**Запросы на обновление**

Под обновлением данных подразумевается изменение значений в существующих записях таблицы. При этом возможно как изменение значений полей в группе строк (даже всех строк таблицы), так и правка значения поля отдельной строки.

Изменение записей в таблице реализуется с помощью запроса UPDATE. Простейший запрос на обновление выглядит так:

**UPDATE таблица SET поле = выражение**

где

таблица – имя таблицы, в которой будут проводиться изменения;

поле – поле таблицы, в которое будет внесено изменение; (поле - это столбец в таблице)

выражение – выражение, значение которого будет занесено в поле.

Пример: Уменьшить на 30% цену книг в таблице book.

**UPDATE book**

**SET price = price/100\*70;**

С помощью запросов на обновление можно изменять не все записи в таблице (как в предыдущем запросе), а только часть из них. Для этого в запрос включается ключевое слово WHERE, после которого указывается условие отбора строк для изменения.

Пример: Уменьшить на 30% цену тех книг в таблице book, количество которых меньше 5.

**UPDATE book**

**SET price = price/100\*70**

**WHERE amount < 5;**

Уменьшить на 10% цену тех книг в таблице book, количество которых принадлежит интервалу от 5 до 10, включая границы.

**UPDATE book**

**SET price=price/100\*90**

**WHERE amount BETWEEN 5 AND 10**

**Запросы на обновление нескольких столбцов**

Запросом UPDATE можно обновлять значения нескольких столбцов одновременно, при этом ключевое слово SET указывается только один раз. В этом случае простейший запрос будет выглядеть так:

**UPDATE** таблица **SET** поле1 = выражение1, поле2 = выражение2

где

таблица – имя таблицы, в которой будут проводиться изменения;

поле – поле таблицы, в которое будет внесено изменение; (поле - это столбец в таблице)

выражение – выражение, значение которого будет занесено в поле.

Пример: В столбце buy покупатель указывает количество книг, которые он хочет приобрести. Для каждой книги, выбранной покупателем, необходимо уменьшить ее количество на складе на указанное в столбце buy количество, а в столбец buy занести 0.

**UPDATE book**

**SET amount = amount - buy,** ← SET пишется только один раз,если столбцов несколько

**buy = 0;**

Пример: В таблице book необходимо скорректировать значение для покупателя в столбце buy таким образом, чтобы оно не превышало количество экземпляров книг, указанных в столбце amount. А цену тех книг, которые покупатель не заказывал, снизить на 10%.

**UPDATE book**

**SET buy=IF(buy>amount, amount,buy),**

**price=IF(buy=0,price/100\*90,price);**

Пояснения: В общем можно использовать IF.

-

**Запросы на обновление нескольких таблиц**

В запросах на обновление можно использовать несколько таблиц, но тогда

- для столбцов, имеющих одинаковые имена, необходимо указывать имя таблицы, к которой они относятся, например, book.price – столбец price из таблицы book, supply.price – столбец price из таблицы supply;

- все таблицы, используемые в запросе, нужно перечислить после ключевого слова UPDATE через запятую

- в запросе **обязательно условие WHERE**, в котором указывается условие при котором обновляются данные.

Пример: Если в таблице supply есть те же книги, что и в таблице book, добавлять эти книги в таблицу book не имеет смысла. Необходимо увеличить их количество на значение столбца amount таблицы supply.

**UPDATE book, supply**

**SET book.amount = book.amount + supply.amount**

**WHERE book.title = supply.title AND book.author = supply.author;**

Пример: Для тех книг в таблице book , которые есть в таблице supply, не только увеличить их количество в таблице book ( увеличить их количество на значение столбца amount таблицы supply), но и пересчитать их цену (для каждой книги найти сумму цен из таблиц book и supply и разделить на 2).

**UPDATE book, supply**

**SET book.amount=book.amount+supply.amount,**← **SET пишется только один раз,если столбцов несколько**

**book.price=((book.price+supply.price)/2)**

**WHERE book.title=supply.title AND book.author=supply.author;**

Пример: Занести в таблицу fine суммы штрафов, которые должен оплатить водитель, в соответствии с данными из таблицы traffic\_violation. При этом суммы вносить только в пустые поля столбца sum\_fine.

**UPDATE fine,traffic\_violation ← указать 2-е таблицы**

**SET fine.sum\_fine = traffic\_violation.sum\_fine**

**WHERE fine.violation = traffic\_violation.violation AND fine.sum\_fine IS Null;**

Пример (сложный !) <https://stepik.org/lesson/305762/step/6?unit=287773> :

Часть\_1 - Вывести фамилию, номер машины и нарушение только для тех водителей, которые на одной машине нарушили одно и то же правило два и более раз. При этом учитывать все нарушения, независимо от того оплачены они или нет.

Часть\_2 - В таблице fine увеличить в два раза сумму неоплаченных штрафов для отобранных на предыдущем шаге записей.

**CREATE TABLE temp AS** ← Создаём таблицу и туда заносим значения из выборки

**SELECT name, number\_plate, violation**

**FROM fine**

**GROUP BY name, number\_plate, violation**

**HAVING COUNT(number\_plate)>1**

**ORDER BY name, number\_plate, violation;**

**UPDATE fine, temp ←** Используем имеющуюся таблицу и созданную

**SET sum\_fine = sum\_fine \* 2**

**WHERE fine.name = temp.name AND fine.date\_payment IS NULL;**

В общем нельзя **апдэйтить и делать выборку** одновременно в подзапросе одного и того-же запроса. Можно создать временную таблицу, поместить нужные значения туда, а потом использовать их)))

Пример: в таблицу fine занести дату оплаты соответствующего штрафа из таблицы payment;

уменьшить начисленный штраф в таблице fine в два раза (только для тех штрафов, информация о которых занесена в таблицу payment) , если оплата произведена не позднее 20 дней со дня нарушения.

**UPDATE fine, payment**

**SET fine.date\_payment = payment.date\_payment,**

**fine.sum\_fine = IF(DATEDIFF(payment.date\_payment, payment.date\_violation)<=20,**

**fine.sum\_fine/2,**

**fine.sum\_fine)**

**WHERE fine.date\_violation = payment.date\_violation AND fine.date\_payment IS NULL;**

**SELECT \***

**FROM fine**