INSERT INTO название\_таблицы (column1, column2, column3, ... ) VALUES (value1, value2, value3, ...); ← Specify both the column names and the values to be inserted:

Если вносимое значение является строкой и вней имеется симол одинарной кавычки → ‘ ← то его надо экранировать обратным слэшем, \’ ←так как написание значений в этой команде идёт в одинарных кавычках и программа падает, не понимая , что выполнять.

Ещё Пример.

**INSERT INTO имя\_таблицы(столбец\_1, столбец\_2, ..., столбец\_N)**

**VALUES**

**(значение\_1\_1, значение\_1\_2, ..., значение\_1\_N),**

**(значение\_2\_1, значение\_2\_2, ..., значение\_2\_N),**

**...**

**(значение\_M\_1, значение\_M\_2, ..., значение\_M\_N);**

В результате выполнения запроса новая запись/записи заносится в конец обновляемой таблицы.

При составлении списка полей и списка значений необходимо учитывать следующее:

- количество полей и количество значений в списках должны совпадать;

- должно существовать прямое соответствие между позицией одного и того же элемента в обоих списках, поэтому первый элемент списка значений должен относиться к первому столбцу в списке столбцов, второй – ко второму столбцу и т.д.;

- типы данных элементов в списке значений должны быть совместимы с типами данных соответствующих столбцов таблицы ( целое число можно занести в поле типа DECIMAL, обратная операция - недопустима);

- **новые значения нельзя добавлять в поля, описанные как PRIMARY KEY AUTO\_INCREMENT;**

- рекомендуется заполнять все поля записи, если же поле пропущено, значение этого поля зависит от установленных по умолчанию значений, если значения не установлены - на данной платформе вставляется пустое значение (NULL).

**Добавление записей из другой таблицы**

С помощью запроса на добавление можно не только добавить в таблицу конкретные значения (список VALUES), но и записи из другой таблицы, отобранные с помощью запроса на выборку. В этом случае вместо раздела VALUES записывается запрос на выборку, начинающийся с SELECT. В нем можно использовать WHERE, GROUP BY, ORDER BY.

Правила соответствия между полями таблицы и вставляемыми значениями из запроса:

количество полей в таблице и количество полей в запросе должны совпадать;

должно существовать прямое соответствие между позицией одного и того же элемента в обоих списках, поэтому первый столбец запроса должен относиться к первому столбцу в списке столбцов таблицы, второй – ко второму столбцу и т.д.

типы столбцов запроса должны быть совместимы с типами данных соответствующих столбцов таблицы ( целое число можно занести в поле типа DECIMAL, обратная операция – недопустима).

Пример: Занести все книги из таблицы supply в таблицу book.

**INSERT INTO book (title, author, price, amount)**

**SELECT title, author, price, amount**

**FROM supply;**

**SELECT \* FROM book;**

| book\_id | title | author | price | amount

| 1 | Мастер и Маргарита | Булгаков М.А. | 670.99 | 3

| 2 | Белая гвардия | Булгаков М.А. | 540.50 | 5

| 3 | Идиот | Достоевский Ф.М. | 460.00 | 10

| 4 | Братья Карамазовы | Достоевский Ф.М. | 799.01 | 2

| 5 | Стихотворения и поэмы | Есенин С.А. | 650.00 | 15

| 6 | Лирика | Пастернак Б.Л. | 518.99 | 2

| 7 | Черный человек | Есенин С.А. | 570.20 | 6

| 8 | Белая гвардия | Булгаков М.А. | 540.50 | 7

| 9 | Идиот | Достоевский Ф.М. | 360.80 | 3

Ещё пример: Добавить из таблицы supply в таблицу book, все книги, кроме книг, написанных Булгаковым М.А. и Достоевским Ф.М.

**INSERT INTO book (title, author, price, amount)**

**SELECT title, author, price, amount**

**FROM supply**

**WHERE author NOT IN('Булгаков М.А.', 'Достоевский Ф.М.')**

**В запросах на добавление можно использовать вложенные запросы.**

Пример: Занести из таблицы supply в таблицу book только те книги, названия которых отсутствуют в таблице book.

**INSERT INTO book (title, author, price, amount)**

**SELECT title, author, price, amount**

**FROM supply**

**WHERE title NOT IN (**

**SELECT title**

**FROM book**

**);**