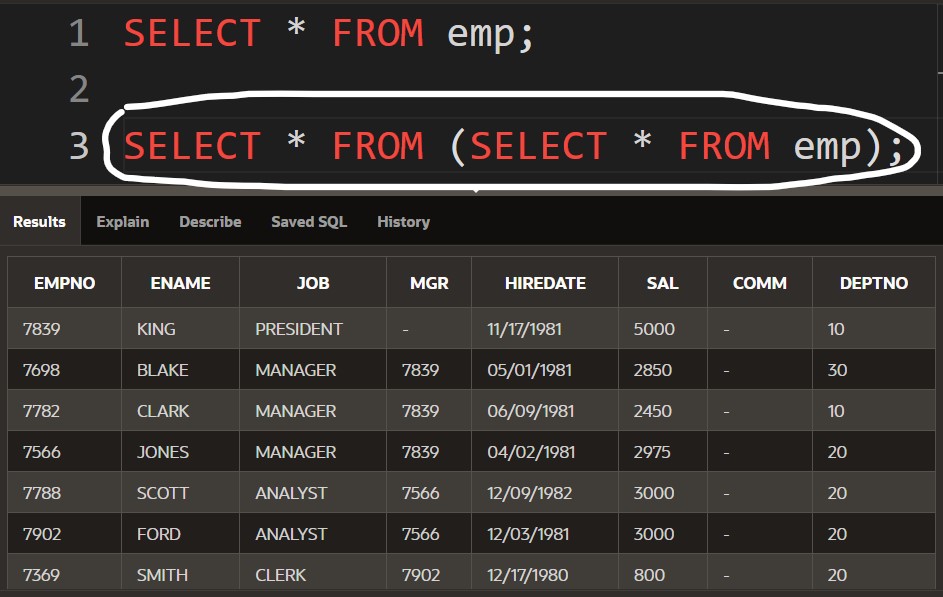
1. Sadə deyimlə desək, bir **select** query-sinin içində başqa bir **select** query-si yazılıbsa bu adlanır **SUB QUERY**.
2. Elə isə gəlin bir dənə **SUB QUERY** misalına baxaq, eynən aşağıdakı şəkildə olduğu kimi.

gördüyünüz kimi biz, yuxarıdakı şəkildə **WHERE** clause-ında başqa bir **SELECT** query-si yazmış olduq, bax bu hissə adlanır **SUB QUERY**.

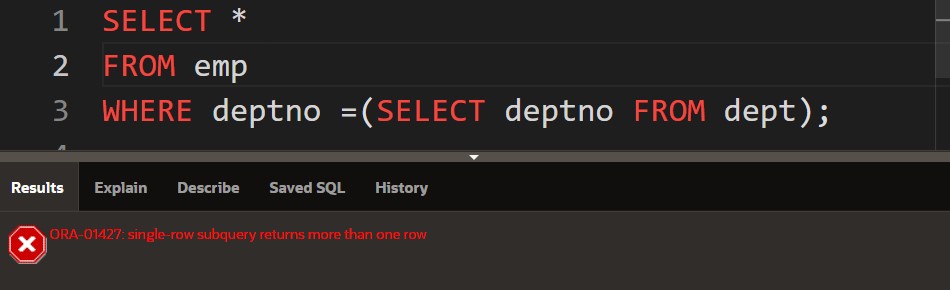
1. **SQL** də əgər bir **SUB QUERY** yazılıbsa ilk işə düşən hissə o olacaqdır, daha sonrasında isə həmən **SUB QUERY**-dən qayıdan nəticə nədirsə **outher query** tərəfindən istifadə olunacaqdır. Bunun işlənmə prosesini başa düşmək istəyirsinizsə, ilk öncə **SUB QUERY** hissəsini ayrılıqda çalışdırıb işə salsanız, eynən aşağıdakı şəkildə olduğu kimi oradan qayıdan bir **resultset** əldə edəcəksiniz, eynən aşağıdakı şəkildə olduğu kimi.

gördüyünüz kimi, yuxarıdakı şəkildə olan query də **SUB QUERY** hissəsini çalışdırdım və **resultsetdə** bir dəyər əldə etdim, bu dəyər daha sonrasında **WHERE** clause-ında **deptno** sütununa görə filterlama prosesindən keçərək bizə ən birinci şəkildə göstərmiş olduğum **resultset**-içıxardır qarşımıza.

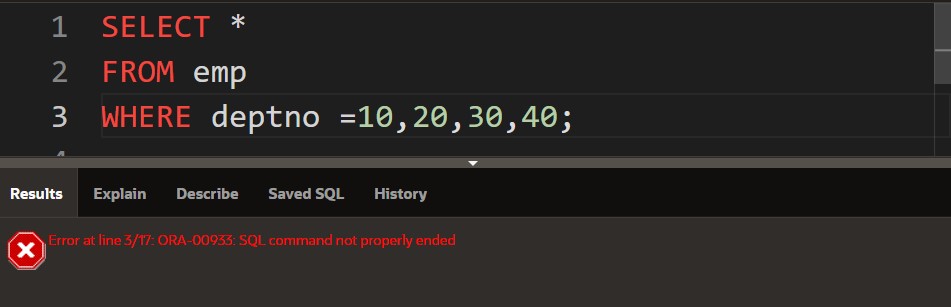
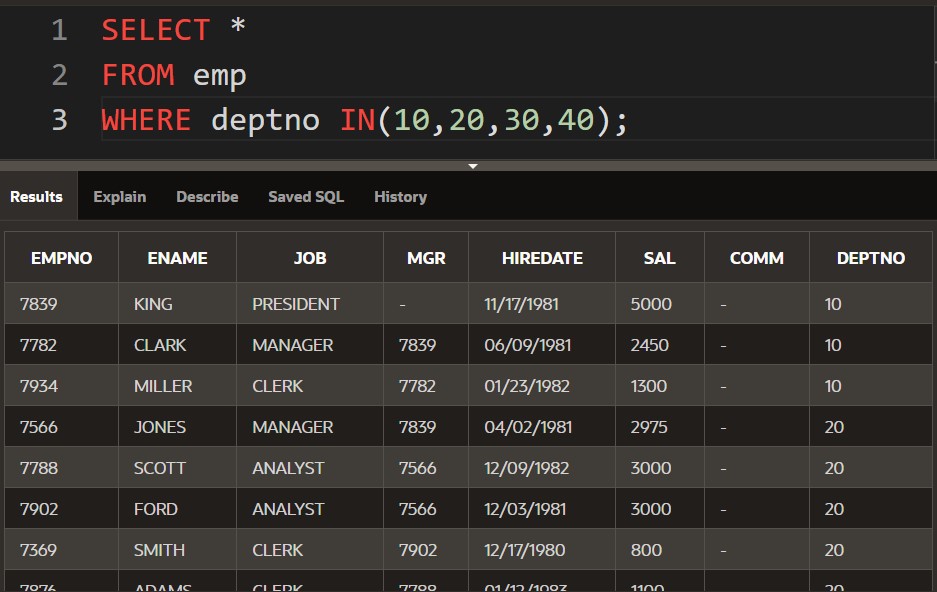
1. Yuxarıda biz **sub query**-nin **WHERE** clause-ında istifadə qaydasını gördük, indi isə gəlin **sub query**-in **FROM** clause-ında istifadə qaydasını görək, , eynən aşağıdakı şəkildə olduğu kimi.

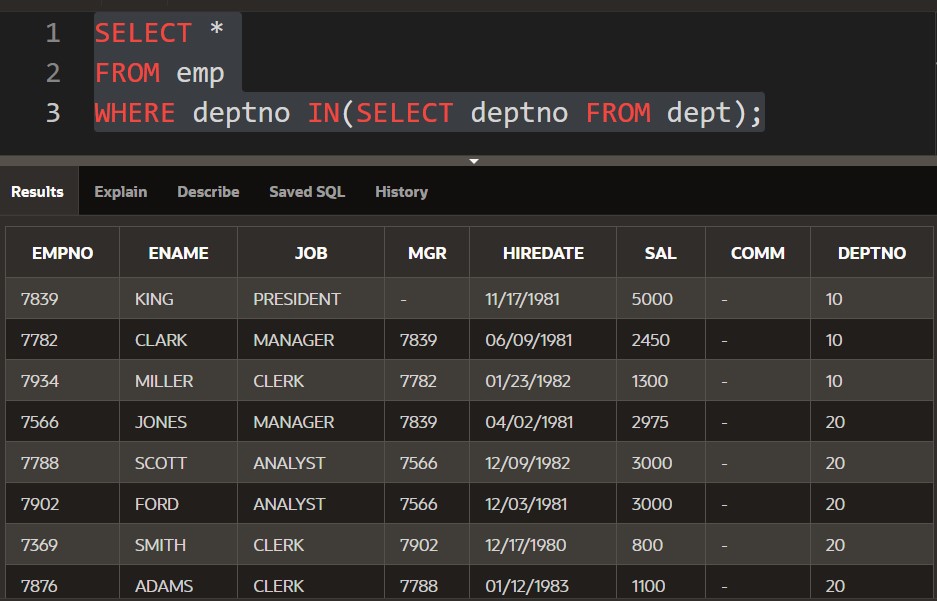
gördüyünüz kimi, yuxarıdakı şəkildə altından ağ rəng ilə işarələdiyim query bu dəfə isə **FROM** clause-ında istifadə olunan **sub query**-dir və bu query özündən yuxarıdakı query-in birə-bir gördüyü eyni işi görməkdədir. Bəs siz sual verə bilərsiniz ki, necə olduki biz **FROM** clause-ında heç bir table adı qeyd etmədən **resultset** əldə etdik, hansıki biz bilirik axı, **FROM** clause-ından sonra table adı gəlməlidir, ancaq gördüyünüz kimi belə vəziyyətdə də **resultset** əldə etmiş olduq. İndi isə gəlin bunun işlənmə prinsipini anlamağa çalışaq. Bu belə işləyir ki, **sub query**-ni siz ayrılıqda çalışdırıb işə salsanız görəcəksiniz ki, bir **resultset** əldə edirsiniz və **SQL** də **resultset**-in özü belə bir reserved olunmuş xüsusi **table** olduğuna görə **SQL** artıq onu **FROM** clause-ından sonra gələn bir **table** olaraq qəbul etdiyindən yəni onu bir **emp** table-ı olaraq qəbul edir və beləliklə siz sanki bir növ **emp** table-ından **resultset** əldə etmiş olursunuz. Həmçinin **FROM** hissəsində yazdığımız **sub query** bizim özmüz tərəfindən yaradılan table olduğuna görə ona spesific sütunlarda verərək **outher query** də həmən **FROM** clause-ında elan etdiyimiz **sub query**-dəki sütunları əldə edə bilərik, eynən aşağıdakı şəkildə olduğu kimi.

gördüyünüz kimi yuxarıdakı query də biz **FROM** hissəsində **subquery** istifadə edərək **SELECT** hissəsində özmüz tərəfindən elan etdiyimiz sütun adlarını **aliaslar** vasitəsi ilə verərək **outher query** daxilində məhz həmən sütunlara görə məlumatları çəkmiş olduq.

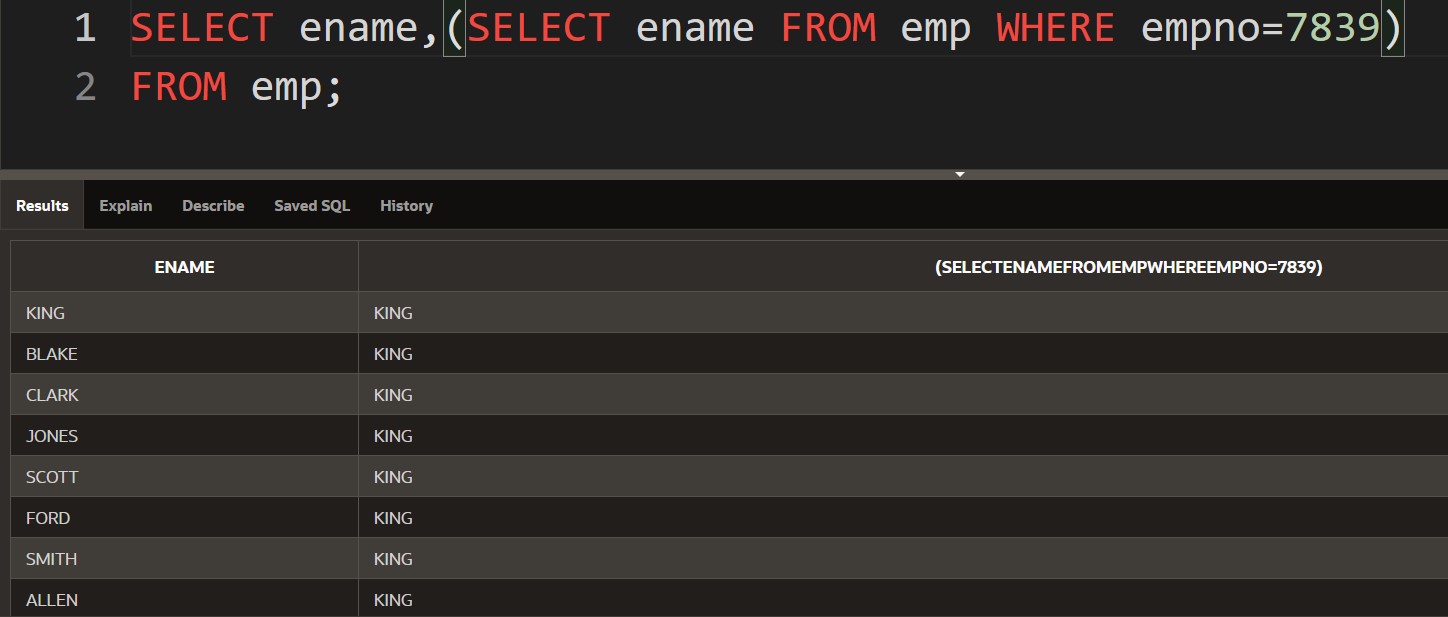
1. Bəzi hallarda **WHERE** clause-ında **sub query** işlətdiyimiz zaman xətalar ilə qarşılaşırıq, eynən aşağıdakı şəkildə olduğu kimi.

bu xətanın olmasına səbəb, bizim **sub query**-miz birdən çox **record** qaytardığına görə və biz də bilirik ki, **WHERE** clause-ında eyni anda bir sətirdə birdən çox sütun dəyərinə görə **filterlama** əməliyyatı aparılmadığından bu xəta ilə qarşılaşırıq, eynən aşağıdakı şəkildə olduğu kimi.

bu problemi həll edə bilmək üçün biz **SQL** də **IN** operatorundan istifadə edirdik, eynən aşağıdakı şəkildə olduğu kimi.

gördüyünüz kimi, bu problemi **SQL** da **IN** operatorundan istifadə edərək həll etmiş olduq. İndi isə bunun birə-bir eynisini biz az öncə yuxarıdakı şəkildə **sub query** də yaşamış olduğumuz problemi həll etməyə çalışaq, eynən aşağıdakı şəkildə olduğu kimi.

* **NOTE:** Qısası əgər **sub query** birdən çox record qaytaracaqdırsa, bu zaman həmən **sub query**-ni **WHERE** clause-ında problemsiz şəkildə istifadə olunmasını istəyirsənə, bu zaman **IN** operatorundan istifadə etməlisən.

1. İndi isə gəlin **SELECT clause**-ında **sub query**-in istifadə olunma qaydasını görək, eynən aşağıdakı şəkildə olduğu kimi.

gördüyünüz kimi yuxarıdakı query də **SELECT** hissəsində **sub query** istifadə etmiş olduq, lakin bu query bir o qədərdə məntiqli deyil, buna səbəb gördüyünüz kimi hər bir sətirdə **KING** yazısı təkrarlanmışdır.