


# Оператор объединения UNION



Оператор **UNION** используется для объединения выходных данных двух или более SQL-запросов в единое множество строк и столбцов.

# пример

получить в одной таблице фамилии и идентификаторы студентов и преподавателей из Москвы

```
SELECT 'Студент ', фамилия, код_студента
FROM СТУДЕНТЫ
WHERE город = 'Москва'
UNION
SELECT 'Преподаватель', фамилия,
код_преподавателя
FROM ПРЕПОДАВАТЕЛИ
WHERE город = 'Москва';
```

## Соединение таблиц с использованием оператора JOIN

Если в операторе **SELECT** после ключевого слова **FROM** указывается не одна, а две таблицы, то в результате выполнения запроса, в котором отсутствует предложение **WHERE**, каждая строка одной таблицы будет соединена с каждой строкой второй таблицы.

# ПРИМЕР

Если необходимо получить фамилии студентов (таблица СТУДЕНТЫ), и для каждого студента – название университетов (таблица УЧЕБНЫЕ\_ЗАВЕД), расположенных в городе, где живет студент, то необходимо получить все комбинации записей о студентах и университетах в обеих таблицах, в которых значение поля ГОРОД совпадает.

```
SELECT СТУДЕНТЫ.фамилия, УЧЕБНЫЕ_ЗАВЕД.название,  
СТУДЕНТЫ.город
```

```
FROM СТУДЕНТЫ, УЧЕБНЫЕ_ЗАВЕД
```

```
WHERE СТУДЕНТЫ.город = УЧЕБНЫЕ_ЗАВЕД. Город;
```

## *внутреннее (INNER) соединение*

При этом соединяются только те строки таблиц, для которых истинным является предикат, задаваемый в предложении **ON** выполняемого запроса.

Приведенный ниже запрос может быть записан иначе, с использованием ключевого слова **JOIN**.

```
SELECT СТУДЕНТЫ.фамилия, УЧЕБНЫЕ_ЗАВЕД.  
название, СТУДЕНТЫ.город  
FROM СТУДЕНТЫ INNER JOIN  
УЧЕБНЫЕ_ЗАВЕД  
ON СТУДЕНТЫ.город = УЧЕБНЫЕ_ЗАВЕД. Город;
```

Ключевое слово **INNER** в этом запросе может быть опущено, так как эта опция в операторе **JOIN** действует по умолчанию

# ПРИМЕР

Пусть требуется найти фамилии всех студентов, получивших неудовлетворительную оценку, вместе с названиями предметов обучения, по которым получена эта оценка.

**SELECT** наименование, фамилия, оценка

**FROM** СТУДЕНТЫ, ПРЕДМЕТЫ, ОЦЕНКИ

**WHERE**

СТУДЕНТЫ.код\_студента = ОЦЕНКИ. код\_студента

**AND**

ПРЕДМЕТЫ.код\_предмета = ОЦЕНКИ. код\_предмета

**AND** ОЦЕНКИ.оценка = 2;

# То же самое с использованием оператора **JOIN**

```
SELECT наименование, фамилия, оценка  
FROM СТУДЕНТЫ JOIN ПРЕДМЕТЫ JOIN ОЦЕНКИ  
ON СТУДЕНТЫ.код_студента = ОЦЕНКИ.  
код_студента  
AND ПРЕДМЕТЫ.код_предмета = ОЦЕНКИ.  
код_предмета  
AND ОЦЕНКИ.оценка = 2;
```



# Внешнее объединение

Часто бывает полезна операция объединения двух запросов, в которой второй запрос выбирает строки, исключенные первым. Такая операция называется внешним объединением. Как отмечалось ранее, при использовании *внутреннего (INNER)* соединения таблиц соединяются только те их строки, в которых совпадают значения полей, задаваемые в запросе предложением **WHERE**. Однако во многих случаях это может привести к нежелательной потере информации. Рассмотрим еще раз приведенный выше пример запроса на выборку списка фамилий студентов с полученными ими оценками и идентификаторами предметов. При использовании, как это было сделано в рассматриваемом примере, внутреннего соединения в результат запроса не попадут студенты, которые еще не сдавали экзамен, и которые следовательно отсутствуют в таблице ОЦЕНКИ. Если же необходимо иметь записи об этих студентах в выдаваемом запросом списке, то можно присоединить сведения о студентах, не сдавших экзамен, путем использования оператора **UNION** с соответствующим запросом.

# Например:

```
SELECT SURNAME, CAST MARK AS CHAR (1), CAST SUBJ_ID AS  
  CHAR(10)  
FROM STUDENT, EXAM_MARKS  
WHERE STUDENT.STUDENT_ID = EXAM_MARKS.STUDENT_ID  
UNION  
SELECT SURNAME, CAST NULL AS CHAR (1), CAST NULL AS CHAR (10)  
FROM STUDENT  
WHERE NOT EXIST  
  (SELECT *  
   FROM EXAM_MARKS  
   WHERE STUDENT.STUDENT_ID = EXAM_MARKS.STUDENT_ID);
```