HỌC VIỆN NGÂN HÀNG
KHOA HỆ THỐNG THÔNG TIN QUẢN LÝ

LẬP TRÌNH WEB

CHƯƠNG 4:
NGÔN NGỮ LẬP TRÌNH PHP & HỆ QUẢN TRỊ CSDL MYSQL

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.1. Giới thiệu và cài đặt

5.1.1. Giới thiệu ngôn ngữ lập trình PHP

- PHP được phát triển từ một sản phẩm có tên là PHP/FI. PHP/FI do Rasmus Lerdorf tạo ra năm 1995, ban đầu được xem như là một tập con đơn giản của các mã kịch bản Perl để theo dõi tình hình truy cập đến bản sơ yếu lý lịch của ông trên mạng;
- PHP/FI, viết tắt từ "Personal Home Page/Forms Interpreter", bao gồm một số các chức năng cơ bản cho PHP như ta đã biết đến chúng ngày nay. Nó có các biến kiểu như Perl, thông dịch tự động các biến của form và cú pháp HTML nhúng. Cú pháp này giống như của Perl, mặc dù hạn chế hơn nhiều, đơn giản và có phần thiếu nhất quán.

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.1. Giới thiệu và cài đặt

5.1.1. Giới thiệu ngôn ngữ lập trình PHP

- Vào năm 1997, PHP/FI 2.0, lần viết lại thứ hai của phiên bản C, đã thu hút được hàng ngàn người sử dụng trên toàn thế giới với xấp xỉ 50.000 tên miền đã được ghi nhận là có cài đặt nó, chiếm khoảng 1% số tên miền có trên mạng Internet. Tuy đã có tới hàng nghìn người tham gia đóng góp vào việc tu chỉnh mã nguồn của dự án này thì vào thời đó nó vẫn chủ yếu chỉ là dự án của một người.
- PHP/FI 2.0 được chính thức công bố vào tháng 11 năm 1997, sau một thời gian khá dài chỉ được công bố dưới dạng các bản beta. Nhưng không lâu sau đó, nó đã được thay thế bởi các bản alpha đầu tiên của PHP 3.0.

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.1. Giới thiệu và cài đặt

5.1.1. Giới thiệu ngôn ngữ lập trình PHP

- PHP 3.0 là phiên bản đầu tiên cho chúng ta thấy một hình ảnh gần gũi với các phiên bản PHP mà chúng ta được biết ngày nay. Nó đã được Andi Gutmans và Zeev Suraski tạo ra năm 1997 sau khi viết lại hoàn toàn bộ mã nguồn trước đó. Lý do chính mà họ đã tạo ra phiên bản này là do họ nhận thấy PHP/FI 2.0 hết sức yếu kém trong việc phát triển các ứng dụng thương mại điện tử mà họ đang xúc tiến trong một dự án của trường đại học. Trong một nỗ lực hợp tác và bắt đầu xây dựng dựa trên cơ sở người dùng đã có của PHP/FI, Andi, Rasmus và Zeev đã quyết định hợp tác và công bố PHP 3.0 như là phiên bản thế hệ kế tiếp của PHP/FI 2.0, và chấm dứt phát triển PHP/FI 2.0.

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.1. Giới thiệu và cài đặt

5.1.1. Giới thiệu ngôn ngữ lập trình PHP

- Ngôn ngữ hoàn toàn mới đã được công bố dưới một cái tên mới, xóa bỏ mối liên hệ với việc sử dụng vào mục đích cá nhân hạn hẹp mà cái tên PHP/FI 2.0 gợi nhắc. Nó đã được đặt tên ngắn gọn là 'PHP', một kiểu viết tắt hồi quy của "PHP: Hypertext Preprocessor".
- Vào cuối năm 1998, PHP đã phát triển được con số cài đặt lên tới hàng chục ngàn người sử dụng và hàng chục ngàn Web site báo cáo là đã cài nó. Vào thời kì đỉnh cao, PHP 3.0 đã được cài đặt cho xấp xỉ 10% số máy chủ Web có trên mạng Internet.
- PHP 3.0 đã chính thức được công bố vào tháng 6 năm 1998, sau thời gian 9 tháng được cộng đồng kiểm nghiệm.

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.1. Giới thiệu và cài đặt

- 5.1.1. Giới thiệu ngôn ngữ lập trình PHP
  - Vào mùa đông năm 1998, ngay sau khi PHP 3.0 chính thức được công bố, Andi Gutmans và Zeev Suraski đã bắt đầu bắt tay vào việc viết lại phần lõi của PHP. Mục đích thiết kế là nhằm cải tiến tốc độ xử lý các ứng dụng phức tạp, và cải tiến tính mô đun của cơ sở mã PHP. Những ứng dụng như vậy đã chạy được trên PHP 3.0 dựa trên các tính năng mới và sự hỗ trợ khá nhiều các cơ sở dữ liệu và API của bên thứ ba, nhưng PHP 3.0 đã không được thiết kế để xử lý các ứng dụng phức tạp như thế này một cách có hiệu quả.
  - Với PHP 4, số nhà phát triển dùng PHP đã lên đến hàng trăm nghìn và hàng triệu site đã công bố cài đặt PHP, chiếm khoảng 20% số tên miền trên mạng Internet.
  - Nhóm phát triển PHP cũng đã lên tới con số hàng nghìn người và nhiều nghìn người khác tham gia vào các dự án có liên quan đến PHP như PEAR, PECL và tài liệu kĩ thuật cho PHP.

5.1. Giới thiệu và cài đặt

5.1.1. Giới thiệu ngôn ngữ lập trình PHP

- Sự thành công hết sức to lớn của PHP 4.0 đã không làm cho nhóm phát triển PHP tự mãn. Cộng đồng PHP đã nhanh chóng giúp họ nhận ra những yếu kém của PHP 4 đặc biệt với khả năng hỗ trợ lập trình hướng đối tượng (OOP), xử lý XML, không hỗ trợ giao thức máy khách mới của MySQL 4.1 và 5.0, hỗ trợ dịch vụ web yếu. Những điểm này chính là mục đích để Zeev và Andi viết Zend Engine 2.0, lõi của PHP 5.0. Một thảo luận trên Slashdot đã cho thấy việc phát triển PHP 5.0 có thể đã bắt đầu vào thời điểm tháng 12 năm2002 nhưng những bài phỏng vấn Zeev liên quan đến phiên bản này thì đã có mặt trên mạng Internet vào khoảng tháng 7 năm 2002. Ngày 29 tháng 6 năm 2003, PHP 5 Beta 1 đã chính thức được công bố để cộng đồng kiểm nghiệm. Đó cũng là phiên bản đầu tiên của Zend Engine 2.0. Phiên bản Beta 2 sau đó đã ra mắt vào tháng 10 năm 2003 với sự xuất hiện của hai tính năng rất được chờ đợi: Iterators, Reflection nhưng namespaces một tính năng gây tranh cãi khác đã bị loại khỏi mã nguồn. Ngày 21 tháng 12 năm 2003: PHP 5 Beta 3 đã được công bố để kiểm tra với việc phân phối kèm với Tidy, bỏ hỗ trợ Windows 95, khả năng gọi các hàm PHP bên trong XSLT, sửa chữa nhiều lỗi và thêm khá nhiều hàm mới. PHP 5 bản chính thức đã ra mắt ngày 13 tháng 7 năm 2004 sau một chuỗi khá dài các bản kiểm tra thử bao gồm Beta 4, RC 1, RC2, RC3. Mặc dù coi đây là phiên bản sản xuất đầu tiên nhưng PHP 5.0 vẫn còn một số lỗi trong đó đáng kể là lỗi xác thực HTTP.

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.1. Giới thiệu và cài đặt

- 5.1.1. Giới thiệu ngôn ngữ lập trình PHP
  - Ngày 14 tháng 7 năm 2005, PHP 5.1 Beta 3 được PHP Team công bố đánh dấu sự chín muồi mới của PHP với sự có mặt của PDO, một nỗ lực trong việc tạo ra một hệ thống API nhất quán trong việc truy cập cơ sở dữ liệu và thực hiện các câu truy vấn. Ngoài ra, trong PHP 5.1, các nhà phát triển PHP tiếp tục có những cải tiến trong nhân Zend Engine 2, nâng cấp mô đun PCRE lên bản PCRE 5.0 cùng những tính năng và cải tiến mới trong SOAP, streams và SPL.
  - Hiện nay phiên bản tiếp theo của PHP đang được phát triển, PHP 6 bản sử dụng thử đã có thể được download tại địa chỉ http://snaps.php.net. Phiên bản PHP 6 được kỳ vọng sẽ lấp đầy những khiếm khuyết của PHP ở phiên bản hiện tại, ví dụ: hỗ trợ namespace (hiện tại các nhà phát triển vẫn chưa công bố rõ ràng về vấn đề này); hỗ trợ Unicode; sử dụng PDO làm API chuẩn cho việc truy cập cơ sở dữ liệu, các API cũ sẽ bị đưa ra thành thư viện PECL...

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.1. Giới thiệu và cài đặt

- 5.1.1. Giới thiệu ngôn ngữ lập trình PHP
- PHP với doanh nghiệp
  - Rất nhiều nhà phát triển ứng dụng và quản lý dự án có quan điểm rằng PHP vẫn chưa sẵn sàng cho cấp doanh nghiệp (enterprise) và trên thực tế, PHP vẫn chưa xâm nhập sâu được vào thị trường này. Chính vì thế, Zend đã tiến hành nhiều biện pháp nhằm chuẩn hóa PHP, tạo được sự tin cậy hơn cho giới người dùng cao cấp.
    - Zend Platform là một bộ sản phẩm giúp quản lý hệ thống ứng dụng PHP, nâng cao hiệu suất, tăng tốc độ của ứng dụng PHP.
    - Zend Framework là một tập hợp các lớp, các thư viện lập trình viết bằng PHP (PHP 5) nhằm cung cấp một giao diện lập trình chuẩn cho các nhà phát triển ứng dụng.
  - Ngoài ra, một số framework khác cũng đã được phát triển nhằm hỗ trợ lập trình PHP ở cấp doanh nghiệp, trong đó đáng chú ý có thể kể đến là CodeIgniter, CakePHP, Symfony,Seagull...

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.1. Giới thiệu và cài đặt

- 5.1.1. Giới thiệu ngôn ngữ lập trình PHP
- Cú pháp

```
<html>
 <head>
  <title>PHP Test</title>
 </head>
 <body>
   <?php echo '<p>Hello World</p>'; ?>
 </body>
</html>
```

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.1. Giới thiệu và cài đặt

- 5.1.1. Giới thiệu ngôn ngữ lập trình PHP
- Cú pháp cơ bản
  - Được giới hạn bởi cặp kí hiệu <?php …. ?> hoặc <? …. ?>
  - Các biến được bắt đầu bằng dấu $ và không cần khai báo trước kiểu dữ liệu
  - Mỗi câu lệnh được kết thúc bởi dấu ;
  - Chú thích được đặt trong cặp kí hiệu /* …. */ với đoạn chú thích hoặc // và # với dòng chú thích.
  - Về cú pháp các từ khóa và ngôn ngữ, PHP tương tự hầu hết các ngôn ngữ lập trình bậc cao có cú pháp kiểu C, C++, Java, Perl …

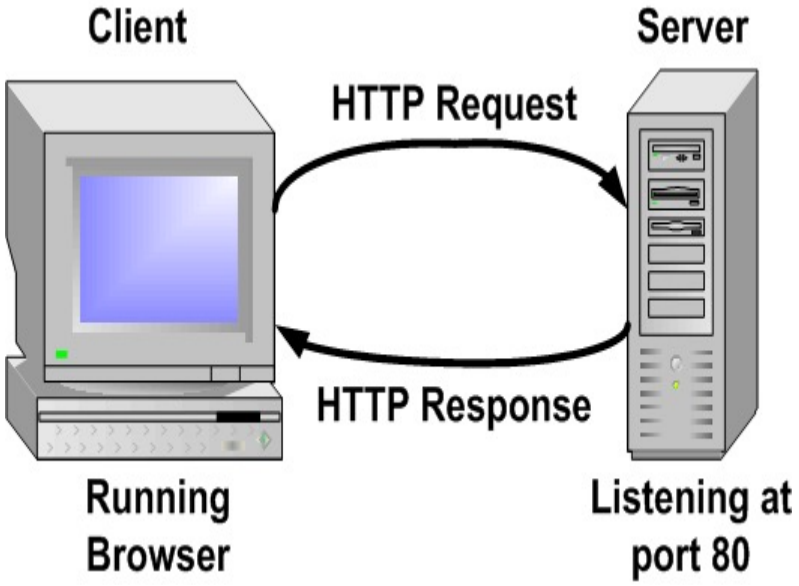
MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.1. Giới thiệu và cài đặt

- 5.1.1. Giới thiệu ngôn ngữ lập trình PHP
- Các lệnh in ra màn hình
  - echo // In ra màn hình chuỗi ký tự hoặc nội dung biến
  - print_r(biến); // In ra màn hình chuỗi ký tự hoặc nội dung biến
  - var_dump(biến); // In ra màn hình kiểu dữ liệu của biến, kích thước biến và nội dung của biến

5.1. Giới thiệu và cài đặt

- 5.1.1. Giới thiệu ngôn ngữ lập trình PHP
- Cơ chế hoạt động



MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.1. Giới thiệu và cài đặt

- 5.1.2. Cài đặt XAMPP server trên localhost
- Download bộ cài XAMPP server
- Cài đặt vào máy tính
- Cấu hình các thông số phù hợp
- Hướng dẫn cài đặt XAMPP tại đây

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.1. Kiểu dữ liệu, biến số, hằng số trong ngôn ngữ PHP
  - a. Kiểu dữ liệu
    - PHP có ba kiểu dữ liệu cơ bản: interger, double và string
      - Integer là kiểu số nguyên chiếm 4byte bộ nhớ, giá trị nằm trong khoảng -2 tỷ đến +2 tỷ
      - Double là kiểu số thực, phạm vi biểu diễn nằm trong khoảng (-10^-308, +10^308)
      - Kiểu dữ liệu string bao gồm các kí tự, chuỗi kí tự và con số.
  - Các kiểu dữ liệu khác: array, object và null
    - Array: Dữ liệu kiểu mảng
    - Object: Dữ liệu kiểu đối tượng
    - Null: Dữ liệu rỗng

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.1. Kiểu dữ liệu, biến số, hằng số trong ngôn ngữ PHP
  - b. Hằng số
    - Hằng số là những giá trị không đổi trong suốt chương trình
    - Khai báo hằng số:
      - define("Tên hằng số", "Giá trị");
    - Kiểm tra một hằng số đã được khai báo chưa
      - defined("Tên hằng số");
    - Tra cứu các hằng số mặc định trong PHP
      - phpinfo();

5.2. Các câu lệnh cơ bản

- 5.2.1. Kiểu dữ liệu, biến số, hằng số trong ngôn ngữ PHP
  - b. Biến số
    - Trong PHP, không cần khai báo kiểu dữ liệu khi khai báo biến.
    - Mọi biến số trong PHP đều bắt đầu bằng ký hiệu $
    - PHP không có khái niệmTRUE và FALSE mà hiểu TRUE là giá trị 1 và FALSE là giá trị 0.
    - Ví dụ
      - $a = 1; // a là biến có kiểu dữ liệu int
      - $b = 1.5; // b là biến có kiểu dữ liệu double
      - $c = “banking academy”; // c là biến có kiểu dữ liệu string
    - Có thể ép kiểu trong PHP theo cách sau
      - $a = 15,5; // Biến a đang có kiểu dữ liệu double
      - $b = (int)$a; // Biến b có kiểu dữ liệu int, giá trị = 15
      - $c = (string) $b; // Biến c có kiểu dữ liệu string, giá trị = “15”

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.1. Kiểu dữ liệu, biến số, hằng số trong ngôn ngữ PHP
  - b. Một số hàm làm việc với biến trong PHP
    - Hàm gettype();
      - Trả về kiểu dữ liệu của một biến nào đó
      - Ví dụ:
        - » $a = 5;
        - » echo gettype($a); // => Kết quả trả về: “interger”;
    - Hàm settype();
      - Ép kiểu cho một biến nào đó, nếu thành công trả về 1, ngược lại trả về 0
      - Ví dụ
        - » $a = 5.1; // Biến a có kiểu dữ liệu “double”
        - » settype($a, “integer”);
        - » echo gettype($a); // => Kết quả trả về lúc này là “integer”;

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.1. Kiểu dữ liệu, biến số, hằng số trong ngôn ngữ PHP
  - b. Một số hàm làm việc với biến trong PHP
    - Hàm isset(biến)
      - Kiểm tra xem một biến đã được gán giá trị chưa, nếu có trả về 1, ngược lại trả về 0
      - Ví dụ:

```
» $a = 1;
» if(isset($a))
»         echo “Biến đã được gán”;
»    else
»         echo “Biến chưa được gán”;
```

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.1. Kiểu dữ liệu, biến số, hằng số trong ngôn ngữ PHP
  - b. Một số hàm làm việc với biến trong PHP
    - Hàm unset(biến)
      - Giải phóng bộ nhớ đã cấp phát cho biến
      - Ví dụ:

```
» $a = 1;
» if(isset($a))
»         echo “Biến đã được gán”;
»   else
»         echo “Biến chưa được gán”;
»   unset($a);
»   if(!isset($a))
»         echo “Bộ nhớ đã được giải phóng”;
```

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.1. Kiểu dữ liệu, biến số, hằng số trong ngôn ngữ PHP
  - b. Một số hàm làm việc với biến trong PHP
    - Hàm empty(biến)
      - Kiểm tra xem một biến có phải là rỗng hay không, nếu có trả về 1, ngược lại trả về 0. Đối với biến kiểu số, giá trị bằng 0 được coi là rỗng. Đối với biến kiểu chuỗi, một xâu rỗng được coi là rỗng.
      - Ví dụ
        - » $a;
          - » empty($a) => trả về giá trị 0 (false);
        - » $a = “abc”;
          - » empty($a) => trả về giá trị 1 (true);
        - » $a = “”;
          - » empty($a) => trả về giá trị 0 (false);
      - PHÂN BIỆT empty() và isset() ??????????

5.2. Các câu lệnh cơ bản

5.2.2. Toán tử trong PHP

a. Các phép toán số học

Phép toán	ý nghĩa	Ví dụ	Giải thích
+	Phép cộng	7 + 2	Thực hiện phép cộng giữa 7 và 2 : 9
-	Phép trừ	7 – 2	Thực hiện phép trừ giữa 7 và 2 : 5
*	Phép nhân	7 * 2	Thực hiện phép nhân giữa 7 và 2 : 14
/	Phép chia	7 / 2	Thực hiện phép chia giữa 7 và 2 : 3.5
%	Chia d	7 % 2	Thực hiện phép chia d giữa 7 và 2 : 1

5.2. Các câu lệnh cơ bản

- 5.2.2. Toán tử trong PHP
  - b. Các phép toán khi viết ngắn gọn

Khi viết	Tương đương với
$h += $i	$h = $h + $i
$h -= $i	$h = $h - $i
$h *= $i	$h = $h * $i
$h /= $i	$h = $h / $i
$h %= $i	$h = $h % $i

5.2. Các câu lệnh cơ bản

5.2.2. Toán tử trong PHP

c. Các phép toán so sánh

Phép toán	ý nghĩa	Ví dụ	Giải thich
==	So sánh bằng	$h == $i	Kiểm tra $h và $i có bằng nhau không
<	So sánh nhỏ hơn	$h < $i	Kiểm tra $h có nhỏ hơn $i không
>	So sánh lớn hơn	$h > $i	Kiểm tra $h có lớn hơn $i không
<=	Nhỏ hơn hoặc bằng	$h <= $i	Kiểm tra $h có nhỏ hơn hoặc bằng $i
>=	Lớn hơn hoặc bằng	$h => $i	Kiểm tra $h có lớn hơn hoặc bằng $i
!=	So sánh khác	$h != $i	Kiểm tra $h có khác $i không
<>	So sánh khác	$h <> $i	Kiểm tra $h có khác $i không

5.2. Các câu lệnh cơ bản

- 5.2.2. Toán tử trong PHP
  - d. Các phép toán logic

Toán hạng a	Toán hạng b	a && b	a || b	!a	!b
1	1	1	1	0	0
1	0	0	1	0	1
0	1	0	1	1	0
0	0	0	0	1	1

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.2. Toán tử trong PHP
  - e. Các phép toán với chuỗi
    - Toán tử ghép chuỗi: dấu chấm "."
      - Ví dụ:
        - » $a = "hệ thống thông tin";
        - » $b = "quản lý";
        - » echo $a." ".$b; // => Kết quả "Hệ thống thông tin quản lý"
    - Có thể chèn biến vào trong một chuỗi bằng cách sử dụng cặp ngoặc nhọn
      - Ví dụ
        - » $a = "học viện";
        - » echo "${a} ngân hang";

5.2. Các câu lệnh cơ bản

5.2.2. Toán tử trong PHP

e. Các phép toán thao tác mức bit

Ký hiệu	ý nghĩa
&	AND bit
|	OR bit
^	XOR bit

&	Kết quả	|	Kết quả	^	Kết quả
1&1	1	1|1	1	1^1	0
1&0	0	1|0	1	1^0	1
0&1	0	0|1	1	0^1	1
0&0	0	0|0	0	0^0	0

5.2. Các câu lệnh cơ bản

- 5.2.2. Toán tử trong PHP
  - e. Các phép toán tăng giảm
    - Phép tăng: Tăng giá trị của toán hạng lên 1 đơn vị
      - Ví dụ:
        - » $a = 1; $a++;
        - » $b = 1; ++$b;
    - Phép giảm: Giảm giá trị của toán hạng đi 1 đơn vị
      - Ví dụ
        - » $a = 1; $a--;
        - » $b = 1; --$b;

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.2. Toán tử trong PHP
  - e. Các phép toán có điều kiện
    - Cú pháp: (biểu thức logic) ? (giá trị nếu đúng) : (giá trị nếu sai)
    - Ví dụ
      - $a;
      - $b = “Đúng”;
      - $c = “Sai”;
      - $result = $a?$b:$c; //=> Kết quả thế nào nếu gán $a = 1 ?
  - f. Toán tử sizeof(biến)
    - Cho biết kích thước của biến.

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.3. Các cấu trúc điều khiển
  - a. Cấu trúc rẽ nhánh if - else
    - Cú pháp:
      - if(biểu thức logic)
      - {
      - //Khối lệnh thực hiện nếu biểu thức đúng
      - }
      - else
      - {
      - // Khối lệnh thực hiện nếu biểu thức sai
      - }
  - Cấu trúc rẽ nhánh lồng nhau ?
  - Ý nghĩa của “else” ?

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.3. Các cấu trúc điều khiển
  - b. Cấu trúc rẽ nhánh switch - case
    - Cú pháp:

```
switch ( biểu thức )
{
    case gt1:
            câu lệnh 1;
            break;
    case gt2:
            câu lệnh 2;
            break;
        ……..
        default:
            câu lệnh n;
}
```

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.3. Các cấu trúc điều khiển
  - c. Vòng lặp for
    - Cú pháp:
      - for(biểu thức 1; biểu thức 2 (logic); biểu thức 3)
      - {
      - //Khối lệnh
      - }
    - Có thể vắng mặc bất kỳ biểu thức nào, tuy nhiên dấu ; vẫn phải được viết

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.3. Các cấu trúc điều khiển
  - d. Vòng lặp while()
    - Cú pháp:
      - while(biểu thức)
      - {
      - //Khối lệnh
      - }
    - Tránh vòng lặp vô hạn

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.3. Các cấu trúc điều khiển
  - e. Vòng lặp do while()
    - Cú pháp:
      - do
      - {
      - //Khối lệnh
      - }
      - while(biểu thức)
    - Tránh vòng lặp vô hạn
  - f. Các lệnh làm việc với vòng lặp:
  - break; // thoát khỏi vòng lặp gần nhất
  - continue; // Bắt đầu lại vòng lặp hiện thời

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.4. Hàm trong PHP
  - Khai báo hàm
    - Cú pháp
      - function <tên hàm>(các tham số)
      - {
      - // Khối lệnh xử lý trong hàm
      - }
    - Lưu ý:
      - Không cần khai báo kiểu dữ liệu trả về
      - Lưu ý cách đặt tên hàm
      - Hàm có thể có giá trị trả về (bằng lệnh return <giá trị>;) hoặc không
      - Các lệnh được gọi bất kỳ hàm nào đã được khai báo và định nghĩa.

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.4. Hàm trong PHP
  - b. Biến toàn cục và biến cục bộ

5.2. Các câu lệnh cơ bản

- 5.2.5. Mảng trong PHP
  - Mảng một chiều
    - Khai báo
      - $mang = array("gt1", "gt2", ...);
      - => $mang[0] = "gt1", $mang[1] = "gt2" .....
    - Truy xuất vào giá trị mảng:
      - Một phần tử của mảng có 2 thành phần: chỉ số (index hoặc key) và giá trị (value)
        - » $mang[<chỉ số>] = giá trị
      - Chỉ số của mảng có thể là số hoặc chuỗi, các giá trị của mảng không cần phải cùng kiểu giá trị
        - » Ví dụ:
          - » $mang[1] = 100;
          - » $mang["hoten"] = "Nguyễn Văn Nam";
          - » $mang["tuoi"] = 19;
          - » $mang[] = "Banking"; // PHP sẽ tự động gán chỉ số là số tăng dần

5.2. Các câu lệnh cơ bản

- 5.2.5. Mảng trong PHP
  - Mảng nhiều chiều
    - Khai báo

```
$danhSachLop =  array("htttak14" => array(
                                        array("hoten" => "Nguyen Van A",
                                              "msv" => "14A4040005"),
                                        array("hoten" => "Nguyen Thi B",
                                              "msv" => "14A4040006")
                                        ),
                    "htttbk14" => array(
                                        array("hoten" => "Nguyen Van C",
                                              "msv" => "14A4040007"),
                                        array("hoten" => "Nguyen Thi D",
                                              "msv" => "14A4040008")
                                        )
                    )
```

5.2. Các câu lệnh cơ bản

- 5.2.5. Mảng trong PHP
  - Một số hàm xử lý trong mảng:
    - Sắp xếp mảng:
      - sort(): Sắp xếp tăng dần
      - rsort(): Sắp xếp giảm dần
      - asort(); arsort(): Sử dụng với mảng có giá trị là kiểu chuỗi

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.5. Mảng trong PHP
  - Một số hàm xử lý trong mảng:
    - in_array("giá trị", tên mảng): Kiểm tra giá trị có tồn tại trong mảng không, trả về 1 nếu có và 0 nếu không có.
    - array_unique(tên mảng): Loại bỏ các giá trị trùng lặp trong mảng
    - array_merge(mảng 1, mảng 2): Ghép mảng 2 vào sau mảng 1

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.5. Mảng trong PHP
  - Xử lý mảng qua từng giá trị
    - Vòng lặp foreach();
    - Cú pháp
      - $mang = array("hoten" => "Nam", "tuoi" => 18, "quequan" => "Hà Nội");
      - foreach($mang as $key => $value)
      - {
      - print_r($key);
      - printr_r($value);
      - }
      - $key: Các chỉ số của lần lượt từ phần tử đầu đến phần tử cuối
      - $value: Các giá trị của màng tương ứng lần lượt từ phần tử đầu đến phần tử cuối
      - Có thể không cần $key trong vòng lặp

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.5. Lớp và đối tượng trong PHP
  - Khai báo lớp
    - Cú pháp
      - class <tên lớp>
      - {
      - var attribute1;
      - var attribute 2; // Định nghĩa các thuộc tính trong lớp
      - function <tên lớp>([params])
      - {
      - //Phương thức khởi tạo
      - }
      - function method1(params …) // Định nghĩa các phương thức trong lớp
      - {
      - //Khối lệnh xử lý trong method1
      - }
      - }

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.5. Lớp và đối tượng trong PHP
  - Sử dụng một lớp đã có
    - Cú pháp
      - $object = new class([params]); // Khai báo một đối tượng mới
      - $object->method1(params); // Gọi đến phương thức nằm trong đối tượng
  - Lớp kế thừa
    - Cú pháp
      - class <tên lớp kế thừa> extends <tên lớp đã có>
      - {
      - // Khối lệnh
      - }

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.6. PHP Form
  - Các phương thức GET và POST khi chuyển dữ liệu từ HTML Form lên server.
  - Tại file vidu.html
    - <form method="get" action="action.php">
    - <input type="text" name="hoten" />
    - <input type="submit">
    - </form>
  - Tại file action.php
    - $ten = $_GET["hoten"];
    - print_r($a);
    - Phân biệt sự khác nhau giữa phương thức GET và phương thức POST ?

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.6. Các hàm API trong PHP
  - PHP có khả năng làm việc với các hệ quản trị cơ sở dữ liệu như Oracle, PostgreSQL, SQLServer, MySQL, NoSQL ….thông qua chuẩn ODBC (open database connectivity) bằng các hàm API (application programming interfaces)
  - Nội dung môn học giới hạn trong việc sử dụng PHP kết hợp với hệ quản trị CSDL MySQL thông qua các hàm API

5.2. Các câu lệnh cơ bản

- 5.2.6. Các hàm API trong PHP
  - Các hàm kết nối đến MySQL server

    // Kết nối đến máy chủ với username, password và CSDL.
    - mysqli_connect($host, $username, $password, $database);

    // Đóng kết nối với máy chủ
    - mysqli_close(connection);
  - Các hàm thao tác trên cơ sở dữ liệu

    // Truy vấn cơ sở dữ liệu bằng câu lệnh SQL
    - mysqli_query(connection, query);
    - Connection: Kết quả trả về của hàm mysqli_connect();
    - Query: Chuỗi truy vấn cơ sở dữ liệu;
    - Ví dụ
      - $con = mysqli_connect("localhost","root","123456", "k19_httt");
      - mysqli_query($con, "SELECT * FROM tbl_user");

MANAGEMENT INFORMATION SYSTEMS
BANKING ACADEMY OF VIETNAM

5.2. Các câu lệnh cơ bản

- 5.2.6. Các hàm API trong PHP
  - Các hàm kết nối đến MySQL server
    - mysqli_fetch_array(tên mảng) // truy cập vào từng dữ liệu của mảng kết quả sau khi thực hiện select dữ liệu
    - Ví dụ
      - Bảng cơ sở dữ liệu dssv(userId, username);
      - $con = mysqli_connect($host, $user, $pass, $db);
      - $a = mysqli_query($con, “SELECT * FROM dssv”);

      => Lúc này $a là một mảng chứa các bản ghi trong bảng dssv
      - while($row = mysqli_fetch_array($a))
      - {
      - echo $row [“userId”].”
”;
      - echo $row[“username”];
      - }