

3 Робота з даними за допомогою SQL

3 Data manipulation using SQL

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

- Додавання в таблицю одного або декількох рядків / Add one or multiple records into a table

INSERT INTO <table_name>

[(<columns_list>)]

VALUES

(<values_list_1>),

(<values_list_2>),

...

(<values_list_N>);

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

Основні параметри команди INSERT / Basic parameters of the INSERT statement

- ім'я таблиці / table name
- список імен стовпців / list of columns names
 - якщо стовпець не включений в список, при додаванні рядка буде встановлено значення за замовчуванням
 - if column is not included into a list, a default value will be set after record insert
- значення, що додаються в таблицю / values inserted into a table

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

Формат значень, що додаються / Inserted values format

- набір значень для одного рядка записується в дужки / a values set for a single row is provided in parentheses
- набір значень, що додаються, повинен відповідати списку стовпців / a values set should correspond to a list of columns
- якщо список стовпців не вказано – списку всіх стовпців таблиці / if a list of columns is not specified, it should correspond to a list of all table's columns
(DESCRIBE)

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

- значення всередині набору і самі набори відокремлюються комами / values inside a set, as well as sets are separated using commas
- символічні значення і значення дати / часу наводяться в одинарних лапках / character values and date / time values are shown using single quotes
- для числових значень лапки не обов'язкові / quotes are not necessary for numeric values
- десятковим роздільником для чисел з дробовою частиною служить точка / dot symbol is used as a separator for numeric values with fractional parts

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

- час і дати вводяться в форматах / time and dates should be formatted as
“YYYY-MM-DD”, “HH:MM:SS”
- щоб ввести невизначене значення, необхідно використовувати ключове слово **NULL** / the keyword **NULL** should be used to insert unidentified value
- замість значення можна вказати ключове слово **DEFAULT** / the keyword **DEFAULT** should be used instead of a value


3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

```
INSERT INTO supplier (supplier_id,  
    supplier_address, supplier_phone) VALUES  
(1, 'Kharkiv, Nauky av., 55, apt. 108', 'phone: 32-18-  
    44'),  
(2, 'Kyiv, Peremohy av., 154, apt. 3', ''),  
(3, 'Kharkiv, Pushkinska str., 77', 'phone: 33-33-44,  
    fax: 22-12-33'),  
(4, 'Odesa, Derebasivska str., 75', ''),  
(5, 'Poltava, Soborna str., 15, apt. 43', '');
```

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

USE supply;

SELECT * FROM supplier;

supplier (3×5)		
 supplier_id	supplier_address	supplier_phone
1	Kharkiv, Nauky av., 55, apt. 108	phone: 32-18-44
2	Kyiv, Peremohy av., 154, apt. 3	
3	Kharkiv, Pushkinska str., 77	phone: 33-33-44, fax
4	Odesa, Derebasivska str., 75	
5	Poltava, Soborna str., 15, apt. 43	

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

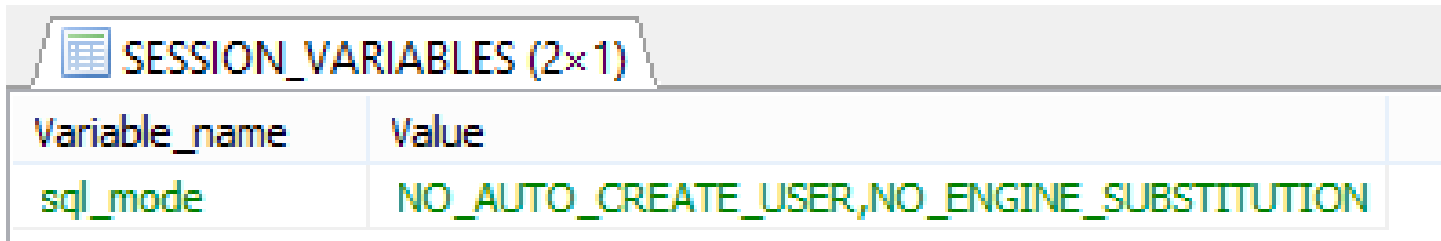
Результат команди INSERT в разі наявності некоректних значень залежить від режиму взаємодії клієнтського додатка з сервером MySQL

A result of execution of INSERT statement with incorrect values depends on a mode according to which a client application is interacting with a MySQL server

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

- Перевірити поточний режим взаємодії / Check current interaction mode

SHOW VARIABLES LIKE 'sql_mode';



SESSION_VARIABLES (2x1)	
Variable_name	Value
sql_mode	NO_AUTO_CREATE_USER,NO_ENGINE_SUBSTITUTION

Відсутність у змінній sql_mode ключових слів
STRICT_TRANS_TABLES и STRICT_ALL_TABLES = **нестрогий**
режим

If the variable sql_mode does not contain keywords
STRICT_TRANS_TABLES and STRICT_ALL_TABLES = **non strict**
mode

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

У **нестрогому** режимі вставляється некоректне значення перетвориться в допустимий

In a **non strict** mode inserted incorrect value is transformed into allowed value

- неправильна дата замінюється нульовою / incorrect date is replaced by a zero-date

0000-00-00 00:00:00

- зайві символи в символьному значенні відкидаються / redundant symbols of a character value are discarded

“abc” => CHAR (2) => “ab”

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

- внесення в числовий стовпець символічного значення / inserting a character value into a numeric column

“123abc” => INT => 123

“abc123” => INT => 0

- для NOT NULL стовпця не вказано значення і не задано значення за замовчуванням / both inserted and default values are not specified for a NOT NULL table

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

Тип даних стовпця Column data type	Значення, що додається	Inserted value
Числовий Numeric	0 черговий порядковий номер	0 next ordinal value
Дата та час Date and time	нульова дата і час поточна дата і час	zero date and time current date and time
Символьний Character	пуста строка	empty string
Перелік ENUM	перший з елементів списку	first element from a list of values

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

При роботі в **нестрогому** режимі операція додавання завершується успішно і генерується попередження

When working in a **non strict** mode, insert operation is finished successfully and a warning is generated

SHOW WARNINGS;

Спроба додати повторюване значення в стовпець первинного ключа або унікального індексу викликає помилку в будь-якому з режимів

An attempt to insert duplicate value into a primary key column or unique index causes an error in all modes

Те ж стосується FK / The same is also for FK (**InnoDB**)

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

sql_mode STRICT_TRANS_TABLES
 STRICT_ALL_TABLES

сервер працює в **строгому** режимі / server works in a **strict** mode

InnoDB – режими еквівалентні / modes are equal

- операція INSERT повністю скасовується / INSERT operation is completely cancelled
- видається повідомлення про помилку / error message is displayed

Числа з дробової частиною округлюються до цілого в будь-якому режимі без помилок і попереджень

Decimal values are transformed into integers without errors and warnings in both modes

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

Строгий режим / Strict mode

- необхідно задавати значення для стовпців без властивості DEFAULT / it is required to define values for columns without the DEFAULT property
- TIMESTAMP, ENUM, AUTO_INCREMENT – в разі відсутності значень, використовуються ті ж значення, що і в нестрогому режимі / if a value is not specified, the same values are used as in a non strict mode

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

Змінити режим взаємодії клієнта з сервером /
Change client-server interaction mode

SET SQL_MODE = '<mode>;

- Встановити нестрогий режим / Set a non strict mode

SET SQL_MODE = '';

- Встановити строгий режим / Set a strict mode

SET SQL_MODE = 'STRICT_TRANS_TABLES';

SET SQL_MODE = 'STRICT_ALL_TABLES;

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

SET SQL_MODE

- змінює режим взаємодії з сервером тільки для поточного з'єднання / changes server interaction mode only for a current connection
- не впливає на взаємодію сервера з іншими клієнтами / do not affect other sessions
- зберігається до моменту відключення від сервера / persists until disconnected from server

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

SET GLOBAL SQL_MODE

- застосовується для всіх знову підключаються до сервера клієнтів / applies to all new clients connected to a server
- раніше підключені клієнти продовжують працювати в колишньому режимі / previously connected clients continue to work in their modes
- зберігається до перезапуску сервера MySQL / remains until a MySQL server is restarted

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

UPDATE

ВСТАНОВИТИ НОВІ ЗНАЧЕННЯ В ОДНІЙ АБО ДЕКІЛЬКОХ
РЯДКАХ / set new values for one or several rows

UPDATE <table_name>

SET <column_name_1> = <value_1>,

...,

<column_name_N> = <value_N>

[**WHERE** <condition>]

[**ORDER BY** <column_name> [**ASC** | **DESC**]]

[**LIMIT** <rows_number>]



3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

UPDATE supplied

SET supplied_cost = supplied_cost * 0.95

WHERE contract_number = 4 AND supplied_product =
'Printer';

SELECT * FROM supplied WHERE contract_number = 4
AND supplied_product = 'Printer';

supplied (4×1)			
 contract_number	 supplied_product	supplied_amount	supplied_cost
4	Printer	41	332.50

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

Основні параметри / Main parameters

- ім'я таблиці / table name
- список стовпців і нових значень / a list of columns and new values

SET <column_name_1> = <value_1>, ...,
<column_name_N> = <value_N>

- можна використовувати колишні значення в рядку / it is possible to use previous values of a record
- умова відбору / filtering condition

WHERE <condition>

3.1 Вставка, видалення і оновлення даних / Insert, delete, and update data

- в умовах відбору можна використовувати вкладений запит / a nested query can be used in a filtering condition
- порядок застосування змін до рядків таблиці / order of applying changes to table rows

ORDER BY <column_name> [**ASC** | **DESC**]

- гранична кількість змінюваних рядків / set a limit number of changing records

LIMIT <rows_number>