

آموزش SQL

با ترجمه

W3schools

قسمت هشتم

SQL در INNER JOIN

کلمه‌ی کلیدی INNER JOIN برای انتخاب رکوردهایی استفاده می‌شود که در هر دو جدول دارای مقدار مشترک هستند.

بیایید نگاهی بندازیم به بخشی از جدول Products (محصولات):

ProductID	ProductName	CategoryID	Price
1	Chais	1	18
2	Chang	1	19
3	Aniseed Syrup	2	10

و حالا بخشی از جدول Categories (دسته‌بندی‌ها):

CategoryID	CategoryName	Description
1	Beverages	Soft drinks, coffees, teas, beers, and ales
2	Condiments	Sweet and savory sauces, relishes, spreads, and seasonings
3	Confections	Desserts, candies, and sweet breads

SQL در INNER JOIN

حالا می‌خواهیم این دو جدول رو بر اساس ستون CategoryID با هم ترکیب کنیم:

```
SELECT ProductID, ProductName, CategoryName  
FROM Products  
INNER JOIN Categories ON Products.CategoryID =  
Categories.CategoryID;
```

[برای مشاهده خروجی کلیک کنید.](#)



SQL در INNER JOIN

کلمه‌ی کلیدی INNER JOIN برای انتخاب رکوردهایی استفاده می‌شود که در هر دو جدول دارای مقدار مشترک هستند.

بیایید نگاهی بندازیم به بخشی از جدول Products (محصولات):

ProductID	ProductName	CategoryID	Price
1	Chais	1	18
2	Chang	1	19
3	Aniseed Syrup	2	10

نکته:

INNER JOIN فقط ردیف‌هایی رو برمی‌گردونه که در هر دو جدول مقدار تطبیق‌پذیر داشته باشد.

یعنی اگه محصولی CategoryID نداشته باشه یا اون شناسه توی جدول Categories وجود نداشته باشه، اون محصول توی نتیجه نشون داده نمی‌شه.

سینتکس INNER JOIN

```
SELECT ستون‌ها  
FROM جدول۱  
INNER JOIN جدول۲  
ON جدول۱.ستون = جدول۲.ستون;
```

نام‌گذاری ستون‌ها

یه روش خوب اینه که همیشه اسم جدول رو کنار اسم ستون بیاری تا اشتباهی پیش نیاد. مثلاً:

```
SELECT Products.ProductID, Products.ProductName,  
Categories.CategoryName  
FROM Products  
INNER JOIN Categories ON Products.CategoryID =  
Categories.CategoryID;
```

در این مثال، چون CategoryID در هر دو جدول وجود داره، اگر اسم جدول رو ننویسی، خطا می‌گیری.

JOIN و INNER JOIN چه فرقی دارن؟ 🤔

در واقع هیچ فرقی ندارن!
کلمه‌ی JOIN به صورت پیش‌فرض همون INNER JOIN هست. پس این دو
دستور کاملاً یکسان عمل می‌کنن:

-- مثل قبلی ولی با JOIN ساده

```
SELECT Products.ProductID, Products.ProductName,  
Categories.CategoryName  
FROM Products  
JOIN Categories ON Products.CategoryID = Categories.CategoryID;
```

در این مثال، چون CategoryID در هر دو جدول وجود داره، اگر اسم جدول رو
ننویسی، خطا می‌گیری.

JOIN بین سه جدول

می‌تونن بین چندین جدول هم ارتباط برقرار کنن. مثلاً در این مثال، اطلاعات سفارش‌ها به همراه مشتری و شرکت حمل‌ونقل:

ShipperID	ShipperName	Phone
1	Speedy Express	(503) 555-9831
2	United Package	(503) 555-3199
3	Federal Shipping	(503) 555-9931



JOIN بین سه جدول

می‌تونی بین چندین جدول هم ارتباط برقرار کنی. مثلاً در این مثال، اطلاعات سفارش‌ها به همراه مشتری و شرکت حمل‌ونقل:

```
SELECT Orders.OrderID, Customers.CustomerName,  
Shippers.ShipperName  
FROM (Orders  
INNER JOIN Customers ON Orders.CustomerID =  
Customers.CustomerID)  
INNER JOIN Shippers ON Orders.ShipperID = Shippers.ShipperID);
```

[برای مشاهده خروجی کلیک کنید.](#)



کلمه‌ی کلیدی LEFT JOIN در SQL

دستور LEFT JOIN تمام رکوردهای جدول سمت چپ (table1) رو برمی‌گردونه، حتی اگر در جدول سمت راست (table2) رکوردی مطابق با اون وجود نداشته باشه.

اگر رکورد مطابقی در جدول سمت راست نباشه، مقدار ستون‌های اون جدول NULL خواهد بود.

```
SELECT نامستونها  
FROM جدول۱  
LEFT JOIN جدول۲  
ON جدول۱.ستون = جدول۲.ستون;
```

نکته: در بعضی از پایگاه‌های داده، به جای LEFT JOIN از LEFT OUTER JOIN هم استفاده می‌شود که معنای یکسانی دارند.

[برای مشاهده خروجی کلیک کنید.](#)

مثال LEFT JOIN:

دستور زیر، همه‌ی مشتری‌ها رو برمی‌گردونه، همراه با هر سفارشی که ممکنه داشته باشن:

```
SELECT Customers.CustomerName, Orders.OrderID
FROM Customers
LEFT JOIN Orders ON Customers.CustomerID = Orders.CustomerID
ORDER BY Customers.CustomerName;
```

[برای مشاهده خروجی کلیک کنید.](#)

نکته:

کلمه‌ی کلیدی LEFT JOIN باعث می‌شه همه‌ی رکوردهای جدول Customers نمایش داده بشن، حتی اگه اون مشتری هیچ سفارشی در جدول Orders نداشته باشه. در این صورت، ستون‌های مربوط به جدول Orders مقدار NULL خواهند داشت.

SQL در RIGHT JOIN

دستور RIGHT JOIN تمام رکوردهای جدول سمت راست (table2) رو برمیگردونه، و فقط رکوردهایی از جدول سمت چپ (table1) که با جدول راست تطابق دارند.

اگه هیچ رکوردی در جدول سمت چپ با جدول راست تطابق نداشته باشه، اون ستونها مقدار NULL خواهند داشت.

SQL در RIGHT JOIN

سینتکس

```
SELECT نام ستون ها  
FROM جدول ۱  
RIGHT JOIN جدول ۲  
ON جدول ۱.ستون = جدول ۲.ستون ;
```

نکته: در برخی پایگاه های داده، از RIGHT OUTER JOIN به جای RIGHT JOIN استفاده می شه که در عمل هیچ تفاوتی ندارن.

SQL در RIGHT JOIN

مثال RIGHT JOIN:

دستور زیر همه‌ی کارمندان رو نمایش می‌ده، همراه با سفارش‌هایی که ممکنه ثبت کرده باشن:

```
SELECT Orders.OrderID, Employees.LastName, Employees.FirstName  
FROM Orders  
RIGHT JOIN Employees ON Orders.EmployeeID = Employees.EmployeeID  
ORDER BY Orders.OrderID;
```

[برای مشاهده خروجی کلیک کنید.](#)

نکته:

در این دستور، چون از RIGHT JOIN استفاده شده، تمام رکوردهای جدول Employees (کارمندان) در خروجی ظاهر می‌شن، حتی اگه هیچ سفارشی در جدول Orders مربوط به اون‌ها نباشه. در مواردی که سفارش ثبت نشده، مقدار ستون‌های مربوط به جدول Orders NULL خواهد بود.

SQL در FULL OUTER JOIN

مثال FULL OUTER JOIN:

دستور FULL OUTER JOIN تمام رکوردهای هر دو جدول (جدول سمت چپ و جدول سمت راست) رو برمیگردونه، چه تطابقی وجود داشته باشه، چه نداشته باشه.

نکته: FULL OUTER JOIN و FULL JOIN با هم برابرند و یکسان عمل می‌کنن.

SQL در FULL OUTER JOIN

سینتکس

```
SELECT نامستونها  
FROM جدول1  
FULL OUTER JOIN جدول2  
ON جدول1.ستون = جدول2.ستون  
WHERE شرط؛
```



w3schoolfa

pyfaw3schools



SQL در FULL OUTER JOIN

مثال از FULL OUTER JOIN

دستور زیر تمام مشتری‌ها و تمام سفارش‌ها رو نمایش می‌ده، حتی اگر بین اون‌ها تطابق وجود نداشته باشه:

```
SELECT Customers.CustomerName, Orders.OrderID
FROM Customers
FULL OUTER JOIN Orders ON Customers.CustomerID =
Orders.CustomerID
ORDER BY Customers.CustomerName;
```

خروجی :

CustomerName	OrderID
<i>Null</i>	10309
<i>Null</i>	10310
Alfreds Futterkiste	<i>Null</i>
Ana Trujillo Emparedados y helados	10308
Antonio Moreno Taquería	<i>Null</i>

SQL در FULL OUTER JOIN

مثال از FULL OUTER JOIN

نکته مهم:

- FULL OUTER JOIN همه رکوردهایی که تطابق دارند رو نشون می‌ده.
- همچنین، رکوردهایی که تطابق ندارند از هر یک از دو جدول هم نمایش داده می‌شن.
- مثلاً سفارش‌هایی که مشتری‌شون در جدول Customers وجود نداره، یا مشتری‌هایی که سفارشی نداشتن.