

- 1. Типи даних в MySQL.**
- 2. Команди для роботи в MySQL**
- 3. Доступ до баз даних за допомогою PHP**

РОБОТА З БАЗАМИ ДАНИХ

1. ТИПИ ДАНИХ В MYSQL

MySQL підтримує такі типи даних:

- числовий
- дата та час
- рядковий

2. КОМАНДИ ДЛЯ РОБОТИ В MYSQL

1. Команди для роботи з БД:

□ створення БД

```
mysql> CREATE DATABASE [IF EXISTS]  
db_name;
```

□ видалення БД

```
mysql> DROP DATABASE [IF EXISTS] db_name;
```

□ встановлення поточної БД

```
mysql> USE db_name;
```

□ перегляд списку існуючих БД

```
mysql> SHOW DATABASES;
```

2. Команди для роботи з таблицями БД:

□ створення таблиць

```
mysql> CREATE TABLE [IF EXISTS] tablename  
  (columns);
```

Приклад:

```
create table customers  
( customerid int unsigned not null auto_increment  
  primary key,  
  name char (30) not null,  
  address char(40) not null,  
  city char(20) not null  
);
```

-
- видалення таблиць

```
mysql> DROP TABLE [IF NOT EXISTS]  
tablename [,tablename...];
```

- перегляд списку існуючих таблиць у
вибраній БД

```
mysql> SHOW TABLES;
```

3. Команды для работы с записями БД:

□ - вставка данных в БД

```
INSERT [INTO] tbl_name  
[(col_name1,col_name2,...)] VALUES (value1,  
value2,...)
```

Приклад:

```
mysql> insert into customers values  
(NULL, "Иван Иванов", "Иванова, 35",  
"Иваново");
```

```
mysql> insert into customers (name, city) values  
("Иван Иванов", "Иваново");
```

□ отримання даних з БД

SELECT items

FROM tables

[WHERE condition]

[GROUP BY group_type]

[HAVING where_definition]

[ORDER BY order_type]

[LIMIT limit_criteria] ;

Приклад:

```
mysql> select name, city from customers ;
```

WHERE — встановлює критерій відбору певних рядків.

3. ДОСТУП ДО БАЗ ДАНИХ ЗА ДОПОМОГОЮ RНР

Основні кроки виконання запитів до БД за допомогою RНР:

1. Перевірка та фільтрація даних, що виходять від користувача
2. Встановлення з'єднання з БД
3. Передача запиту БД
4. Отримання результатів
5. Представлення результатів користувачу

Перевірка та фільтрація даних:

- `trim()` – видалення всіх зайвих пробілів по краях слова.
- перевірка заповнення полів:

```
if (!$searchtype || !$searchterm)
{
echo "Ви не повністю заповнили форму, поверніться
та спробуйте ще раз ";
exit;
}
```
- `addslashes (string str)` – повертає рядок, в якому перед кожним спецсимволом (", \, ') додається додатковий символ \.

2. Встановлення з'єднання з БД:

- ❑ `mysql_pconnect ([string host [-port] [:/socketpath]], [string user], [string password]) ;`
- ❑ `mysql_connect ([string host [-port] [:/socketpath]], [string user], [string password]) ;`

Приклад:

```
$db = mysql_pconnect("localhost", "root",  
    "root");
```

-
- `mysql_pconnect` – встановлює постійне з'єднання з сервером MySQL
 - `mysql_connect` – встановлює одиничне з'єднання з сервером MySQL.
Завершується з'єднання командою `mysql_close()`
 - Вибір БД: `mysql_select_db (string database);`

Приклад: `mysql_select_db ("books");`

3. Передача запиту БД

□ `mysql_query(string query, [int database_connection]);`

`mysql_query` – відсилає запит MySQL.

Приклад:

```
$q = mysql_query("SELECT * FROM mytable");
```

```
mysql_db_query(string database, string query, [int database_connection] );
```

Приклад:

```
$q = mysql_db_query("books", "SELECT * FROM mytable");
```

4. Отримання результатів

- `mysql_result (resource result, int row)` – повертає значення одного елемента з набору записів.

Приклад:

```
$result = mysql_query("SELECT name FROM  
work.employee");
```

```
echo mysql_result($result,2);
```

- `mysql_fetch_array(resource result)` – повертає масив з опрацьованим рядом результатів.

Приклад:

```
$row = mysql_fetch_array($result) ;
```

-
- ❑ `mysql_fetch_row (resource result)` – працює аналогічно команді `mysql_fetch_array ()`, але опрацьовує тільки один ряд результатів.
 - ❑ `mysql_fetch_assoc(resource result)` – повертає асоціативний масив з опрацьованим рядом результатів.
 - ❑ `mysql_num_rows(resource result)` – повертає кількість рядків результату запиту.

Приклад:

```
$result = mysql_query("SELECT * FROM table");  
$num_rows = mysql_num_rows($result);
```

- ❑ `mysql_num_fields (resource result)` – повертає кількість полів (стовбців) результату запиту.

5. Представлення результатів користувачу

- `stripslashes(string str)` – команда обернена до `addslashes()`.
- `quotemeta (string str)` – повертає рядок, в який додається `/` перед кожним зі спецсимволів: `. \ + * ? [^] ($)`.
- `htmlspecialchars()` – перетворює спеціальні символи.