

## Introduction to database programming

In this tutorial we will explain how to work with database using Joomla database functions. We will show you how to execute a query and work with dataset that query returned.

### Introduction

In this tutorial we will explain how to work with database using Joomla database functions. We will show you how to execute a query and work with dataset that query returned.

Few more things...

We assume that you will be using tables that exist in the database that is used by Joomla installation you are working on. For the purpose of this tutorial we will be using table "#\_\_users". We also assume that you are familiar with SQL syntax and have a basic knowledge on Joomla component development.

First of all it is important to explain the prefix "#\_\_". If you were installing Joomla before you probably remember the table prefix field you were asked to fill in (default is "jos\_"). That prefix allows you to install many different Joomla installations (some other php applications support this feature) into one database. That is very handy feature when you are faced with hosting account that limits you to have a single database. So, when you install Joomla every table gets a prefix "jos\_" (or other if you changed the table prefix field during installation), and the "#\_\_" is the universal prefix that Joomla will parse and replace with the prefix defined during installation.

### Executing a simple query

Following example will demonstrate how to execute a query and retrieve dataset result.

```
$query = "SELECT name, username, email FROM #__users WHERE username = 'admin' ";
```

```
$database->setQuery( $query );
```

```
$rows = $database -> loadObjectList();  
for( $i=0; $i<count($_rows); $i++ )  
{  
    $row = $rows[$j];  
    echo "Name : ".$row->name;  
    echo " / Username : ".$row->username;  
    echo " / Email : ".$row->email;  
    echo "<br>";  
}
```

This code will display something like this :

```
Administrator / admin / some_email_address@domain.com
```

Although the code functionality is obvious it is important to highlight the `$row` object variable. It represents a dataset row and the field names from query are its properties.

## Database object methods

Every time we want to execute a query we have to load it into database object using `setQuery` function. After that we can use one of the following functions to execute SQL statement.

To execute a query that doesn't return result (or we don't need the result) we can use `query` method.

```
$query = "UPDATE #__users SET name = 'John Smith' WHERE username = 'admin' ";  
$database->query();
```

If we expect the query will return single row we can use `loadObject` method. This will return single object (unlike `loadObjectList` which returns array of objects) so we can work with data immediately.

```
$query = "SELECT name, username, email FROM #__users WHERE username = 'admin' ";
```

```
$database->setQuery( $query );
```

```
$row = $database -> loadObject();
```

Now we can access data directly ( `$row->name` ).

For queries that return more than single row we use `loadObjectList` method (first code example in this article).

## Summary

Working with database in Joomla is quite simple. You don't need to worry about connection to database, you make SQL statement, execute it, and if successful you will get a dataset in a form of an array where each row is an object containing data of each row.