

# Chương 4 : PHP & MySQL

Giảng Viên: ThS. Tạ Việt Phương

1

1

---

---

---

---

---

---

---

## Lập trình với CSDL

Phát triển ứng dụng Web 2

2

---

---

---

---

---

---

---

### Nhu cầu lưu trữ dữ liệu của chương trình

- Một chương trình đầy đủ chức năng gồm 3 thành phần xử lý chính là :
  - Thành phần giao diện:** Xử lý các chức năng giao tiếp với người dùng
  - Thành phần xử lý:** Thực hiện các xử lý, quy định, kiểm tra, ... chính của chương trình
  - Thành phần dữ liệu:** Quản lý việc lưu trữ, đọc, ghi dữ liệu làm việc.

Phát triển ứng dụng Web

3

---

---

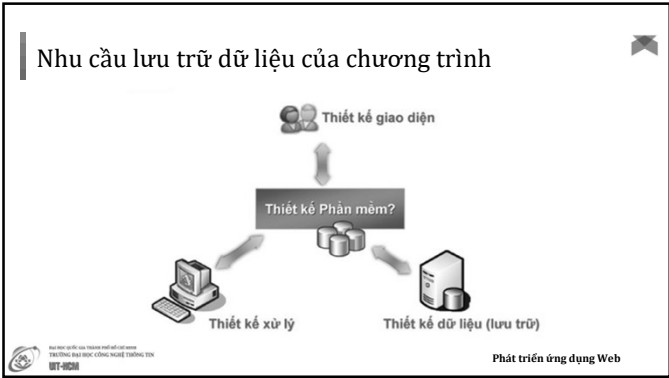
---

---

---

---

---



4

---

---

---

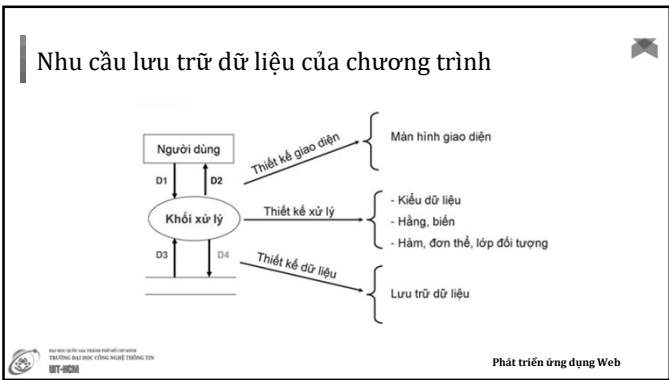
---

---

---

---

---



5

---

---

---

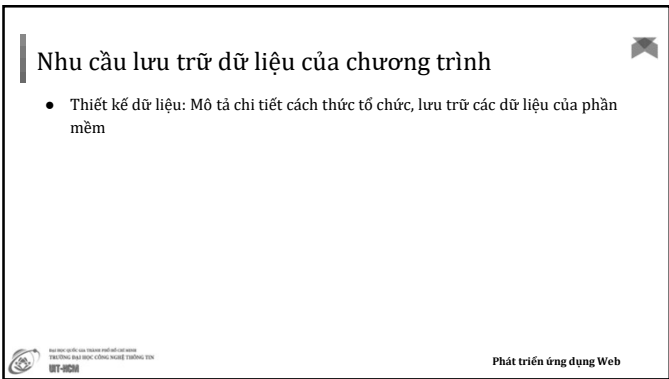
---

---

---

---

---



6

---

---

---

---

---

---

---

---

## Dữ liệu (Data)

- **(Trong KHMT)** Là các chỉ số của đối tượng (người, vật, một khái niệm, sự việc...) được lưu trữ trên máy tính.
- Dữ liệu được mô tả dưới nhiều dạng khác nhau (các ký tự, ký số, hình ảnh, ký hiệu, âm thanh...).
- Dữ liệu về đối tượng có thể khác nhau, tùy thuộc vào ngữ cảnh

**Ví dụ:** dữ liệu về SV có thể khác nhau tùy vào mục đích quản lý:

Quản lý điểm: Tên, mã sinh viên, điểm môn 1, điểm môn 2, điểm môn 3.

Quản lý nhân thân: Tên, địa chỉ, ngày sinh, quê quán, lớp



Phát triển ứng dụng Web

7

## Dữ liệu - Thông tin - Kiến thức

- Bản thân dữ liệu thường không có ý nghĩa trực tiếp đối với con người.
- **Thông tin** (Information) là tất cả những gì mà con người có thể cảm nhận được một cách trực tiếp thông qua các giác quan của mình hoặc gián tiếp thông qua các phương tiện kỹ thuật như tivi, radio, cassette,... Thông tin đối với con người luôn có một ý nghĩa nhất định nào đó. Khi dữ liệu được đưa vào ngữ cảnh, trong tầm hiểu biết của một người quan sát nhận thức, nó sẽ trở thành một thông tin.
- Thông tin là những hiểu biết của con người về một thực thể nào đó, có thể thu thập, lưu trữ, xử lý được. Dữ liệu được dùng để biểu diễn thông tin.



Phát triển ứng dụng Web

8

## Quản lý dữ liệu

- Quản lý dữ liệu là quản lý một số lượng lớn dữ liệu, bao gồm cả việc lưu trữ và cung cấp cơ chế cho phép Thao tác (thêm, sửa, xóa dữ liệu) và Truy vấn dữ liệu.
- Hai phương pháp quản lý dữ liệu:
  - Hệ thống quản lý bằng file
  - Hệ thống quản lý bằng CSDL



Phát triển ứng dụng Web

9

## Quản lý dữ liệu

- **Quản lý dữ liệu bằng file**
  - Dữ liệu được lưu trữ trong các file riêng biệt. Ví dụ: các chương trình lưu trữ thông tin bằng hệ thống các file dạng text.
  - Nhược điểm của việc quản lý bằng file:
    - Dư thừa và mâu thuẫn dữ liệu.
    - Kém hiệu quả trong truy xuất ngẫu nhiên hoặc xử lý đồng thời.
    - Dữ liệu lưu trữ rời rạc.
    - Gặp vấn đề về an toàn và bảo mật
- **Quản lý dữ liệu bằng CSDL**
  - Quản lý dữ liệu bằng CSDL giúp dữ liệu được lưu trữ một cách hiệu quả và có tổ chức, cho phép quản lý dữ liệu nhanh chóng và hiệu quả.

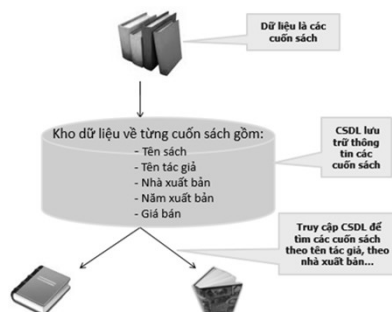
10

## Cơ sở dữ liệu

- **Cơ sở dữ liệu (CSDL - Database)** là một tập hợp dữ liệu được tổ chức và lưu trữ theo một cấu trúc chặt chẽ, thường mô tả hoạt động của một hay nhiều tổ chức liên quan và được lưu trữ trong máy tính
- CSDL được thiết kế, xây dựng cho phép người dùng lưu trữ dữ liệu, truy xuất thông tin hoặc cập nhật dữ liệu.
- CSDL được tổ chức có cấu trúc: Các dữ liệu được lưu trữ có cấu trúc thành các bản ghi (record), các trường dữ liệu (field). Các dữ liệu lưu trữ có mối quan hệ (relation) với nhau

11

## Cơ sở dữ liệu



12

## Hệ quản trị CSDL

- Hệ quản trị cơ sở dữ liệu(DataBase Management System – DBMS) là một hệ thống phần mềm, cung cấp các chức năng cho phép người dùng tạo các CSDL và quản lý các đối tượng trong một CSDL.
- Ví dụ:** Microsoft Access, Microsoft SQL Server, Oracle...
- Khác với CSDL, Hệ quản trị CSDL chỉ đóng vai trò quản lý, cho phép người dùng tạo ra các đối tượng như bảng, query, quan hệ,...

13

---

---

---

---

---

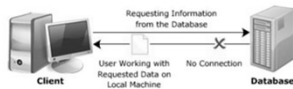
---

---

---

## Kiến trúc không kết nối

- Kiến trúc không kết nối
  - Kết nối được thiết lập và tạo một bản sao của CSDL ngay trên máy tính, sau đó kết nối được đóng
  - Thao tác với bản sao và update lại CSDL khi cần thiết
  - Ưu điểm
    - Tiết kiệm tài nguyên
    - Bảo mật vì kết nối được mở trong thời gian ngắn



14

---

---

---

---

---

---

---

---

## Nội dung

- Giới thiệu MySQL
- Giới thiệu các cách kết nối PHP với MySQL
- Các bước thao tác với CSDL
- Các câu lệnh truy vấn CSDL
- Font chữ khi thao tác với MySQL

15

---

---

---

---

---

---

---

---

## Tổng quan về MySQL

- MySQL là một trong những **Hệ quản trị cơ sở dữ liệu quan hệ** phổ biến đang được sử dụng trên Web. Nó có sẵn miễn phí và dễ cài đặt.
- PHP làm việc được với hầu hết các hệ quản trị cơ sở dữ liệu như Oracle, MS SQL Server, ... nhưng thông dụng nhất là MySQL.



16

---

---

---

---

---

---

---

---

## Lịch sử của MySQL

- Công ty Thụy Điển MySQL AB phát triển MySQL vào năm 1994.
- Phiên bản đầu tiên của MySQL phát hành năm 1995
- Công ty Sun Microsystems mua lại MySQL AB trong năm 2008
- Năm 2010 tập đoàn Oracle thu tóm Sun Microsystems. Ngay lúc đó, đội ngũ phát triển của MySQL tách MySQL ra thành 1 nhánh riêng gọi là MariaDB. Oracle tiếp tục phát triển MySQL lên phiên bản 5.5.
- 2013 MySQL phát hành phiên bản 5.6
- 2015 MySQL phát hành phiên bản 5.7
- MySQL phiên bản mới nhất là phiên bản 9.0.1

17

---

---

---

---

---

---

---

---

## Tổng quan về MySQL

- **Ưu điểm của MySQL**
  - Dễ sử dụng, mạnh mẽ, nhanh chóng, an toàn và có thể mở rộng.
  - Miễn phí.
  - Chạy trên nhiều hệ điều hành: UNIX, Windows, Mac và các hệ điều hành khác.
  - Hỗ trợ SQL chuẩn.
  - Là giải pháp cơ sở dữ liệu lý tưởng cho cả các ứng dụng nhỏ và lớn.
  - Chứa các lớp bảo mật dữ liệu bảo vệ dữ liệu nhạy cảm

18

---

---

---

---

---

---

---

---

Tổng quan về MySQL

- Nhược điểm:
  - Giới hạn: Có hạn chế về chức năng mà một vài ứng dụng có thể cần.
  - Độ tin cậy: Cách các chức năng cụ thể được xử lý với MySQL (ví dụ tài liệu tham khảo, các giao dịch, kiểm toán,...) làm cho nó kém tin cậy hơn so với một số hệ quản trị cơ sở dữ liệu quan hệ khác.
  - Dung lượng hạn chế: Nếu số bản ghi lớn dần lên thì việc truy xuất dữ liệu là khá khó khăn, khi đó chúng ta sẽ phải áp dụng nhiều biện pháp để tăng tốc độ truy xuất dữ liệu như là chia tải database này ra nhiều server, hoặc tạo cache MySQL

Phát triển ứng dụng Web

19

---

---

---

---

---

---

---

---

MariaDB

- MariaDB là hệ quản trị cơ sở dữ liệu miễn phí được phát triển từ hệ quản trị cơ sở dữ liệu mã nguồn mở MySQL.
- MariaDB được phát triển nhằm thay thế công nghệ cơ sở dữ liệu MySQL, vì thế nó tương thích và cho một hiệu suất cao hơn so với MySQL.
- Nền móng cơ sở đầu tiên của MariaDB được phát triển bởi “trụ cột” của MySQL AB là Michael “Monty” Widenius. Năm 2008, sau khi Sun mua lại MySQL AB, Michael “Monty” Widenius rời khỏi MySQL AB và tiếp tục phát triển một hệ cơ sở quản trị mới của mình.
- Đầu năm 2009, Michael cùng với 1 vài đồng nghiệp khác bắt đầu tiến hành dự án chuyên sâu về công cụ lưu trữ MySQL, sau này trở thành MariaDB. Tên gọi MariaDB được đặt tên theo tên con gái út của Widenius – Maria. Sau nhiều lần nâng cấp và phát triển, hiện tại MariaDB đã ra mắt phiên bản mới nhất là MariaDB 11.5.

Phát triển ứng dụng Web

20

---

---

---

---

---

---

---

---

MariaDB

- Những ưu điểm lớn nhất của hệ quản trị này phải kể đến bao gồm:
  - Hoàn toàn miễn phí
  - MariaDB được thiết kế để hoàn toàn tương thích ngược với MySQL. Điều này có nghĩa là các ứng dụng chạy trên MySQL có thể dễ dàng chuyển sang MariaDB mà không cần thay đổi mã nguồn.
  - Khắc phục những hạn chế của MySQL
  - Bổ sung thêm nhiều Engine hơn, chẳng hạn như Aria cho giao dịch, ColumnStore để hỗ trợ Big Data và xử lý dữ liệu phân tích
  - Kết hợp cả SQL và NoSQL
  - Hỗ trợ tiếng Việt

Phát triển ứng dụng Web

21

---

---

---

---

---


---

---


---

## MariaDB

- Ngôn ngữ lập trình hỗ trợ của MySQL và MariaDB khá là tương đồng nhau
- MariaDB ra đời sau nên được dựa trên những nền tảng vốn có của MySQL nên sẽ kế thừa những ưu điểm và khắc phục những hạn chế của MySQL đã có
- MariaDB là sự tích hợp ưu điểm của cả 2 thư viện SQL và NoSQL nên tính năng lưu trữ được mở rộng hỗ trợ người dùng nhiều hơn



MariaDB  
Foundation



Phát triển ứng dụng Web

22

---

---

---

---

---

---

---

## Cài đặt và cấu hình, kết nối



Phát triển ứng dụng Web

23

---

---

---

---


---

---

---

## Cài đặt và cấu hình

- Cài đặt XAMPP
- Thiết lập MySQL và MariaDB:
  - **Bước 1:** Sau khi cài đặt XAMPP, mở XAMPP Control Panel và khởi động Apache và MySQL.
  - **Bước 2:** Kiểm tra cổng mặc định (3306) cho MySQL/MariaDB hoặc thay đổi nếu cần thiết.



Phát triển ứng dụng Web

24

---

---

---

---

---

---

---



Kết nối - Giới thiệu

- Hiện nay, có 2 phương thức để kết nối PHP với MySQL: MySQLi và PDO.
- Mặc định khi cài PHP thì cả 2 thư viện MySQLi và PDO đã được tích hợp.
- Lưu ý: thư viện MySQL đã không còn được sử dụng từ phiên bản PHP7 trở lên.

Đại học Khoa học  
TP. HCM

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web25

25

---

---

---

---

---

---

---

Các bước thao tác kết nối CSDL trong PHP

- Kết nối với CSDL
- Chọn CSDL
- Xây dựng câu truy vấn dữ liệu
- Thực thi câu truy vấn
- Xử lý kết quả trả về từ câu truy vấn SELECT, INSERT, UPDATE, DELETE
- Ngắt kết nối với CSDL

Đại học Khoa học  
TP. HCM

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web26

26

---

---

---

---

---

---

---

Sử dụng MySQLi

Đại học Khoa học  
TP. HCM

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web27

27

---

---

---

---


---

---

---

### Giới thiệu

- MySQLi (MySQL Improved):
  - Là dạng kết nối thường được sử dụng
  - Được tích hợp sẵn trong PHP từ phiên bản 5.0
  - Tận dụng được các tính năng mới của MySQL phiên bản 4.1.3 trở lên.
  - Hỗ trợ giao tiếp theo hướng thủ tục (procedural) và hướng đối tượng (object-oriented).
  - Xem dữ liệu như là bộ các đối tượng với functions, người dùng thêm xóa dữ liệu dễ dàng.
  - Hỗ trợ dạng **prepared statements**, tăng cường khả năng chống SQL Injection
  - Nhược điểm: không làm việc được với các hệ quản trị CSDL khác, ngoài MySQL

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web28

28

---

---

---

---

---


---

---

---

### Sử dụng MySQLi (dạng procedural)

- Sử dụng các hàm của MySQLi để thao tác trên CSDL
  - mysqli\_connect
  - mysqli\_select\_db
  - mysqli\_query
  - mysqli\_num\_rows
  - mysqli\_fetch\_array
  - mysqli\_close

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web29

29

---

---

---

---


---

---

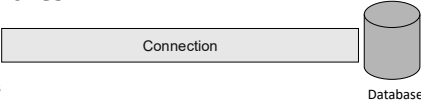
---

---

### Tạo kết nối CSDL



PHP Pages



Connection


Database

- Kết nối MySQL và chọn CSDL

```
resource mysqli_connect(string host, string LoginName, string password)
int mysqli_select_db(string dbName, resource connection)
```

Hoặc

```
resource mysqli_connect(string host, string LoginName, string password,
string dbName)
```

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web30

30

---

---

---

---

---

---

---

---

Tạo kết nối CSDL

• Cách 1

```
<?php
$SERVER = "localhost";
$USERNAME = "root";
$PASSWORD = "";
$DBNAME = "qlsv";
$conn = mysqli_connect($SERVER, $USERNAME, $PASSWORD);
if ( !$conn ) {
    //Không kết nối được, thoát ra và báo lỗi
    die("Không kết nối được vào MySQL server");
} //end if
echo "Kết nối thành công";
//chọn CSDL để làm việc
mysqli_select_db($conn, $DBNAME);
//đóng kết nối
mysqli_close($conn);
```

Phát triển ứng dụng Web

31

31

---

---

---

---

---

---

---

---

Tạo kết nối CSDL

• Cách 1

```
<?php
$SERVER = "localhost";
$USERNAME = "root";
$PASSWORD = "";
$DBNAME = "qlsv";
$conn = mysqli_connect($SERVER, $USERNAME, $PASSWORD);
//Không kết nối được, thoát ra và báo lỗi
Or die("Không kết nối được vào MySQL server" .mysqli_connect_error());

echo "Kết nối thành công";
//chọn CSDL để làm việc
mysqli_select_db($conn, $DBNAME);
//đóng kết nối
mysqli_close($conn);

?>
```

Phát triển ứng dụng Web

32

32

---

---

---

---

---

---

---

---

Tạo kết nối CSDL

• Cách 2

```
<?php
$SERVER = "localhost";
$USERNAME = "root";
$PASSWORD = "";
$DBNAME = "qlsv";
$conn= mysqli_connect($SERVER, $USERNAME, $PASSWORD,$DBNAME,3306) ;
if ( !$conn ) {
    //Không kết nối được, thoát ra và báo lỗi
    die("Không kết nối được vào MySQL server");
} //end if
echo "Kết nối thành công";
//đóng kết nối
mysqli_close($conn);
```

Phát triển ứng dụng Web

33

33

---

---

---

---

---

---

---


---

Tạo kết nối CSDL

- Cách 2

```
<?php
$SERVER = "localhost";
$USERNAME = "root";
$PASSWORD = "";
$dbname = "qlsv";
$PORT = "3306";
$conn= mysqli_connect($SERVER, $USERNAME, $PASSWORD, $dbname, $PORT);
//Không kết nối được, thoát ra và báo lỗi
Or die("Không kết nối được vào MySQL server");

echo "Kết nối thành công";
//đóng kết nối
mysqli_close($conn);
?>
```



Đại học Công nghệ Thông tin  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web

34

---

---

---

---

---

---


---

---

34

Xây dựng câu truy vấn

SQL	Ví dụ
INSERT	<code>strSQL = "INSERT INTO Users (UserName, Password) VALUES ('admin','123')"</code>
DELETE	<code>strSQL = "DELETE FROM Users WHERE UserName = 'lvminh'"</code>
UPDATE	<code>strSQL = "UPDATE Users SET Password = 'abc' WHERE UserName = 'admin'"</code>
SELECT	<code>strSQL = "SELECT * FROM Users WHERE UserName = 'admin'"</code>



Đại học Công nghệ Thông tin  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web

35

---

---

---

---

---


---

---

---

35

Thực thi câu truy vấn




PHP Pages

011010011

Connection

011010011



Database

`resource mysqli_query(resource connection, string strSQL)`

```
<?php
$strSQL = "INSERT INTO Users (UserName, Password) VALUES ('admin','123')";
mysqli_query($conn, $strSQL);
?>
```


```
<?php
$strSQL = "SELECT * FROM Users";
$result = mysqli_query($conn, $strSQL);
?>
```

Insert

delete

update

select



Đại học Công nghệ Thông tin  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web

36

---

---

---

---

---

---

---

---

36

12

Thực thi câu truy vấn

- Giá trị trả về của hàm `mysqli_query`
  - Không trả về giá trị với câu lệnh sql `CREATE DATABASE` và `CREATE TABLE`

```
$strSQL = "Create table User (id int AUTO_INCREMENT primary key, Username varchar(20), Password varchar(30))";  
mysqli_query($conn, $strSQL);
```
  - Đối với câu lệnh `SELECT` trả về kết quả thu được

```
$result = mysqli_query($conn, $strSQL);
```
  - Đối với câu lệnh `INSERT`, `UPDATE`, `DELETE` trả về giá trị 1 (true), 0 (false)

Phát triển ứng dụng Web

37

37

---

---

---

---

---

---

---

---

Xử lý kết quả trả về

- Hiện thị dữ liệu trả về
  - Sử dụng hàm

```
$row = mysqli_fetch_array($result)  
$row = mysqli_fetch_assoc($result)  
$row = mysqli_fetch_row($result)
```
  - Số lượng FIELD, ROW

```
$num = mysqli_num_fields($result);  
$num = mysqli_num_rows($result);
```
  - Truy cập đến từng FIELD

```
echo $row["UserName"];
```
  - Hoặc

```
echo $row[0];
```

UserName	Password
A	AA
B	BB
C	CC
D	DD
...	...
X	XX
Y	YY
Z	ZZ

cursor

Phát triển ứng dụng Web

38

38

---

---

---

---

---

---

---

---

Xử lý kết quả trả về

- ```
$rowArray = mysqli_fetch_array($result, MYSQLI_ASSOC | MYSQLI_NUM | MYSQLI_BOTH)  
while ($rowArray = mysqli_fetch_array($result)){  
    echo $rowArray[0] . $rowArray["UserName"];  
}
```
- ```
$rowAssoc = mysqli_fetch_assoc($result)  
while ($rowAssoc = mysqli_fetch_assoc($result)){  
    echo $rowAssoc["UserName"];  
}
```
- ```
$row = mysqli_fetch_row($result)  
while ($row = mysqli_fetch_row($result)){  
    echo $row[1];  
}
```

Phát triển ứng dụng Web

39

39

---

---

---

---

---

---

---

---

```
<?php
$server = "localhost";
$username = "root";
$password = "";
$dbname = "qlsv";
$conn = mysqli_connect($server, $username, $password, $dbname)
Or die("<p>Không thể connect</p>");
mysqli_set_charset($conn, "utf8");
$sql = "select * from User";
$result = mysqli_query($conn, $sql);
Or die("<p>Không thể thực thi câu truy vấn.</p>");
echo "<table border = 1 >";
echo "<tr><th>Username</th><th>Password</th></tr>";
while ($row = mysqli_fetch_row($result)) {
    echo "<tr>";
    echo "<td>" . $row[0] . "</td>";
    echo "<td>" . $row[1] . "</td>";
    echo "</tr>";
};
echo "</table>";
//Bóng kết nối
mysqli_close($conn);
?>
```



Phát triển ứng dụng Web 40

40

---

---

---

---

---

---

---

---

---

---

## Ngắt kết nối dữ liệu

- Ngắt kết nối cơ sở dữ liệu  
`mysqli_close($conn);`
- Tự động thực thi khi kết thúc mã lệnh nhờ cơ chế tự động thu rác (garbage collector) – PHP4,5
- Giải phóng vùng nhớ chứa dữ liệu sau khi truy vấn  
`mysqli_free_result($result);`

Phát triển ứng dụng Web 41

41

---

---

---

---

---

---

---

---

---

---

## Tổng hợp các bước thao tác CSDL MySQL

```
<?php
$SERVER = "localhost";
$USERNAME = "root";
$PASSWORD = "";
$dbname = "mydata";
$conn = mysqli_connect($SERVER, $USERNAME, $PASSWORD);
if ( !$conn ) {
    //Không kết nối được, thoát ra và báo lỗi
    die("Không kết nối được MySQL server:". mysqli_connect_error());
} //end if
//chọn CSDL để làm việc
mysqli_select_db($conn, $dbname)
or die("Không thể chọn được CSDL: ". mysqli_error($conn));
?>
```

Phát triển ứng dụng Web 42

42

---

---

---

---

---

---

---

---

---

---

## Tổng hợp các bước thao tác CSDL MySQL

- Sử dụng câu lệnh `mysqli_query($conn, $sql);`
- Thêm:
 

```
$strSQL = "Insert into User values('Mai Xuân Hùng', '0901')";
$result = mysqli_query($conn, $strSQL)
  Or die("<p>Không thể thực thi câu truy vấn.</p>"
    . "<p>Error code " . mysqli_errno($conn)
    . ": " . mysqli_error($conn) . "</p>");
echo "Thêm thành công";
```

43

---

---

---

---

---

---

---

---

## Tổng hợp các bước thao tác CSDL MySQL

- Sửa, Xóa
 

```
$strSQL = "Update User Set Username = 'Mai Xuân Đền' where Password = '0901'";
$result = mysqli_query($conn, $strSQL)
  Or die("<p>Không thể thực thi câu truy vấn.</p>"
    . "<p>Error code " . mysqli_errno($conn)
    . ": " . mysqli_error($conn) . "</p>");
echo "Cập nhật thành công" . mysqli_affected_rows($conn) . "dòng" ;
```

44

---

---

---

---

---

---

---

---

## Tổng hợp các bước thao tác CSDL MySQL

- Câu lệnh select
 

```
$sql = "SELECT * FROM HS";
$result = mysqli_query($conn, $sql);
if ( !$result )
  die("Không thể thực hiện được câu lệnh SQL: " . mysqli_error($conn));
echo "Số lượng row tìm được: " . mysqli_num_rows($result) . "<br>";
while ( $row = mysqli_fetch_row($result) ) {
  echo "Ten: " . $row[0];
  echo " Ma so: " . $row[1] . "<br>";
} //end while
//nên luôn giải phóng bộ nhớ sau khi lấy hết các row trả về từ câu lệnh SELECT
mysqli_free_result($result);
//đóng kết nối
mysqli_close($conn);
```

45

---

---

---

---

---


---

---

---

Sử dụng Prepared statements

```
$sql = "SELECT * FROM HS where ID=?";
$id = 6;
if ( $stmt = mysqli_prepare($conn, $sql)){
    //Truyền dữ liệu vào tham số
    mysqli_stmt_bind_param($stmt, "i", $id)
    //Thực thi câu truy vấn
    mysqli_stmt_execute($stmt);
    //Gán dữ liệu trả về vào biến $result
    mysqli_stmt_bind_result($stmt, $result);
    //Thực hiện fetch dữ liệu
    mysqli_stmt_fetch($stmt);
    //in ra kết quả trả về
    printf('Result: %s', $result);
    //close mysqli_stmt
    mysqli_stmt_close($stmt);
}
```



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web

46

46

---

---

---

---

---


---

---

---

Sử dụng Prepared statements

- Tham số ẩn danh trong câu truy vấn dùng ký tự ?
- Hàm `mysqli_stmt_bind_param($stmt, "i", $id)`:
  - Giá trị thứ 2 "i" chính là khai báo kiểu dữ liệu cho các tham số ẩn danh trong câu truy vấn. Các giá trị như sau:
    - i: integer
    - d: double
    - s: string
    - b: blob
  - Nếu nhiều tham số ẩn danh thì sẽ khai báo kiểu cho đủ số tham số ẩn danh. Vd: câu truy vấn có 2 tham số ẩn danh kiểu string: khai báo "ss"
  - Tham số cuối cùng (\$id) là các tham số dạng tham chiếu, không cần phải được khai báo trước khi gọi hàm.



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web

47

47

---

---

---

---

---

---


---

---

Sử dụng Prepared Statements

- Câu lệnh Insert

```
// Chuẩn bị câu lệnh INSERT
$sql = "INSERT INTO persons (first_name, last_name) VALUES (?, ?)";
if($stmt = mysqli_prepare($conn, $sql)){
    // Liên kết biến với tham số trong câu lệnh đã chuẩn bị
    mysqli_stmt_bind_param($stmt, "ss", $first_name, $last_name);
    // Thiết lập tham số lần 1
    $first_name = "abc";
    $last_name = "xyz";
    // Thực hiện câu lệnh
    mysqli_stmt_execute($stmt);
    // Thiết lập tham số lần 2
    $first_name = "efg";
    $last_name = "vmk";
    // Thực hiện câu lệnh
    mysqli_stmt_execute($stmt);
}
```



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web

48

48

---

---

---

---

---

---

---

---



Sử dụng MySQLi (dạng object oriented)

- Sử dụng các hàm của đối tượng MySQLi để thao tác trên CSDL

ĐẠI HỌC QUỐC GIA VIỆT NAM

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web

49

49

---

---

---

---

---

---

---

Tạo kết nối CSDL

PHP Pages

Connection

Database

- Khởi tạo đối tượng để kết nối

```
$conn = new mysqli("hostname", "username", "password", "database");
```

- Kiểm tra kết nối

```
if ($conn->connect_error) {  
    die("Kết nối lỗi: " . $conn->connect_error);  
}  
echo "Kết nối thành công!!!";
```

ĐẠI HỌC QUỐC GIA VIỆT NAM

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web

50

50

---

---

---

---

---

---

---

Tạo kết nối CSDL

```
<?php  
$SERVER = "localhost";  
$USERNAME = "root";  
$PASSWORD = "";  
$DBNAME = "qlsv";  
$conn = new mysqli($SERVER, $USERNAME, $PASSWORD, $DBNAME);  
if ($conn->connect_error) {  
    die("Kết nối lỗi: " . $conn->connect_error);  
}  
echo "Kết nối thành công!!!";  
?>
```

ĐẠI HỌC QUỐC GIA VIỆT NAM

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web

51

51

---

---

---

---

---

---

---

Thực thi câu truy vấn Insert

- Sử dụng hàm **query** của đối tượng **mysqli**

```
$strSQL = "Insert into User values('Mai Xuân Hùng', '0901')";
if ($conn->query($strSQL) === true)
    echo "Thêm thành công!!!";
}else echo "Thêm không thành công!!!";
```

Đại học Khoa học và Công nghệ Hà Nội

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web

52

52

---

---

---

---

---

---

---

---

Thực thi câu truy vấn Delete

- Sử dụng hàm **query** của đối tượng **mysqli**

```
$strSQL = "Delete from User where Password = '0901'";
if ($conn->query($strSQL) === true)
    echo "Xóa thành thành công!!!";
}else echo "Xóa không thành công!!!";
```

Đại học Khoa học và Công nghệ Hà Nội

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web

53

53

---

---

---

---

---

---

---

---

Thực thi câu truy vấn Update

- Sử dụng hàm **query** của đối tượng **mysqli**

```
$strSQL = "Update User Set Username = 'Mai Xuân Đến' where Password = '0901'";
if ($conn->query($strSQL) === true)
    echo "Cập nhật thành thành công!!!";
}else echo "Cập nhật không thành công!!!";
```

Đại học Khoa học và Công nghệ Hà Nội

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web

54

54

---

---

---

---

---

---


---

---

### Thực thi câu truy vấn Select

- Sử dụng hàm `query` của đối tượng `mysqli`

```
$strSQL = "select * from User";  
$result = $conn->query($strSQL)
```



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UIT-HCM

Phát triển ứng dụng Web55

---

---

---

---

---

---

---

---

### Thực thi câu truy vấn Select

- Hiển thị dữ liệu trả về
  - Sử dụng các hàm của biến `$result`

```
$row = $result->fetch_array();  
$row = $result->fetch_assoc();  
$row = $result->fetch_row();
```
  - Số lượng FIELD, ROW


```
$num = $result->num_fields();  
$num = $result->num_rows();
```
  - Truy cập đến từng FIELD

```
echo $row["UserName"];
```
  - Hoặc

```
echo $row[0];
```

| UserName | Password |
|----------|----------|
| A        | AA       |
| B        | BB       |
| C        | CC       |
| D        | DD       |
| ...      | ...      |
| X        | XX       |
| Y        | YY       |
| Z        | ZZ       |

cursor



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UIT-HCM

Phát triển ứng dụng Web56

---

---

---

---

---


---

---

---

### Xử lý kết quả trả về

- ```
$rowArray = $result->fetch_array(int $resulttype = MYSQLI_ASSOC |  
MYSQLI_NUM | MYSQLI_BOTH)  
while ($rowArray = $result->fetch_array()){  
    echo $rowArray[0] . $rowArray["UserName"];  
}  
$rowAssoc = $result->fetch_assoc()  
while ($rowAssoc = $result->fetch_assoc()){  
    echo $rowAssoc["UserName"];  
}  
$row = $result->fetch_row()  
while ($row = $result->fetch_row()){  
    echo $row[1];  
}
```



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UIT-HCM

Phát triển ứng dụng Web57

---

---

---

---

---


---

---

---

Sử dụng Prepared Statements

```
$sql = "SELECT * FROM HS where ID=?";
$id = 6;
$stmt = $conn->prepare($sql)
//Truyền dữ liệu vào tham số
$stmt->bind_param("i", $id)
//Thực thi câu truy vấn
$stmt->execute();
//Gán dữ liệu trả về vào biến $result
$stmt->bind_result($result);
//Thực hiện fetch dữ liệu
$stmt->fetch();
//in ra kết quả trả về
printf('Result: %s', $result);
```



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web

58

58

---

---

---

---

---


---

---

---

Ngắt kết nối dữ liệu

- Giải phóng vùng nhớ chứa dữ liệu sau khi truy vấn  
`$result->close();`
- Ngắt kết nối cơ sở dữ liệu  
`$conn->close();`



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web

59

59

---

---

---

---


---

---

---

---

Sử dụng PDO



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web

60

60

---

---

---

---

---


---

---

---

## Giới thiệu

- **PDO (PHP Data Object):**
  - Hỗ trợ 12 loại hệ quản trị CSDL khác nhau.
  - Là dạng database abstraction layer; sử dụng các extension để giao tiếp với các CSDL khác nhau.
  - Mỗi hệ quản trị khác nhau sẽ có các driver khác nhau để chuyển đổi sang câu lệnh SQL tương ứng.
  - Là dạng hướng đối tượng (object oriented)
  - Hỗ trợ dạng **prepared statements** và **name parameters** giúp tăng cường khả năng chống SQL Injection
  - PDO được ưu tiên trong các dự án lớn nhờ vào tính linh hoạt và khả năng hỗ trợ nhiều loại cơ sở dữ liệu. Khi dự án có yêu cầu khả năng mở rộng và dễ dàng chuyển đổi qua các cơ sở dữ liệu khác, PDO là lựa chọn tốt hơn.



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web

61

61

---

---

---

---

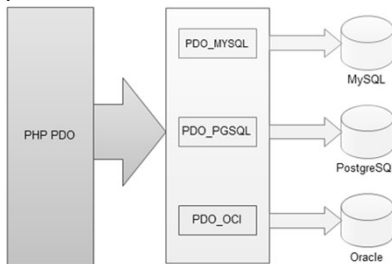
---

---

---

---

## Giới thiệu



```
graph LR; PHP_PDO[PHP PDO] --> PDO_MYSQL[PDO_MYSQL]; PHP_PDO --> PDO_PGSQL[PDO_PGSQL]; PHP_PDO --> PDO_OCI[PDO_OCI]; PDO_MYSQL --> MySQL[(MySQL)]; PDO_PGSQL --> PostgreSQL[(PostgreSQL)]; PDO_OCI --> Oracle[(Oracle)];
```



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web

62

62

---

---

---

---

---

---

---

---

## Tạo kết nối CSDL

- Mỗi hệ quản trị CSDL sẽ có phương thức kết nối khác nhau (có loại cần Username, Password, đường dẫn tới Database, Port, có loại không)
- Với MySQL:

```
$conn = new PDO("mysql:host=localhost;dbname=database", 'username', 'password');
```



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web

63

63

---

---

---

---

---

---

---

---

Tạo kết nối CSDL

- VD:

```
<?php
$SERVER = "localhost";
$USERNAME = "root";
$PASSWORD = "";

try{
    $conn = new PDO("mysql:host=$SERVER;dbname=qlsv", $USERNAME, $PASSWORD);
    // Thiết lập lỗi của PDO cho ngoại lệ
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    echo "Kết nối thành công!!!";
}catch(PDOException $e) {
    echo "Kết nối thành công!!!". $e->getMessage();
}
?>
```

Phát triển ứng dụng Web 64

64

---

---

---

---

---

---

---

---

Thực thi câu truy vấn Select

- Khi thực thi câu truy vấn select, PDO trả dữ liệu về theo mảng (array) hoặc đối tượng (object) bằng phương thức **fetch()**.
  - PDO::FETCH\_ASSOC**: Trả về dữ liệu dạng mảng với key là tên thuộc tính trong bảng của câu truy vấn select.
  - PDO::FETCH\_BOTH (default)**: Trả về dữ liệu dạng mảng với key là tên và cả số thứ tự của thuộc tính
  - PDO::FETCH\_BOUND**: Gán giá trị của từng thuộc tính cho từng biến đã khởi tạo trước đó qua phương thức bindColumn()
  - PDO::FETCH\_CLASS**: Gán giá trị của từng thuộc tính (column) của bảng cho từng thuộc tính (property/attribute) của một lớp Class theo tên column và tên thuộc tính.
  - PDO::FETCH\_INTO**: Gán giá trị của từng column cho từng thuộc tính của một Class Instance (thể hiện của một lớp)

Phát triển ứng dụng Web 65

65

---

---

---

---

---

---

---

---

Thực thi câu truy vấn Select

- PDO::FETCH\_LAZY**: Gộp chung PDO::FETCH\_BOTH/PDO::FETCH\_OBJ
- PDO::FETCH\_NUM**: Trả về dữ liệu dạng mảng với key là số thứ tự của thuộc tính
- PDO::FETCH\_OBJ**: Trả về một Object của stdClass (link is external) với tên thuộc tính của Object là tên của column.

- Cần thiết lập dạng PDO Fetch trước khi sử dụng bằng: **setFetchMode**
- VD: `$stmt->setFetchMode(PDO:::FETCH_ASSOC);`
- Các kiểu fetch thường dùng: **FETCH\_ASSOC**, **FETCH\_CLASS** và **FETCH\_OBJ**

Phát triển ứng dụng Web 66

66

---

---

---

---

---

---

---

---

## Thực thi câu truy vấn Select (FETCH\_ASSOC)

```
//Tạo Câu truy vấn và thực thi
$stmt = $conn->query('SELECT * From USER');
//Thiết lập kiểu dữ liệu trả về
$stmt->setFetchMode(PDO::FETCH_ASSOC);
//Hiển thị kết quả
while($row = $stmt->fetch()) {
    echo $row['username'] , '\n';
    echo $row['password'];
}
```



Phát triển ứng dụng Web 67

67

---

---

---

---

---

---

---

---

## Thực thi câu truy vấn Select (FETCH\_OBJ)

- Kết quả trả về là một object của stdClass cho mỗi dòng kết quả

```
//Tạo câu truy vấn và thực thi
$stmt = $conn->query('SELECT * From USER');
//Thiết lập kiểu dữ liệu trả về
$stmt->setFetchMode(PDO::FETCH_OBJ);
//Hiển thị kết quả
while($row = $stmt->fetch()) {
    echo $row->username, '\n';
    echo $row->password;
}
```



Phát triển ứng dụng Web 68

68

---

---

---

---

---

---

---

---

## Thực thi câu truy vấn Select (FETCH\_CLASS)

- Kết quả trả về là một object theo class được chỉ định (ảnh xạ đối tượng)
- Thuộc tính của class sẽ được gán trước khi hàm khởi tạo của class đó được gọi.
- Nếu **không** có thuộc tính **khớp** với tên cột trong bảng thì tự động **tạo** một thuộc tính public trong class

```
class User {
    public $username;
    public $password;
    public $isAdmin = 'No';
    function __construct() {
        if ($this->name == 'Microsoft Window')
            $this->isAdmin = 'Yes';
    }
    public function display(){
        return $this->username. ': '. $this->password. ' là admin '. $this->isAdmin;
    }
}
```



Phát triển ứng dụng Web 69

69

---

---

---

---

---

---

---

---

Thực thi câu truy vấn Select (FETCH\_CLASS)

```
//Tạo câu truy vấn và thực thi
$stmt = $conn->query('SELECT * From USER);
//Thiết lập kiểu dữ liệu trả về, chỉ định đưa dữ liệu vào object của class User
$stmt->setFetchMode(PDO::FETCH_CLASS, 'User');
//Hiển thị kết quả
while($obj = $stmt->fetch()) {
    echo $obj->username.'\n';
    echo $obj->password.'\n';
    echo $obj->isAdmin.'\n';
    Hoặc echo $obj->display().'\n';
}
```

Phát triển ứng dụng Web

70

70

---

---

---

---

---

---

---

---

Thực thi câu truy vấn Select (FETCH\_CLASS)

- Kết quả trả về là một object theo class được chỉ định.
- Để constructor của class được gọi trước khi các thuộc tính được gán giá trị, sử dụng thêm **PDO::FETCH\_PROPS\_LATE**

```
$stmt->setFetchMode(PDO::FETCH_CLASS | PDO::FETCH_PROPS_LATE, 'User');
```

- Để truyền tham số cho constructor của class thông qua fetch(): đặt các tham số trong một array theo thứ tự tương ứng

```
$stmt->setFetchMode(PDO::FETCH_CLASS, 'User', array('param1', 'param2', 'param3'));
```

Phát triển ứng dụng Web

71

71

---

---

---

---

---

---

---

---

Thực thi câu truy vấn Insert, Update, Delete

- Đối với các câu lệnh Insert, Update, Delete (không có dữ liệu trả về) và không cần truyền tham số cho câu lệnh thì sử dụng phương thức **exec()** của đối tượng PDO
- VD:

```
$conn = new PDO("mysql:host=$SERVER;dbname=qlsv",$USERNAME,$PASSWORD);
// Thiết lập lỗi của PDO cho ngoại lệ
$conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
//Câu truy vấn
$sql = "DELETE FROM USER WHERE id = 6";
//thực hiện truy vấn
$conn->exec($sql);
```

Phát triển ứng dụng Web

72

72

---

---

---

---

---

---

---

---



### Sử dụng Prepared statements

- Sử dụng phương thức **prepare** của đối tượng PDO để thiết lập câu truy vấn

```
$stmt = $conn->prepare("INSERT INTO User (username, password) VALUES (:username, :pass)");
```
- Sử dụng **Name Parameters** để tạo liên kết tham số truyền dữ liệu và phương thức **bindParam** để gắn các biến giá trị vào các tham số.

```
$stmt->bindParam(':username', $username);  
$stmt->bindParam(':pass', $pass);
```


Hoặc dùng mảng:

```
$params = array(':username' => 'test', ':pass' => '1234')
```
- Dùng phương thức **execute** để thực thi truy vấn

```
$stmt->execute();
```

Hoặc: 

```
$stmt->execute($param);
```



Đại học Khoa học và Công nghệ Hà Nội

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web73

---

---

---

---

---

---


---

---

73

### Sử dụng Prepared Statements (Insert)

```
// Chuẩn bị câu lệnh INSERT  
$sql = "INSERT INTO Persons (first_name, last_name) VALUES (:firstname, :lastname)";  
$stmt = $conn->prepare($sql)  
// Liên kết biến với tham số trong câu lệnh đã chuẩn bị  
$stmt->bindParam(':firstname', $first_name);  
$stmt->bindParam(':lastname', $last_name);  
// Thiết lập tham số lần 1  
$first_name = "abc";  
$last_name = "xyz";  
// Thực hiện câu lệnh  
$stmt->execute();  
// Thiết lập tham số lần 2  
$first_name = "efg";  
$last_name = "vmk";  
// Thực hiện câu lệnh  
$stmt->execute();
```



Đại học Khoa học và Công nghệ Hà Nội

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web74

---

---

---

---

---

---


---

---

74

### Sử dụng Prepared Statements (Select)

```
//Tạo Câu truy vấn dạng Prepared  
$stmt = $conn->prepare('SELECT * From USER where username = :name');  
//Thiết lập kiểu dữ liệu trả về  
$stmt->setFetchMode(PDO::FETCH_ASSOC);  
// Liên kết biến với tham số trong câu lệnh đã chuẩn bị  
$stmt->bindParam(':name', 'abc');  
// Thực hiện câu lệnh  
$stmt->execute();  
//Hiển thị kết quả  
while($row = $stmt->fetch()) {  
    echo $row['username'] , '\n';  
    echo $row['password'];  
}
```



Đại học Khoa học và Công nghệ Hà Nội

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UT-HCM

Phát triển ứng dụng Web75

---

---

---

---

---

---

---


---

75

Ngắt kết nối dữ liệu

- Ngắt kết nối cơ sở dữ liệu

```
$conn = null;
```



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UIT-HCM

Phát triển ứng dụng Web76

76

---

---

---

---


---

---

---

So sánh MySQLi và PDO

	MySQLi	PDO
CSDL hỗ trợ	Chỉ MySQL/MariaDB	12 loại CSDL
API	OOP + Procedural	OOP
Kết nối	Đơn giản	Đơn giản
Named parameters	Không	Có
Prepared statements	Có	Có
Ảnh xạ đối tượng	Có	Có
Stored procedures	Có	Có
Hiệu năng	Cao	Cao



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UIT-HCM

Phát triển ứng dụng Web77

77

---

---

---

---

---

---

---

Tiếng Việt trong PHP và MySQL



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

UIT-HCM

Phát triển ứng dụng Web78

78

---

---

---

---

---

---

---

```

<?php
$server = "localhost";
$username = "root";
$password = "";
$dbname = "qlsv";
$conn = mysqli_connect($server, $username, $password, $dbname)
    Or die("<p> Không thể connect</p>");
$sql = "select * from User";
$result = mysqli_query($conn, $sql)
    Or die("<p>Không thể thực thi câu truy vấn.</p>");
echo "<table border =1 >";
echo "<tr><th>Username</th><th>Password</th></tr>";
while ($row = mysqli_fetch_row($result)) {
    echo "<tr>";
    echo "<td>" . $row[1] . "</td>";
    echo "<td>" . $row[2] . "</td>";
    echo "</tr>";
}
echo "</table>";
//Đóng kết nối
mysqli_close($conn);

```

user name	password
Mai Xuān Hāng	0901
Thị Bā Thān	0902
Nguyễn Hāu ?t	0903
Nguyễn Vān Ph?c	0904

Học viện Công nghệ Việt Nam  
 TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web

79

---

---

---

---

---

---

---

---

---

---

## MySQL & Font Unicode

- Check trong MySQL console từ XAMPP Control Panel  
SHOW VARIABLES LIKE 'character\_set\_server';  
SHOW VARIABLES LIKE 'collation\_server';
- Nếu kết quả không phải là utf8mb4 và utf8mb4\_general\_ci, có thể thay đổi như sau (utf8mb4 là phiên bản đầy đủ của UTF-8 hỗ trợ tất cả các ký tự, bao gồm emoji)
- Chỉnh sửa file cấu hình my.cnf hoặc my.ini của MySQL (nằm trong thư mục xampp/mysql/bin/my.ini). Sau khi chỉnh sửa, khởi động lại MySQL từ XAMPP Control Panel để các thay đổi có hiệu lực  
[mysqld]  
character-set-server = utf8mb4  
collation-server = utf8mb4\_general\_ci

Học viện Công nghệ Việt Nam  
 TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web

80

---

---

---

---

---

---

---

---

---

---

## MySQL & Font Unicode

- Tạo trong trình quản trị CSDL MySQL

Học viện Công nghệ Việt Nam  
 TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Phát triển ứng dụng Web

81

---

---

---

---

---

---

---

---

---

---

## MySQL & Font Unicode

- Dùng câu lệnh SQL để tạo

- Thiết lập thuộc tính Database: Khi tạo CSDL MySQL thiết lập **CHARACTER SET utf8 COLLATE utf8\_general\_ci**;

```
CREATE DATABASE sinhvien DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;
CREATE DATABASE sinhvien CHARACTER SET utf8mb4 COLLATE utf8mb4_general_ci;
```

- Hoặc thiết lập các thuộc tính của bảng

```
CREATE TABLE SinhVien (
    MaSV VARCHAR(8) NOT NULL,
    HoTen VARCHAR(50),
    NgaySinh DATE,
    PRIMARY KEY (MaSV)
) COLLATE utf8_general_ci;
```



82

## MySQL & Font Unicode

- Thiết lập các thuộc tính của field

```
CREATE TABLE SinhVien (
    MaSV VARCHAR(8) NOT NULL,
    HoTen VARCHAR(50) COLLATE utf8mb4_general_ci NOT NULL,
    NgaySinh DATE,
    PRIMARY KEY (MaSV)
);
```

**Chú ý:** Có thể sử dụng 1 trong 3 cách trên, tùy từng trường hợp lưu trữ; và có thể sử dụng COLLATE nào bắt đầu bằng utf8 đều được. (vd: *utf8\_unicode\_ci*, *utf8\_bin*,...)



83

## Thiết lập Unicode trong file .php

- Khai báo thẻ meta, charset=utf-8

```
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Trang quản lý thông tin SV</title>
</head>
```

- Đoạn code kết nối CSDL MySQL:

```
$conn = mysqli_connect($server, $user, $pass);
mysqli_set_charset($conn, "utf8"); // hoặc utf8mb4
```

Hoặc \$conn->set\_charset('utf8');

Hoặc PDO: \$pdo->exec('SET NAMES "utf8"');

- Các thẻ form phải có thuộc tính method = "post"



84

## Thiết lập Unicode trong file .php

### • PDO:

```
try {
    $conn = new PDO("mysql:host=localhost;dbname=mydb;charset=utf8mb4",
        "root", "");
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    echo "Connected successfully";
} catch (PDOException $e) {
    echo "Connection failed: " . $e->getMessage();
}
```

### • Thiết lập UTF-8 trong HTTP Header:

```
header('Content-Type: text/html; charset=utf-8');
```



85

---

---

---

---

---

---

---

---

---

---

## Bài tập

Xét lược đồ CSDL “Quản lý Tour du lịch” như sau:

### TOUR (MaTour, TenTour, NgayKhoiHanh, SoNgay, SoDem, Gia)

**Tên từ:** Mỗi tuyến du lịch cần lưu trữ mã Tour (MaTour), tên Tour (TenTour), ngày khởi hành (NgayKhoiHanh), bao nhiêu ngày (SoNgay), bao nhiêu đêm (SoDem), giá tour (Gia)

### TINH TP (MaTTP, TenTTP)

**Tên từ:** Lưu trữ tất cả các tỉnh (thành phố) trong nước, gồm có những thông tin: mã tỉnh (hoặc thành phố) (MaTTP), tên tỉnh thành phố (TenTTP).

### DIEMDL (MaDDL, TenDDL, MaTTP, DaTrung)

**Tên từ:** Mỗi điểm du lịch cần lưu trữ mã điểm du lịch (MaDDL), tên điểm du lịch (TenDDL) và điểm này thuộc tỉnh (thành phố) nào (MaTTP), đặc trưng (DaTrung) của điểm du lịch này là gì? (chỉ lưu 1 đặc trưng chính như: ‘Tắm biển’, ‘Leo núi’, ‘Mua sắm’, ‘Thăm quan’...).

### CHITiet (MaTour, MaDDL, Ngay, Dem)

**Tên từ:** Ghi nhận mỗi Tour (MaTour) đi qua các điểm du lịch (MaDDL) nào, bao nhiêu ngày (Ngay), bao nhiêu đêm (Dem) của mỗi địa điểm du lịch. Ngay, Dem có thể là số lẻ (ví dụ: 0.5, 1.5,...).



86

---

---

---

---

---

---

---

---

---

---

## Bài tập

➤ Dùng HTML, JavaScript, jQuery, php thiết kế và viết code xử lý cho các trang web sau:

### a. Trang thêm Tour du lịch (1.5 điểm)

Mã tour:

Tên tour:

Ngày khởi hành:

Số ngày:

Số đêm:

Giá:

Khi nhấn nút “Thêm” chương trình lấy thông tin từ các control và thêm vào bảng “TOUR” trong CSDL

### b. Trang thêm điểm du lịch (2.5 điểm)

Tên thành phố:

Mã điểm du lịch:

Tên điểm du lịch:

Đặc trưng:

Khi nhấn nút “Thêm” chương trình lấy dữ liệu từ các control và thêm dữ liệu vào bảng “DIEMDL”, tên thành phố Load từ CSDL



87

---

---

---

---

---

---

---

---

---

---

Bài tập

c. Trang liệt kê điểm du lịch (2.5 điểm)

STT	Mã điểm du lịch	Tên điểm du lịch	Tên thành phố	Đặc trưng	Chức năng
1	DDL01	Núi bà	Đà Nẵng	Leo núi	View Delete
2	DDL02	Chợ Hàn	Đà Nẵng	Mua sắm	View Delete

Liệt kê các điểm du lịch vào một bảng như trên. Dùng kỹ thuật lập trình Ajax để xử lý thao tác nhấn nút "Delete". Khi người dùng nhấn nút "Delete" thì chương trình xóa điểm du lịch tại dòng nhấn nút trên trang web đồng thời xóa điểm du lịch trong bảng "CHUỖI". Khi người dùng nhấn View thì chương trình hiển thị chi tiết điểm du lịch như hình ở câu d.

d. Thông tin chi tiết hợp đồng (2.0 điểm)

Mã điểm du lịch:

Tên điểm du lịch:

Tên thành phố:

Đặc trưng:

Khi người dùng nhấn nút "Update" thì chương trình cập nhật lại các thông tin mới lấy từ những control cho điểm du lịch hiện tại (chú ý: Tên thành phố là một combobox có dữ liệu lấy từ CSDL và mã điểm du lịch không được thay đổi)

e. Dùng kỹ thuật lập trình Ajax để xử lý thao tác khi người dùng nhập số vào textfield và nhấn phím tab thì chương trình liệt kê tên tour cùng với số điểm du lịch của những tour có số điểm du lịch đi qua lớn hơn hoặc bằng giá trị lấy từ textfield (1.5 điểm)

Số điểm du lịch đi qua:

Số điểm du lịch mà Các tour đi qua

STT Tên tour Số điểm du lịch

1 Cẩm trãi 3

2 Du lịch bãi 4

ứng dụng Web

88

88

Q & A



Cảm ơn đã theo dõi

Hy vọng cùng nhau đi đến thành công.

Phát triển ứng dụng Web

89

89