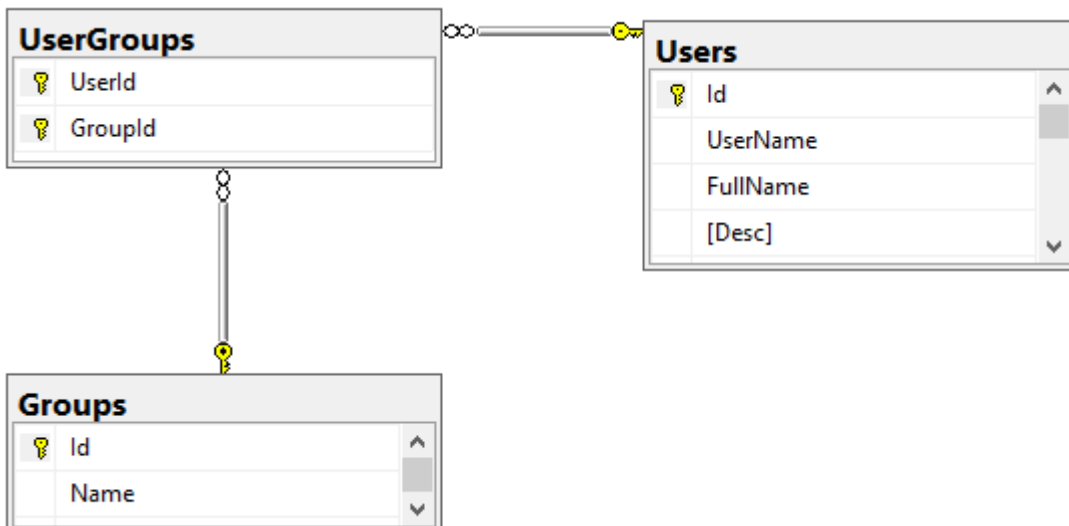


# SQL 'IN' ve 'NOT IN' Operatörleri

SQL dilinde sorgulama yapılırken bir değeri belli değerler arasından herhangi birine eşit olabileceği istendiği durumda değerler OR şartı ile birbirine bağlanır. "WHERE Id = 4 OR Id = 10 OR Id = 37" gibi... Değerlerin az olduğu durumlarda bu şekilde yazılabilir ancak değerlerin sayısının arttığı veya bu değerlerin aşağıdaki örnekte olduğu gibi kodun yazıldığı zamanda belli olmaması durumunda bu şekilde OR şartı ile bağlama hem zahmetli hem performans olarak sorgu optimizasyonunu olumsuz etkilemektedir.

Kolon değerinin bir liste içerisinde herhangi bir değere eşit olması istenen durumda **IN** operatörü kullanılabilir. Listede herhangi bir değere eşit olması istenilmeyen durumda da **NOT IN** operatörü kullanılır.

Konumuzu en iyi şekilde basit bir örnek ile açıklayabiliriz.



## Örnek Senaryo Tabloları

Yukarıdaki tablo şemasında;

- Users tablosu, kullanıcı listesini
- Groups tablosu, gruplar tablosunu
- UserGroups tablosu, kullanıcıların üye oldukları grupları n – n ilişkisi içerisinde tutar.

Sorumuz şu: Hiç bir gruba üye olmayan kullanıcıların listesini veren SQL scriptini yazınız.

```
select Id, UserName, FullName  
from Users  
where Id NOT IN (select UserId from UserGroups)
```

|    | Id   | UserName | FullName |
|----|------|----------|----------|
| 1  | 8    | aozk     | AL       |
| 2  | 10   | hka      | HK       |
| 3  | 12   | hka      | HK       |
| 4  | 13   | hka      | HK       |
| 5  | 14   | hka      | HK       |
| 6  | 15   | hka      | HK       |
| 7  | 34   | 6b7      | YA       |
| 8  | 35   | esa      | EV       |
| 9  | 41   | mka      | MU       |
| 10 | 42   | mo       | MU       |
| 11 | 44   | hva      | VA       |
| 12 | 45   | me       | ME       |
| 13 | 1076 | eto      | EM       |
| 14 | 1079 | say      | SA       |
| 15 | 1096 | edd      | EF       |
| 16 | 1098 | ndc      | NU       |
| 17 | 1101 | oul      | ÖM       |
| 18 | 1111 | md       | MU       |
| 19 | 1112 | asa      | AL       |
| 20 | 1118 | ukil     | US       |
| 21 | 1122 | am       | AT       |
| 22 | 1124 | 01       | 21 HA    |

Bu sorgu çalışırken öncelikle UserGroups tablosundan her hangi bir gruba üye olan kullanıcıların UserId değerlerini alır, daha sonra Users tablosundan Id değeri UserGroups tablosundan alınan UserId değerlerinin içerisinde olmayan kullanıcıları getirir.

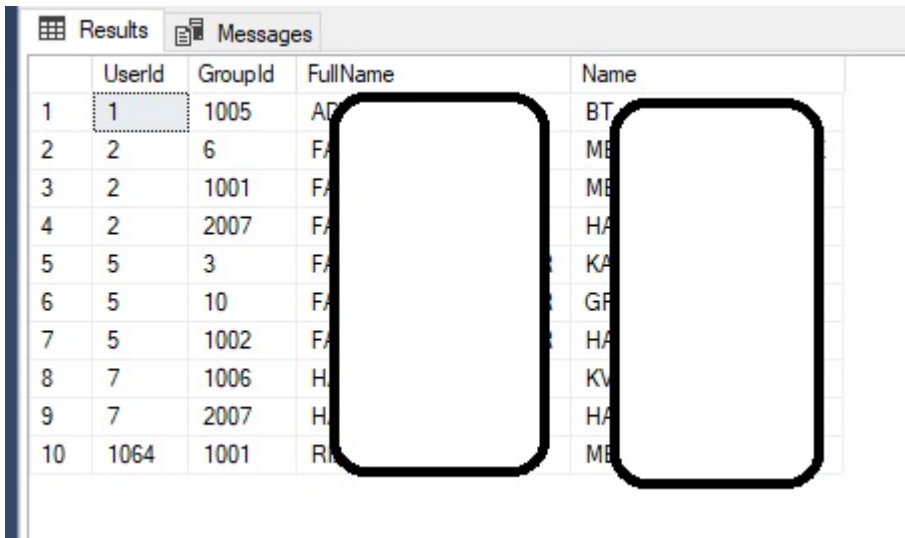
```
select * from UserGroups  
where UserId IN (1,3,6,87,45,332)
```

Bu sorguda da UserId değeri "1,3,6,87,45,332" değerlerinden herhangi birine eşit olan kayıtlar gelecektir.

Şimdi de sorguyu biraz daha karmaşıktırarak "YÖNETİCİLER" grubuna üye olan kullanıcılardan başka gruba da üye olanların

listesini ve üye oldukları grupların listesini çekeceğiz.

```
select
u.Id UserId,
gl.Id GroupId,
u.FullName,
gl.Name
from UserGroups ugl join Groups gl on ugl.GroupId = gl.Id
join Users u on ugl.UserId = u.Id
where UserId IN (
select UserId from UserGroups ug join Groups g on ug.GroupId =
g.Id
where g.Name = 'YÖNETİCİLER'
) and gl.Name != 'YÖNETİCİLER'
```



|    | UserId | GroupId | FullName | Name |
|----|--------|---------|----------|------|
| 1  | 1      | 1005    | AD       | BT   |
| 2  | 2      | 6       | FA       | MB   |
| 3  | 2      | 1001    | FA       | MB   |
| 4  | 2      | 2007    | FA       | HA   |
| 5  | 5      | 3       | FA       | KA   |
| 6  | 5      | 10      | FA       | GR   |
| 7  | 5      | 1002    | FA       | HA   |
| 8  | 7      | 1006    | HA       | KV   |
| 9  | 7      | 2007    | HA       | HA   |
| 10 | 1064   | 1001    | RI       | MB   |

---

# SQL Server Veri Tabanının MySql Üzerine Taşınması

[SQL Server](#) piyasa da en sık kullanılan veri tabanlarının

başında gelmektedir. Ancak özellikle [SQL Server](#)'ın ücretli olması sebebi ile zamanla kullanıcılar sistemlerini [MYSQL](#) gibi ücretsiz bir veri tabanı sistemine geçirmek isteyebiliyorlar. Bu noktada eski verilerin ne olacağı, yeni sisteme nasıl taşınacağı problemi ortaya çıkmaktadır.

Veri taşıma işlemi için ücretli alternatifler olmasının yanında [MYSQL](#)'in kendi içerisinde de dahili gelen veri taşıma aracı bulunmaktadır. Aşağıdaki YouTube videomuzda bu işlemin nasıl yapıldığını işlemeye çalışıyoruz.

---

## MySQL Kurulum

---

## Telefon Rehberi

[Java](#) ve [MySQL Server](#) ile yazılmış basit bir telefon rehberi uygulamasıdır. Projeye [GitHub](#) üzerinden aşağıdaki linkten ulaşabilirsiniz.

<https://github.com/saitorhan/Telefon-Rehberi>

