



EBook Gratuito

APPENDIMENTO

pdo

Free unaffiliated eBook created from
Stack Overflow contributors.

#pdo

Sommario

Di.....	1
Capitolo 1: Iniziare con pdo.....	2
Osservazioni.....	2
Examples.....	2
Installazione o configurazione.....	2
Sommario.....	2
Connessione al database con PDO:.....	2
Selezione sicura dal database usando PDO:.....	3
Tutorial:.....	3
Impostazione errata di attributo PDO.....	3
Capitolo 2: Prepara ed esegui le tue istruzioni SQL.....	7
Osservazioni.....	7
Examples.....	7
uso.....	7
Titoli di coda.....	9

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

Capitolo 1: Iniziare con pdo

Osservazioni

Questa sezione fornisce una panoramica di cosa è pdo e perché uno sviluppatore potrebbe volerlo usare.

Dovrebbe anche menzionare qualsiasi argomento di grandi dimensioni all'interno di pdo e collegarsi agli argomenti correlati. Poiché la documentazione di pdo è nuova, potrebbe essere necessario creare versioni iniziali di tali argomenti correlati.

Examples

Installazione o configurazione

PDO è un comando di connessione al database universale in PHP, supporta 12 tipi di database diversi, ad esempio *MySQL*, *MongoDB*, *NoSQL*. Un grande vantaggio su PDO è che calcola il codice per supportare il tipo di database, quindi non è necessario avere alcuna possibilità quando si passa a un altro sistema di database.

Sommario

	DOP	MySQLi
Supporto per database	12 diversi driver	MySQLi
API	OOP	OOP + procedurale
Connessione	Facile	Facile
Parametri nominati	sì	No
Mappatura degli oggetti	sì	sì
Dichiarazioni preparate (lato client)	sì	No
Prestazione	Veloce	Veloce
Procedura di archiviazione	sì	sì

Connessione al database con PDO:

La parte di connessione sembra imbarazzante ma che dobbiamo affrontare. Il resto del PDO è semplice e utile, aiuta anche a rendere la parte sicura ancora più semplice.

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

La connessione PDO è ordinata come segue:

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

Selezione sicura dal database usando 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"
$pdo->prepare('SELECT * FROM users WHERE username = :username AND email = :email');

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

Il codice che hai appena letto è l'iniezione SQL degli agenti protetti

Tutorial:

Installare:

[Come installare PDO se non ce l'hai](#)

Guide:

[Tutorial su W3Schools](#)

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

Impostazione errata di attributo PDO

PDO :: setAttribute imposta un attributo sull'handle del database. La descrizione di setAttribute è:

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

PDO: ATTR_ERRMODE: questo attributo viene utilizzato per la segnalazione degli errori. Può avere uno dei seguenti valori.

- **PDO::ERRMODE_SILENT** : se ATTR_ERRMODE non è impostato nel codice, ERRMODE_SILENT è il valore predefinito dell'attributo ATTR_ERRMODE. Imposta i codici di errore. In modalità

silenziosa, se c'è un errore in SQL, PDO non farà eccezione; DOP non emetterà avvisi; restituirà semplicemente false. Il valore di PDO :: ERRMODE_SILENT è 0. Lo script verrà eseguito senza generare alcun errore o avviso.

- PDO::ERRMODE_WARNING : questo valore aumenta E_WARNING. In modalità di avviso, se c'è un errore in SQL, PDO emetterà avvisi ma lo script continuerà a funzionare. Valore di PDO :: ERRMODE_WARNING è 1. Lo script verrà eseguito generando un avviso sull'errore.
- PDO::ERRMODE_EXCEPTION : questo valore genera eccezioni. In modalità eccezione, se c'è un errore in SQL, PDO genererà eccezioni e lo script smetterà di funzionare. Il valore di PDO :: ERRMODE_EXCEPTION è 2. Lo script interromperà l'esecuzione dell'errore che genera l'eccezione.

Esempio: ora vedremo i vari valori dell'attributo ATTR_ERRMODE con alcuni esempi. Per fare ciò, creare un database chiamato `learn_project_db` e inserire una tabella denominata `user_table` al suo interno. Il seguente frammento SQL può essere utilizzato per ottenerlo:

```
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');
```

Per prima cosa controlleremo cosa accadrà se non impostiamo `ATTR_ERRMODE` e avremo errore nella query SQL. Crea un file PHP chiamato `default.php` e prova questo:

```
<?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;
?>
```

Si noti che il nome della tabella è `wrong_user_table` nella query che non è definita nel database creato in precedenza. Tuttavia, poiché non abbiamo impostato `ATTR_ERRMODE`, verrà eseguito lo script senza `ATTR_ERRMODE` eccezioni o avvisi. Produrrà un array vuoto come set di risultati.

Dovremmo anche notare che il valore di `PDO::ATTR_ERRMODE` è 0.

Ora controlleremo cosa accadrà se impostiamo `ATTR_ERRMODE` con valore `PDO::ERRMODE_WARNING` e

avremo errore nella query SQL. Crea un file PHP chiamato `warning.php` e prova questo:

```
<?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;
?>
```

L'output di `warning.php` è:

```
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
```

Questa volta impostando il valore `ATTR_ERRMODE` con `PDO::ERRMODE_WARNING` verrà visualizzato un messaggio di avviso. Lo script viene eseguito correttamente e mostra un array vuoto come output con il valore di `PDO::ATTR_ERRMODE` di 1.

Infine, controlleremo cosa accadrà se impostiamo `ATTR_ERRMODE` con valore `PDO::ERRMODE_EXCEPTION` e avremo errore nella query SQL. Crea un file PHP chiamato `error.php` e prova questo:

```
<?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;
?>
```

L'output di `error.php` è:

```
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
```

Questa volta impostando il valore `ATTR_ERRMODE` con `PDO::ERRMODE_EXCEPTION`, verrà generato `PDOException` che genera un errore irreversibile. Lo script smetterà di essere eseguito dopo che l'eccezione è stata lanciata. Questo è l'approccio più comune per gestire gli errori relativi alle query del database. Nella maggior parte del tempo, impostiamo l'attributo `ATTR_ERRMODE` con questo valore per gestire qualsiasi eccezione che potrebbe risiedere nella query SQL.

Leggi Iniziare con pdo online: <https://riptutorial.com/it/pdo/topic/4393/iniziare-con-pdo>

Capitolo 2: Prepara ed esegui le tue istruzioni SQL

Osservazioni

avvertimento

L'istruzione preparata non può interessare un parametro jolly per i nomi delle tabelle. Ad esempio questa seguente affermazione non è corretta:

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

La query preparata corretta sarebbe:

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

Examples

uso

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

Leggi [Prepara ed esegui le tue istruzioni SQL online](https://riptutorial.com/it/pdo/topic/6511/prepara-ed-esegui-le-tue-istruzioni-sql):

<https://riptutorial.com/it/pdo/topic/6511/prepara-ed-esegui-le-tue-istruzioni-sql>

Titoli di coda

S. No	Capitoli	Contributors
1	Iniziare con pdo	arsho , Community , hjpotter92 , Ilyas Mimouni , manian , TheCrazyProfessor , Your Common Sense
2	Prepara ed esegui le tue istruzioni SQL	Anwar Nairi , Your Common Sense