



Бесплатная электронная книга

# УЧУСЬ

---

## pdo

Free unaffiliated eBook created from  
**Stack Overflow contributors.**

#pdo

.....	1
<b>1: pdo</b> .....	<b>2</b>
.....	2
Examples.....	2
.....	2
.....	<b>2</b>
<b>PDO:</b> .....	<b>3</b>
<b>PDO:</b> .....	<b>3</b>
.....	3
PDO Errormode.....	4
<b>2: SQL-</b> .....	<b>7</b>
.....	7
Examples.....	7
.....	7
.....	<b>9</b>

---

# Около

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [pdo](#)

It is an unofficial and free pdo ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official pdo.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to [info@zzzprojects.com](mailto:info@zzzprojects.com)

---

# глава 1: Начало работы с pdo

## замечания

В этом разделе представлен обзор того, что такое pdo, и почему разработчик может захотеть его использовать.

Следует также упомянуть о любых крупных предметах в pdo и ссылки на связанные темы. Поскольку Documentation for pdo является новым, вам может потребоваться создать начальные версии этих связанных тем.

## Examples

### Установка или настройка

PDO - это универсальная команда подключения к базе данных в PHP, она поддерживает 12 различных типов баз данных, например *MySQL*, *MongoDB*, *NoSQL*. Большим преимуществом PDO является то, что он вычисляет ваш код для поддержки типа базы данных, поэтому вам не нужно делать никаких шансов при переходе на другую систему баз данных.

---

## Резюме

	PDO	MySQLi
Поддержка базы данных	12 различных драйверов	MySQLi
API	ООП	ООП + процедурный
соединение	Легко	Легко
Именованные параметры	да	нет
Отображение объектов	да	да
Подготовленные отчеты (клиентская сторона)	да	нет
Спектакль	Быстро	Быстро

	PDO	MySQLi
Хранимые процедуры	да	да

---

## Подключение к базе данных с PDO:

Соединительная часть выглядит неловко, но нам нужно иметь дело. Остальная часть PDO проста и полезна, это также помогает сделать безопасную часть еще проще.

```
$connection = new PDO("mysql:host=localhost;dbname=myDatabase, username, password);
```

Соединение PDO выполняется следующим образом:

```
PDO ( database type : host = host ; dbname = database name , root , password );
```

---

## Безопасный выбор из базы данных с помощью PDO:

```
// We use a array to hold the data about whats the :var is in normal $var
$params = array(
    ':username' => '$username',
    ':email' => $mail,
);

// Prepare the SQL and using named secure parameters ":username"
$stmt = $pdo->prepare('SELECT * FROM users WHERE username = :username AND email = :email');

// Execute the $params and send them to the $pdo->prepare
$stmt->execute($params);
```

Код, который вы только что прочитали, - это защищенные агенты SQL-инъекции

---

## Учебники:

Установка:

[Как установить PDO, если у вас его нет](#)

Гиды:

[Учебник W3Schools](#)

[Tuts + Tutorial \(рекомендуется\)](#)

## PDO Настройка атрибута Errormode

PDO :: `setAttribute` устанавливает атрибут в дескрипторе базы данных. Описание `setAttribute`:

```
public bool PDO::setAttribute ( int $attribute , mixed $value )
```

**PDO :: ATTR\_ERRMODE:** этот атрибут используется для сообщения об ошибках. Он может иметь одно из следующих значений.

- `PDO::ERRMODE_SILENT` : Если `ATTR_ERRMODE` не задан в коде, `ERRMODE_SILENT` является значением по умолчанию атрибута `ATTR_ERRMODE`. Он устанавливает коды ошибок. В бесшумном режиме, если в SQL возникает ошибка, PDO не будет исключать исключений; PDO не выдаст никаких предупреждений; он просто вернет `false`. Значение `PDO :: ERRMODE_SILENT` равно 0. Скрипт будет работать без возникновения каких-либо ошибок или предупреждений.
- `PDO::ERRMODE_WARNING` : Это значение вызывает `E_WARNING`. В предупреждающем режиме, если в SQL возникает ошибка, PDO выдаст предупреждения, но скрипт будет продолжать работать. Значение `PDO :: ERRMODE_WARNING` равно 1. Скрипт будет запускаться с предупреждением об ошибке.
- `PDO::ERRMODE_EXCEPTION` : это значение `PDO::ERRMODE_EXCEPTION` исключения. В режиме исключения, если в SQL возникает ошибка, PDO будет генерировать исключения, и скрипт перестанет работать. Значение `PDO :: ERRMODE_EXCEPTION` равно 2. Сценарий прекратит выполнение генерации ошибки, которая выдает исключение.

Пример. Теперь мы рассмотрим различные значения атрибута `ATTR_ERRMODE` с некоторыми примерами. Для этого создайте базу данных с именем `learn_project_db` и вставьте в нее таблицу `user_table` . Для этого можно использовать следующий фрагмент кода SQL:

```
DROP DATABASE IF EXISTS `learn_project_db`;  
CREATE DATABASE `learn_project_db`;  
USE `learn_project_db`;  
CREATE TABLE `user_table` (  
  `user_email` varchar(50) PRIMARY KEY,  
  `user_password` varchar(50) NOT NULL  
);  
INSERT INTO `user_table` (`user_email`, `user_password`) VALUES  
( 'test1@example.com', '123' ),  
( 'test2@example.com', '1234' ),  
( 'test3@example.com', '12345' );
```

Сначала мы будем проверять, что произойдет, если мы не установим `ATTR_ERRMODE` и у нас будет ошибка в SQL-запросе. Создайте файл PHP с именем `default.php` и попробуйте следующее:

```
<?php
```

```

$server = "localhost";
$db_username = "root";
$db_password = "";
$db_name = "learn_project_db";
$conn = new PDO("mysql:host=$server;dbname=$db_name",$db_username,$db_password);
$sql_query = "SELECT * FROM wrong_user_table";
$stmt = $conn->prepare($sql_query);
$stmt->execute();
$result_set = $stmt->fetchAll();
var_dump($result_set);
/*Get the current error mode of PDO*/
$current_error_mode = $conn->getAttribute(PDO::ATTR_ERRMODE);
echo "<br>";
echo "Value of PDO::ATTR_ERRMODE: ".$current_error_mode;
?>

```

Обратите внимание, что имя таблицы `wrong_user_table` в запросе, который не определен в базе данных, которую мы создали ранее. Но, поскольку мы не устанавливали `ATTR_ERRMODE` он запускает сценарий, не выдавая никаких исключений или не выдавая никаких предупреждений. Он выведет пустой массив в качестве результирующего набора. Следует также заметить, что значение `PDO::ATTR_ERRMODE` равно 0.

Теперь мы будем проверять, что произойдет, если мы установим `ATTR_ERRMODE` со значением `PDO::ERRMODE_WARNING` и у нас будет ошибка в SQL-запросе. Создайте файл PHP с именем `warning.php` и попробуйте следующее:

```

<?php
$server = "localhost";
$db_username = "root";
$db_password = "";
$db_name = "learn_project_db";
$conn = new PDO("mysql:host=$server;dbname=$db_name",$db_username,$db_password);
$conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_WARNING);
$sql_query = "SELECT * FROM wrong_user_table";
$stmt = $conn->prepare($sql_query);
$stmt->execute();
$result_set = $stmt->fetchAll();
var_dump($result_set);
/*Get the current error mode of PDO*/
$current_error_mode = $conn->getAttribute(PDO::ATTR_ERRMODE);
echo "<br>";
echo "Value of PDO::ATTR_ERRMODE: ".$current_error_mode;
?>

```

Результат `warning.php` :

```

Warning: PDOStatement::execute(): SQLSTATE[42S02]: Base table or view not found: 1146 Table
'learn_project_db.wrong_user_table' doesn't exist in E:\xampp\htdocs\oop\db.php on line 10
array(0) { }
Value of PDO::ATTR_ERRMODE: 1

```

На этот раз, когда мы установим `ATTR_ERRMODE` с параметром `PDO::ERRMODE_WARNING` он покажет предупреждение. Скрипт выполняется успешно и показывает пустой массив как вывод со значением `PDO::ATTR_ERRMODE` из 1.

Наконец, мы собираемся проверить, что произойдет, если мы установим `ATTR_ERRMODE` со значением `PDO::ERRMODE_EXCEPTION` и у нас будет ошибка в SQL-запросе. Создайте файл PHP с именем `error.php` и попробуйте следующее:

```
<?php
    $server = "localhost";
    $db_username = "root";
    $db_password = "";
    $db_name = "learn_project_db";
    $conn = new PDO("mysql:host=$server;dbname=$db_name", $db_username, $db_password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    $sql_query = "SELECT * FROM wrong_user_table";
    $stmt = $conn->prepare($sql_query);
    $stmt->execute();
    $result_set = $stmt->fetchAll();
    var_dump($result_set);
    /*Get the current error mode of PDO*/
    $current_error_mode = $conn->getAttribute(PDO::ATTR_ERRMODE);
    echo "<br>";
    echo "Value of PDO::ATTR_ERRMODE: ".$current_error_mode;
?>
```

Результат `error.php` :

```
Fatal error: Uncaught exception 'PDOException' with message 'SQLSTATE[42S02]: Base table or
view not found: 1146 Table 'learn_project_db.wrong_user_table' doesn't exist' in
E:\xampp\htdocs\oop\db.php:10 Stack trace: #0 E:\xampp\htdocs\oop\db.php(10): PDOStatement-
>execute() #1 {main} thrown in E:\xampp\htdocs\oop\db.php on line 10
```

На этот раз, когда мы установим значение `ATTR_ERRMODE` с `PDO::ERRMODE_EXCEPTION` оно будет генерировать `PDOException` которое генерирует фатальную ошибку. Сценарий прекратит выполнение после того, как будет выбрано исключение. Это наиболее распространенный подход для обработки ошибок, связанных с запросами базы данных. В большинстве случаев мы устанавливаем атрибут `ATTR_ERRMODE` с этим значением для обработки любого исключения, которое может находиться в SQL-запросе.

Прочитайте Начало работы с pdo онлайн: <https://riptutorial.com/ru/pdo/topic/4393/начало-работы-с-pdo>



---

# глава 2: Подготовьте и выполните свои SQL-запросы

## замечания

### Предупреждение

Подготовленный оператор не может заботиться о диком параметре для имен таблиц. Например, это следующее утверждение неверно:

```
$query = "SELECT name, city FROM ? WHERE id = ? AND country = ?";
```

Правильный подготовленный запрос:

```
$query = "SELECT name, city FROM users WHERE id = ? AND country = ?";
```

## Examples

### использование

```
// 1. Connect to the database (this example with MySQL)
$host = 'localhost';
$dbname = 'users';
$user = 'root';
$password = '';
$dsn = "mysql:host=$host;dbname=$dbname";
$pdo = new PDO($dsn, $user, $password);

// 2. Prepare your query

// 2.1 First way
$query_1 = "SELECT name, city FROM users WHERE id = ? AND country = ?";
$stmt_1 = $pdo->prepare($query_1);

// 2.2 Second way
$query_2 = "SELECT name, city FROM users WHERE id = :id AND country = :country";
$stmt_2 = $pdo->prepare($query_2);

// 3. Execute your query

// 3.1 With the first way
$stmt_1->execute([1, 'US']);

// 3.2 With the second way
$stmt_2->execute([
    ':id' => 1,
    ':country' => 'US'
]);
```

```
// 4. Fetch your data
$data_1 = $statement_1->fetchAll();
$data_2 = $statement_2->fetchAll();
```

Прочитайте **Подготовьте и выполните свои SQL-запросы онлайн:**

<https://riptutorial.com/ru/pdo/topic/6511/подготовьте-и-выполните-свои-sql-запросы>

---

## кредиты

S. No	Главы	Contributors
1	Начало работы с pdo	<a href="#">arsho</a> , <a href="#">Community</a> , <a href="#">hjpotter92</a> , <a href="#">Ilyas Mimouni</a> , <a href="#">manian</a> , <a href="#">TheCrazyProfessor</a> , <a href="#">Your Common Sense</a>
2	Подготовьте и выполните свои SQL-запросы	<a href="#">Anwar Nairi</a> , <a href="#">Your Common Sense</a>