

# Администрирование баз данных



# Вставка данных в таблицы. Оператор *insert*

- **Однострочный оператор *insert*** – добавляет в таблицу новую запись
- **Многострочный оператор *insert*** – добавляет в таблицу несколько записей
- **Пакетная загрузка данных** – добавление в таблицу данных из внешнего источника (файла)

Создать базу данных

**lesson2**

Создать таблицу

**customers**  
**unit\_code** int unsigned auto\_increment primary key

**name** varchar(100) not null

**phone** varchar(20)

**e\_mail** varchar(50)

**status** enum('active', 'not active')

**reg\_date** date

**insert\_time** timestamp default current\_timestamp

```
CREATE TABLE customers (  
    unite_code INT unsigned NOT NULL AUTO_INCREMENT PRIMARY KEY,  
    name VARCHAR(100) NOT NULL,  
    phone VARCHAR(20),  
    e_mail VARCHAR(50),  
    status ENUM('active', 'not active') DEFAULT 'active',  
    reg_date DATE NOT NULL,  
    turnover DECIMAL(10,2);  
    insert_time TIMESTAMP DEFAULT CURRENT_TIMESTAMP  
);
```

## Однострочный оператор insert

Необходимо передать значения для всех столбцов таблицы

```
insert into tbl_name values (value_1, value_2, .. value_N);
```

Передаем названия необходимых столбцов для вставки и их значения

```
insert into tbl_name (col_1, col_2, .. col_N) values (value_1, value_2, value_N);
```

Строковые данные необходимо заносить в одиночных кавычках 'value\_1'.

Числовые значения допускается заводить без кавычек, при введение числовых данных в кавычках затрачивается дополнительное время на преобразование типа данных.

Пример

Необходимо передать значения для всех столбцов таблицы

```
insert into customers values (null, 'West company ltd.', 2394885697,  
'smith@email.com', 'active', date'2018-10-30', 304500.00, null);
```

Передаем названия необходимых столбцов для вставки и их значения

```
insert into customers (name, phone, e_mail, reg_date, turnover) values  
( 'East company ltd.', 2394885697, 'east@eastcompany.com', date'2018-10-30',  
904500.00 );
```

Приме

В процессе вставки порядок столбцов можно изменять

```
insert into customers (phone, e_mail, name, reg_date, turnover) values  
(89009910090, 'info@mdm.com', 'MDM company ltd.', date'2018-11-02',  
1750000.00);
```

Выборка внесенных данных из

```
Select * from customers;
```



## Многострочный оператор

### insert

Необходимо передать значения для всех столбцов таблицы

```
insert into tbl_name values (value_1,.. value_N), (value_2,.. value_2N) ;
```

Передаем названия необходимых столбцов для вставки и их значения

```
insert into tbl_name (col_1, col_2,.. col_N) values  
(value_1, ... value_N),  
(value_2, ... value_2N);
```

Приме

р

Необходимо передать значения для всех столбцов таблицы

```
insert into customers values (null, 'A company ltd.', 334885697,  
'acom@email.com', 'active', date'2018-10-20', 754500.50, null),  
(null, 'B company ltd.', 355555557, 'bcom@email.com', 'active',  
date'2018-10-20', 954500.50, null);
```

Передаем названия необходимых столбцов для вставки и их значения

```
insert into customers (name, phone, e_mail, reg_date, turnover) values  
( 'E company ltd.', 7884885697, 'ecom@eastcompany.com', date'2018-09-30',  
1904500.00),  
( 'D company ltd.', 6666885697, 'dcom@eastcompany.com', date'2018-09-21',  
404500.00);
```

## Пакетная загрузка данных

В файл c:/mysql/bin/my.ini в раздел [mysqld] добавить запись `secure-file-priv= ""`  
Настройка необходима для разрешения записи информации из файла, находящегося в любой директории

Перезапустить

MySQL

```
C:\mysql\bin>Net stop MYSQL  
C:\mysql\bin>Net start MYSQL
```

# Пакетная загрузка

~~данных~~  
загрузка данных из  
файла

```
LOAD DATA [LOCAL] INFILE 'filename' INTO TABLE tbl
```

Если параметр LOCAL указан, то файл читается с клиентского хоста. Если параметр не указан, то файл должен находиться на сервере

Создать

таблицу

```
create table tbl_load_data (  
  id int primary key,  
  car_make varchar(50)  
);
```

По умолчанию данные в импортируемом файле должны быть разделены символом табуляции.  
Создать текстовый файл **cars.txt** файл сохранить в c:/mysql/bin столбцы разделить табуляцией

```
1  AUDI
2  BMW
3  FORD
4  OPEL
5  VOLKSWAGEN
6  TOYOTA
```

Загрузить данные в

таблицу  
`mysql> LOAD DATA INFILE 'c:/mysql/bin/cars.txt' INTO TABLE tbl_load_data;`

Сделать выборку данных из

таблицы  
`mysql> select * from tbl_load_data;`

По умолчанию. Строка в текстовом файл должна заканчиваться невидимым символом \n, однако при создании текстового файла в среде Windows символ конца строки обозначается \r\n

Очищаем таблицу

```
truncate tbl_load_data;
```

командой:

```
LOAD DATA INFILE 'c:/mysql/bin/cars.txt' INTO TABLE tbl_load_data LINES
TERMINATED BY '\r\n';
```

Перед конструкцией **INTO TABLE** можно разместить одно из ключевых слов, которые обрабатывают ситуацию, когда данные из текстового файла дублируют значения первичного ключа или уникальных ключей

**IGNORE** – пропуск строк с дублирующими значениями

**REPLACE** – замена уже существующих записей новыми

**IGNORE X LINES** – позволяет задать **X** строк, которые необходимо пропустить от начала

файла  
Создать текстовый файл **new\_cars.txt** файл сохранить в c:/mysql/bin (столбцы разделить табуляцией)

1 MERCEDES

2 VOLVO

3 CITROEN

4 MAZDA

Загрузить данные в

таблицу

```
mysql> LOAD DATA INFILE 'c:/mysql/bin/new_cars.txt' INTO TABLE tbl_load_data  
LINES TERMINATED BY '\r\n';
```

```
ERROR 1062: Duplicate entry '1' for key 1
```

Загрузка без использования ключевых слов **IGNORE**, **REPLACE** приводит к ошибке, т.к. пытаемся загрузить уже существующие значения ключей.

Использование REPLACE приведет к замене содержимого таблицы (заменяет только то количество данных, сколько строк в новом файле)

```
mysql> LOAD DATA INFILE 'c:/mysql/bin/new_cars.txt'  
REPLACE INTO TABLE tbl_load_data LINES TERMINATED BY '\r\n';
```

```
mysql> select * from tbl_load_data;
```

Использование *IGNORE X LINES* приведет пропуску X строк от начала файла

Создать текстовый файл **add\_cars.txt** файл сохранить в c:/mysql/bin (столбцы разделить табуляцией)

10 FERRRI

11 BENTLEY

12 PORSCHE

```
LOAD DATA INFILE 'c:/mysql/bin/add_cars.txt' INTO TABLE tbl_load_data
```

```
LINES TERMINATED BY '\r\n'
```

```
IGNORE 1 LINES
```

```
;
```

```
mysql> select * from tbl_load_data;
```

В таблицу добавились BENTLEY, PORSCHE, строка с FERRARI пропущена



## Дополнительные опции оператора LOAD

DATA

**LINES STARTING BY** - определяет символ начала строки данных

**LINES TERMINATED BY** - определяет символ конца строки данных

**FIELDS TERMINATED BY** - определяет символ-разделитель между полями в строке

**FIELDS ENCLOSED BY** - данная конструкция определяет символ кавычек, которыми ограничиваются поля, по умолчанию равен пустой строке, т.к. кавычки не применяются

Создать файл cars2.txt в

c:/mysql/bin

Содержание файла:

"1","DACIA"

"2","MITSUBISHI"

"3","FIAT"

"4","SKODA"

```
LOAD DATA INFILE 'c:/mysql/bin/cars2.txt'  
REPLACE INTO TABLE tbl_load_data  
FIELDS TERMINATED BY ',' ENCLOSED BY ''  
LINES TERMINATED BY '\r\n' ;
```

## Задани

### 1. Создать таблицу **tbl\_electronics**

```
id int auto_increment primary key

name varchar(50)

price decimal(10,2)
```

### 2. Добавить в таблицу **tbl\_electronics**

Name	Price
55 Ultra HD 4K OLED	1499.50
SSD 256GB 860 Evo Basic	80.40
Apple MacBook Pro (2017)/13/256GB	1789.00
DSLR Canon 5D IV	2350.70

3. Взять файл с <http://80.232.241.14/mysql/files/tbl-import.txt> На основе данных в файле создать таблицу **cars\_import** и внести в данных из файла, используя пакетную загрузку данных.

## Решени

1.

```
create table tbl_electronics (  
    id int auto_increment primary key,  
    name varchar(50),  
    price decimal(10,2)  
);
```

2.

```
insert into tbl_electronics (name, price) values  
( '55 Ultra HD 4K OLED', 1499.50),  
( 'SSD 256GB 860 Evo Basic', 80.40),  
( 'Apple MacBook Pro (2017)/13/256GB', 1789.00),  
( 'DSLR Canon 5D IV', 2350.70);
```

## Решени

e

```
create table cars_import (  
    id int primary key,  
    make varchar(50),  
    model varchar(50),  
    price decimal(10,2)  
);
```

```
LOAD DATA INFILE 'c:/mysql/bin/tbl-import.txt'  
REPLACE INTO TABLE cars_import  
FIELDS TERMINATED BY ';'   
LINES TERMINATED BY '\r\n'  
IGNORE 1 LINES;
```