

## Исключение от правилото за LEFT/RIGHT

При наличие на **NULL** стойности в FK колоната, трябва да сменим типа на връзката без да сменяме разположението на таблиците (1 L - M R), за да извадим онези от страната много, които нямат съответствие в страната 1.

(pk) user			(pk) articles			(fk)		
UID	UserName		AID	Title		UID		
---	---		---	---		---		
1	John	1 M	1	MySQL Performance Tunning		1		
2	Maria	o-----<	2	PostgreSQL Administration		2		
3	Peter		3	Mastering SQL Queries		1		
4	Anna		4	Aggregate Functions		2		
			5	Big Data		NULL		
			6	Data Science		NULL		

Резултат:

(pk)					
AID	Post		UserName		
-----	-----		-----		
1	MySQL Performance Tunning		John		
2	PostgreSQL Administration		Maria		
3	Mastering SQL Queries		John		
4	Aggregate Functions		Maria		
5	Big Data		NULL		
6	Data Science		NULL		

- FULL OUTER JOIN

(pk) user			(pk) articles			(fk)		
UID	UserName		AID	Title		UID		
---	---		---	---		---		
1	John	1 M	1	MySQL Performance Tunning		1		
2	Maria	o-----<	2	PostgreSQL Administration		2		
3	Peter		3	Mastering SQL Queries		1		
4	Anna		4	Aggregate Functions		2		
			5	Big Data		NULL		
			6	Data Science		NULL		

Резултат:

UID	Post	UserName
1	MySQL Performance Tunning	John
2	PostgreSQL Administration	Maria
1	Mastering SQL Queries	John
2	Aggregate Functions	Maria
NULL	Big Data	NULL
NULL	Data Science	NULL
3	NULL	Peter
4	NULL	Anna

## Изчисления в SELECT

Независимо от това какво и как изчисляваме, данните в таблиците **не се променят**.

### Оператори

- аритметични: +, -, \*, /
- конкатенация: ||

Сървър	Конкатенация
MS SQL	<code>SELECT firstname + ' ' + lastname ...</code>
Sybase	
---	---
MySQL	<code>SELECT concat(firstname, ' ', lastname) ...</code> <code>SELECT concat_ws(' ', firstname, lastname) ...</code>

```
SELECT 'Germany ' + 5 + 10 col
```

col
Germany 510

```
SELECT 'Germany ' + (5 + 10) col
```

col
Germany 15

```
SELECT
    firstname || ' ' || lastname Employee
FROM
    employees
```

## Oracle Docs

### Функции (Single-row functions)

#### 1. математически

Функция	Действие
ROUND(n[,m])	закръглява n до m-я знак
TRUNC(n[,m])	"реже" от n m десетични знака
CEIL(n)	закръглява до нагоре до цяло число
FLOOR(n)	закръглява надолу до цяло число

```
SELECT
    product_name
  , unit_price
  , unit_price * 1.95583 price_bgn
  , ROUND(unit_price * 1.95583, 2 ) price_round
  , TRUNC(unit_price * 1.95583, 2 ) price_trunc
  , CEIL( unit_price * 1.95583) price_ceil
  , FLOOR( unit_price * 1.95583) price_floor
FROM
    products
ORDER BY
    2 DESC
```

#### 2. текст

Функция	Действие
SUBSTR(txt, p[, l])	връща от позиция p в txt l символа
INSTR(txt, s[, p])	връща позицията на s в txt, започвайки от позиция p
UPPER(txt)/LOWER(txt)	преобразува в главни/малки букви txt
REPLACE(txt,v,s)	замества v в txt с s

| REGEXP\_REPLACE(txt,'reg','with') | замества с помощта на регулярен израз  
|

```
-- Marie Bertrand => M. Bertrand => M. B.
SELECT
    contact_name
-- first letter
, SUBSTR(contact_name, 1,1) ||
    '.' ||
-- last name
    SUBSTR(
-- col --
contact_name,
INSTR(
-- col --
    TRIM(contact_name),
-- what --
    ' ',
-- pos --
    -1,
-- occur --
    1
)
) Contact
--, REPLACE(contact_name, ' ', '@')
FROM
    customers
```