NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

*Ngày ….. tháng ….. năm ……*

**GIÁO VIÊN HƯỚNG DẪN** (Ký và ghi rõ họ tên)

MỤC LỤC

[Chương 1. Giới thiệu ngôn ngữ PHP và MySql 5](#_Toc382389848)

[I. Giới thiệu PHP 5](#_Toc382389849)

[1. PHP là gì? 5](#_Toc382389850)

[2. Lịch sử hình thành và phát triển của ngôn ngữ PHP. 5](#_Toc382389851)

[a. PHP/FI 5](#_Toc382389852)

[b. PHP 3 5](#_Toc382389853)

[c. PHP4 6](#_Toc382389854)

[d. PHP 5 6](#_Toc382389855)

[e. PHP6 7](#_Toc382389856)

[3. Tại sao phải sử dụng PHP ? 7](#_Toc382389857)

[II. Giới thiệu MySql 9](#_Toc382389858)

[1. Mysql là gì? 9](#_Toc382389859)

[2. Tính năng và lợi thế của MySql 10](#_Toc382389860)

[a. Tính năng. 10](#_Toc382389861)

[b. Lợi thế: 10](#_Toc382389862)

[3. Nhược điểm MySQL: 10](#_Toc382389863)

[Chương 2. Phân tích ngôn ngữ PHP 11](#_Toc382389864)

[I. Ưu điểm của ngôn ngữ PHP 11](#_Toc382389865)

[II. Nhược điểm của ngôn ngữ PHP so với các ngôn ngữ khác: 11](#_Toc382389866)

[III. Cú pháp và thành phần trong PHP 11](#_Toc382389867)

[1. Cú pháp 11](#_Toc382389868)

[2. Xuất dữ liệu ra trình duyệt: 12](#_Toc382389869)

[3. Các kiểu dữ liệu 12](#_Toc382389870)

[4. Hằng số: 13](#_Toc382389871)

[a. Khai báo hằng số: 13](#_Toc382389872)

[b. Các hằng số đã được định nghĩa trong PHP (Built\_in Constants): 14](#_Toc382389873)

[5. Toán tử 15](#_Toc382389874)

[6. Thành phần 18](#_Toc382389875)

[a. Biến, chuỗi, mảng 18](#_Toc382389876)

[i. Biến, chuỗi: 18](#_Toc382389877)

[ii. Chuỗi 20](#_Toc382389878)

[iii. Mảng 21](#_Toc382389879)

[b. Vòng lặp 24](#_Toc382389880)

[i. Mệnh đề điều kiện if-else: 24](#_Toc382389881)

[ii. Câu lệnh switch: 25](#_Toc382389882)

[iii. Vòng lặp For: 27](#_Toc382389883)

[iv. Vòng lặp while: 29](#_Toc382389884)

[v. Vòng lặp Do..while: 30](#_Toc382389885)

[vi. Lệnh Break 31](#_Toc382389886)

[vii. Lệnh continue: 31](#_Toc382389887)

[c. Hàm 31](#_Toc382389888)

[i. Cách xây dựng hàm 31](#_Toc382389889)

[ii. Gọi hàm 31](#_Toc382389890)

[iii. Biến toàn cục và biến cục bộ 31](#_Toc382389891)

[d. Post, get, cookie, session 32](#_Toc382389892)

[i. Post 32](#_Toc382389893)

[ii. Get 33](#_Toc382389894)

[iii. Cookie 33](#_Toc382389895)

[iv. Session 34](#_Toc382389896)

[7. Bảo mật 34](#_Toc382389897)

[8. Khai báo tiền xử lí include và require: 34](#_Toc382389898)

[IV. Các hàm cơ bản làm việc với MYSQL: 34](#_Toc382389899)

[1. Các hàm kết nối đến MYSQL SERVER: 34](#_Toc382389900)

[2. Các hàm thao tác trên CSDL: 35](#_Toc382389901)

[3. Các hàm thao tác trên dữ liệu: 35](#_Toc382389902)

[Chương 3. Dome website quản lí karaoke Nnice (Quản lí dịch vụ phát sinh) 36](#_Toc382389903)

[I. Phân tích yêu cầu nghiệp vụ (yêu cầu đề bài): 36](#_Toc382389904)

[II. Sơ đồ cơ sở dữ liệu: 36](#_Toc382389905)

[III. Sơ đồ Usercase 37](#_Toc382389906)

[IV. Bảng và kiểu dữ liệu 38](#_Toc382389907)

[V. Dome: 41](#_Toc382389908)

[1. Sơ đồ cây của đồ án: 41](#_Toc382389909)

[2. Code: 42](#_Toc382389910)

[a. Header.php 42](#_Toc382389911)

[b. Function.php 43](#_Toc382389912)

[c. Footer.php 43](#_Toc382389913)

[d. Sibar-3.php 44](#_Toc382389914)

[e. Mysqli\_connect.php 44](#_Toc382389915)

[f. Quang-cao.php 44](#_Toc382389916)

[g. Sibar-2.php 44](#_Toc382389917)

[h. Index.php 44](#_Toc382389918)

[i. Style.css 45](#_Toc382389919)

[j. Admin.php 45](#_Toc382389920)

[k. Phong\_hoat\_dong.php 50](#_Toc382389921)

[l. Phong\_trong.php 55](#_Toc382389922)

[m. Suacthd.php 59](#_Toc382389923)

[n. Suadv.php 61](#_Toc382389924)

[o. Suaphong.php 63](#_Toc382389925)

[p. Them\_dv.php 64](#_Toc382389926)

[q. Them\_phong.php 66](#_Toc382389927)

[r. Xoacthd.php 69](#_Toc382389928)

[s. Xoadv.php 69](#_Toc382389929)

[t. Xoaphong.php 70](#_Toc382389930)

[3. Dome bài làm: 70](#_Toc382389931)

**Lời nói đầu:**

Với sự phát triển mạnh mẽ của công nghệ thông tin và sự lớn mạnh của mạng máy tính toàn cầu. Việc sử dụng các ứng dụng công nghệ thông tin vào công việc và đời sống không còn xa lại với mọi người, giúp con người ngày càng xích lại gần nhau hơn, giảm được khoảng cách về không gian địa lí, thời gian, ngôn ngữ, giúp cho mọi người hiểu rõ văn hóa, xã hội của nhau hơn. Thông tin mạng giúp cho con người có thể tiếp xúc với các tri thức mới nhanh hơn. Ngày càng có nhiều ứng dụng công nghệ thông tin được áp dụng vào đời sống. Nó giúp cho việc giảm nguồn lao động, giúp quản lí trở nên đơn giản và giảm sự phức tạp của quản lí. Những ứng dụng quản lí thường gặp là: quản lí điểm, quản lí sinh viên (học sinh), quản lí nhân viên, quản lí bán hàng. Trước đây mạng Internet chưa phát triển mạnh mẽ những ứng dụng xây dựng và xử dụng trên desktop. Bây giờ mạng Internet đã phát triển mạnh mẽ nên các ứng dụng đươc phát triển để xử dụng trên web.

Trong đề tài “quản lí dịch vụ phát sinh karaoke Nnice” em xử dụng ngôn ngữ php để làm đồ án.

**LỜI CẢM ƠN**

Em xin gởi lời cảm ơn chân thành và sâu sắc đối với Nguyễn Văn Thắng đã tạo điều kiện cho em làm đồ án.Và em cũng xin chân thành cám ơn thầy đã nhiệt tình hướng dẫn hướng dẫn em hoàn thành tốt khóa thực tập này.

Trong quá trình thực tập, cũng như là trong quá trình làm bài báo cáo, khó tránh khỏi sai sót, rất mong các quý công ty bỏ qua.

Đồng thời do trình độ lý luận cũng như kinh nghiệm thực tiễn còn hạn chế nên bài báo cáo không thể tránh khỏi những thiếu sót, em rất mong nhận được ý kiến đóng góp Thầy để em học thêm được nhiều kinh nghiệm và sẽ hoàn thành tốt khóa học.

Em xin chân thành cảm ơn !

1. Giới thiệu ngôn ngữ PHP và MySql
   1. Giới thiệu PHP
      1. PHP là gì?

PHP (Hypertext Preprocessor) là ngôn ngữ lập trình kịch bản hay một loại mã lệnh chủ yếu được dùng để phát triển các ứng dụng viết cho máy chủ, mã nguồn mở, dùng cho mục đích tổng quát. Nó được tích hợp với web và có thể dể dàng nhúng vào Html. Do được tối ưu hóa cho các ứng dụng web, tốc độ nhanh, nhỏ gọn, cú pháp giống C và java, dễ học và thời gian xây dựng sản phẩm tương đối ngắn so với các ngôn ngữ khác nên PHP nhanh chóng trở thành ngôn ngữ lập trình web phổ biến trên thế giới.

* + 1. Lịch sử hình thành và phát triển của ngôn ngữ PHP.
       1. PHP/FI

PHP được phát triển từ một sản phẩm có tên là [PHP/FI](http://vi.wikipedia.org/w/index.php?title=PHP/FI&action=edit&redlink=1). 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. Ông đã đặt tên cho bộ mã kịch bản này là 'Personal Home Page Tools'. Khi cần đến các chức năng rộng hơn, Rasmus đã viết ra một bộ thực thi bằng C lớn hơn để có thể truy vấn tới các cơ sở dữ liệu và giúp cho người sử dụng phát triển các ứng dụng web đơn giản. Rasmus đã quyết định công bố mã nguồn của PHP/FI cho mọi người xem, sử dụng cũng như sửa các lỗi có trong nó đồng thời cải tiến mã nguồn.

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.

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.

* + - 1. PHP 3

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.

Một trong những sức mạnh lớn nhất của PHP 3.0 là các tính năng mở rộng mạnh mẽ của nó. Ngoài khả năng cung cấp cho người dùng cuối một cơ sở hạ tầng chặt chẽ dùng cho nhiều cơ sở dữ liệu, giao thức và API khác nhau, các tính năng mở rộng của PHP 3.0 đã thu hút rất nhiều nhà phát triển tham gia và đề xuất các mô đun mở rộng mới. Hoàn toàn có thể kết luận được rằng đây chính là điểm mấu chốt dẫn đến thành công vang dội của PHP 3.0. Các tính năng khác được giới thiệu trong PHP 3.0 gồm có hỗ trợ cú pháp hướng đối tượng và nhiều cú pháp ngôn ngữ nhất quán khác.

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.

* + - 1. PHP4

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.

Một động cơ mới, có tên ' Zend Engine' (ghép từ các chữ đầu trong tên của Zeev và Andi), đã đáp ứng được các nhu cầu thiết kế này một cách thành công, và lần đầu tiên được giới thiệu vào giữa năm 1999. PHP 4.0, dựa trên động cơ này, và đi kèm với hàng loạt các tính năng mới bổ sung, đã chính thức được công bố vào tháng 5 năm 2000, gần 2 năm sau khi bản PHP 3.0 ra đời. Ngoài tốc độ xử lý được cải thiện rất nhiều, PHP 4.0 đem đến các tính năng chủ yếu khác gồm có sự hỗ trợ nhiều máy chủ Web hơn, hỗ trợ phiên làm việc HTTP, tạo bộ đệm thông tin đầu ra, nhiều cách xử lý thông tin người sử dụng nhập vào bảo mật hơn và cung cấp một vài các cấu trúc ngôn ngữ mới.

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.

* + - 1. PHP 5

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 năm 2002 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:Iterator, 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 Windown 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.

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.

* + - 1. PHP6

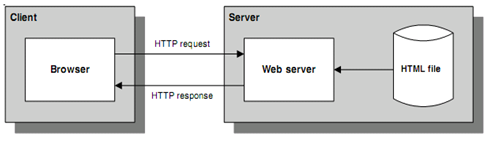
Hiện nay phiên bản tiếp theo của PHP đang được phát triển, PHP 6 bản sử dùng thử đã có thể được download tại địa chỉ [http://snaps.php.net](http://snaps.php.net/). Phiên bản PHP 6 được kỳ vọng sẽ lấp đầy những khiếm khuyết của PHP ở phiên bản hiện tại, ví dụ: hỗ trợ namespace (hiện tại các nhà phát triển 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...

* + 1. Tại sao phải sử dụng PHP ?

Như chúng ta đã biết, có rất nhiều trang web được xây dựng bởi ngôn ngữ HTML (HyperText Markup Language). Đó làm những trang web tĩnh, nghĩa là chúng chỉ chứa một nội dung cụ thể với dòng văn bản đơn thuần, hình ảnh, và có sự hỗ trợ của ngôn ngữ JavaScript, hoặc Java Apple. Những trân web vậy người ta gọi client-side. Tuy nhiên, Internet và Intranets đã được sử dụng cho các ứng dụng cần tới cơ sở dữ liệu. Các ứng dụng vậy gọi là web động, bởi nội dung của chúng luôn thay đổi tùy thuộc vào dữ liệu và người sử dụng. PHP là ngôn ngữ làm được như vậy. Bằng cách chạy chương trình PHP trên máy chủ Web server, bạn có thể tạo ra các ứng dụng có sự tương tác với cơ sở dữ liệu để tạo rat rang web và đây gọi là trang web động.

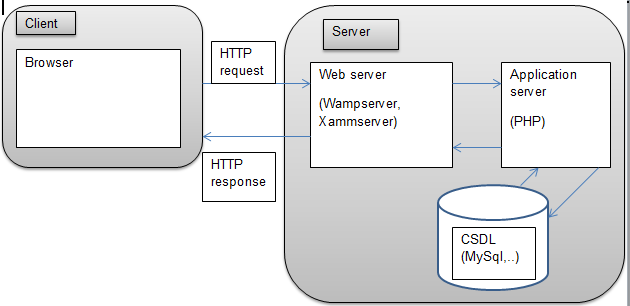
Và dưới đây là mô phỏn hoạt động của web tĩnh (HTML) và web động (PHP)

* Với các trang HTML:



Khi có yêu cầu tới một trang web từ phía người sử dụng (browser). Web server thực hiện các bước sau:

* Đọc yêu cầu từ phía brower.
* Tìm trang web trên server.
* Gửi trang web đó trở lại cho browser(nếu tìm thấy) qua mạng internet hoặc Intranet
* Với các trang PHP:



Khác với các trang HTML, khi một trang PHP được yêu cầu, web server phân tích và thi hành các đoạn mã PHP để tạo ra trang HTML.

Điều đó thực hiện bằng các bước:

* Đọc yêu cầu từ phía browser.
* Tìm trang tren server.
* Thực hiện các đoạn mã PHP trên trang web đó để sửa đổi nội dung của trang.
* Gửi trở lại nội dung cho browser (đây là trang HTML có thể hiển thị được bởi trình duyêt web Internet Explorer hay những trình duyệt web khác nào đó)

Tóm lại:

Sự khác nhau giữa web tĩnh (HTML) và web động (PHP) là:

* Các trang HTML không được thực hiện trên máy chủ Web server.
* Các trang \*.php được viết bằng ngôn ngữ PHP được thực hiện trên máy chủ Web server.
* Những trang viết bằng ngôn ngữ PHP linh động và mềm dẻo hơn những trang viết bằng ngôn ngữ HTML.
  1. Giới thiệu MySql
     1. Mysql là gì?

**MySQL** là hệ quản trị cơ sở dữ liệu tự do nguồn mở phổ biến nhất thế giới và được các nhà phát triển rất ưa chuộng trong quá trình phát triển ứng dụng.

**MySQL** là một trong những ví dụ rất cơ bản về Hệ Quản trị Cơ sở dữ liệu quan hệ sử dụng Ngôn ngữ truy vấn có cấu trúc (SQL).

**MySQL** được sử dụng cho việc bổ trợ PHP, Perl, và nhiều ngôn ngữ khác, nó làm nơi lưu trữ những thông tin trên các trang web viết bằng PHP hay Perl,...

* + 1. Tính năng và lợi thế của MySql
       1. Tính năng.

MySQL rất phức tạp, nhưng giao diện SQL trực giác và dể học.

MySQL là cơ sở dữ liệu tốc độ cao, ổn định và dễ sử dụng, có tính khả chuyển, hoạt động trên nhiều hệ điều hành cung cấp một hệ thống lớn các hàm tiện ích rất mạnh.

* + - 1. Lợi thế:

Với tốc độ và tính bảo mật cao, MySQL rất thích hợp cho các ứng dụng có truy cập CSDL trên internet.

MySQL miễn phí hoàn toàn cho nên bạn có thể tải về MySQL từ trang chủ.

MySQL có nhiều phiên bản cho các hệ điều hành khác nhau: phiên bản Win32 cho các hệ điều hành dòng Windows, Linux, Mac OS X, Unix, FreeBSD, NetBSD, Novell NetWare, SG Irix, Solaris, SunOS, ...

MySQL hỗ trợ các điểm váo là ANSI SQL92 và ODBC mức 0-2 SQL chuẩn.

MySQL hỗ trợ nhiều ngôn ngữ cho việc thông báo lỗi như: Czec, Dutc, English, Estonian, French, German, Hungarian, Italian, Norwegian Nynorsk, Polish, Portuguese, Spanish và Swedish. Ngôn ngữ được hỗ trợ mặc định cho dữ liệu là ISO-8859-1 (Latin1), muốn thay đổi phải sửa trong mã nguồn.

Ngôn ngữ lập trình sử dụng viết các hàm API để thâm nhập cơ sở dữ liệu MySQL có thể là C, Perl, PHP..

Các bảng trong cơ sở dử liệu MySQL có kích thước rất lớn và được lưu ở thư mục Datas. Kích thước lớn tối đa của 1 bảng là 4GB và nó còn phụ thuộc vào kích thước lớn nhất của 1 file do hệ điều hành quy định.

Cơ sở dữ liệu MySQL rất dễ quản lí và có tốc độ xử lí cao hơn tới ba bôn lần so với các hệ cơ sở dữ liệu khác.

MySQL là hệ quản lí cơ sở dữ liệu mô hình quan hệ nó có mã nguồn mở. Nó được cung cấp miễn phí trên các máy chủ UNIX, OS/2 và cả trên Windows.

* + 1. Nhược điểm MySQL:

MySQL không cho phép thực hiện các câu lệnh truy vấn con.

Vi dụ:

SELECT \* FROM tb\_DichVuPhatSinh as dv WHERE dv.id\_dv=(SELECT cthd.id\_dv FROM tb\_CTHD as cthd).

MySQL không hỗ trợ Stored Procedures, Triggers, Transactions, Foreign Keys và Views như các hệ quản trị cơ sở dữ liệu khác.

1. Phân tích ngôn ngữ PHP
   1. Ưu điểm của ngôn ngữ PHP

* PHP thực hiện với tốc độ rất nhanh và hiệu quả. Một Server bình thường có thể đáp ứng được hàng triệu truy cập tới trong một ngày.

PHP hỗ trợ kết nối tới nhiều cơ sở dữ liệu khác nhau:

Postgre SQL, mSQL, Oracle, dbm, filePro, Hyperware, Informix, InterBase, Sybase,.. Ngoài ra còn hỗ trợ kết nối với ODBC thông qua đó có thể kết nối với nhiều ngôn ngữ khác mà ODBC hỗ trợ.

* PHP cung cấp một hệ thống thư viện phong phú: Do PHP ngay từ đầu được thiết kế nhằm mục đích xây dựng và phát triển các ứng dụng trên web nên PHP cung cấp rất nhiều hàm xây dựng xây dựng sẵn giúp thực hiện các công việc rất dễ dàng: gửi, nhận email, làm việc với các cookies, và nhiều thứ khác nữa.
* PHP là một ngôn ngữ rất dễ dùng, dễ học và đơn giản nhiều hơn nhiều so với các ngôn ngữ khác như Per; Java. Nếu đã biết ngôn ngữ C thì mọi việc sẽ hoàn toàn thuận lợi.
* PHP có thể sử dụng được trên nhiều hệ điều hành, chúng ta có thể viết chúng trên Unix, Lunix và các phiên bản của Windowns. Và có thể đem mã PHP này chạy trên các hệ điều hành khác mà không phải sửa đổi lại mã.
* PHP là mã nguồn mở.
  1. Nhược điểm của ngôn ngữ PHP so với các ngôn ngữ khác:
* Mã nguồn không đẹp.
* Chỉ chạy trên ứng dụng web.
  1. Cú pháp và thành phần trong PHP
     1. Cú pháp

PHP cũng có thẻ bắt đầu và kết thúc giống với ngôn ngữ HTML. Chỉ khác, đối với PHP chúng ta có nhiều cách để thể hiện.

Cách 1 : Cú pháp chính:

<?php Mã lệnh PHP ?>

Cách 2: Cú pháp ngắn gọn

<? Mã lệnh PHP ?>

Cách 3: Cú pháp giống với ASP.

<% Mã lệnh PHP %>

Cách 4: Cú pháp bắt đầu bằng script

<script language=php>

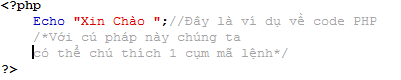
.....

</script>

Mặc dù có 4 cách thể hiện. Nhưng đối với 1 lập trình viên có kinh nghiệm thì việc sử dụng cách 1 vẫn là lựa chon tối ưu.

Trong PHP để kết thúc 1 dòng lệnh chúng ta sử dụng dấu ";"

Để chú thích 1 đoạn dữ liệu nào đó trong PHP ta sử dụng dấu "//" cho từng dòng. Hoặc dùng cặp thẻ "/\*……..\*/" cho từng cụm mã lệnh.

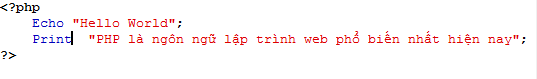


* + 1. Xuất dữ liệu ra trình duyệt:

Để xuất thông tin ra trình duyệt ta làm theo hai cách sau:

+ Echo “PHP là ngôn ngữ lập trình web phổ biến nhất hiện nay”;

+Print “PHP là ngôn ngữ lập trình web phổ biến nhất hiện nay”;



Ta có thể nối hai chuỗi bằng cách sử dụng dấu chấm (.) giữa hai chuỗi mà ta muôn nối.

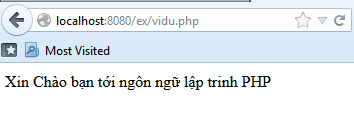
Vi dụ:

<?php

Echo “Xin Chào” . “bạn tới ngôn ngữ lập trinh PHP”;

?>

Kêt quả nó sẽ cho là:



* + 1. Các kiểu dữ liệu

PHP có ba dữ liệu cơ bản: interger, double và string. Ngoài ra còn có các dữ liệu khác (nhưng không phải là kiểu sữ liệu cơ bản) như arrays (các dữ liệu mảng), objects (các dữ liệu đối tượng).

|  |  |  |
| --- | --- | --- |
| Kiểu dữ liệu | Ví dụ | Mô tả |
| Integer | 10 | Một số nguyên |
| Double | 10.00 | Kiểu số thực |
| String | “How are you?” | Một tập hợp các kí tự |
| Boolean | True or False | Giá trị true hoặc False |
| Object | Hướng đối tượng trong PHP | |
| Array | Mảng trong PHP, chứa các tập kí tự. | |

Integer là kiểu chiếm 4 byte bộ nhớ, giá trị của nó khoảng -2 tỷ đến +2 tỷ. Kiểu double là kiểu số thực, phạm vi biểu diễn ±(10-308 ÷ 10308). Kiểu string dùng để biểu diễn các giá trị bao gồm kí tự và con số.

Vi dụ :

1. //Đây là kiểu integer.

1.0 //Đây là kiểu double

"2 "  //Đây là kiểu string

"2 hours" //Đây là kiểu string khác

* + 1. Hằng số:

Hằng số là những giá trị không đổi. Chúng ta thường dùng hằng số đẻ lưu các giá trị không đổi trong suốt chương trình như: nhiệt độ (00C), các giá trị thời gian chỉ sự sự chuyển giao trưa, chiều, tối..

Cũng giống với biến hằng được xem là hợp lệ thì chúng phải đáp ứng 1 số yếu tố :

+ Hằng không có dấu "$" ở trước tên.

+ Hằng có thể truy cập bất cứ vị trí nào trong mã lệnh

+ Hằng chỉ được phép gán giá trị duy nhất 1 lần.

+ Hằng thường viết bằng chữ in để phân biệt với biến

* + - 1. Khai báo hằng số:

Ta dùng hàm **define()** để khai báo hằng số:

Define(“Tên biến hằng số”,”Giá trị biến hằng số”);

Ví dụ:Define(“COMPANY”,”Karaoke Nice”);

Define(“MAU\_VANG”,”#FFFF00”);

Define(“VERSION”,4);

Define(“XD”,”<BR/>”);

Trong các ví dụ trên ta đã dùng hàm define() để khai báo các hằng số.

Chúng ta sẽ sẽ dụng các hằng số trong PHP như sau:

Echo “Employment at ”.COMPANY.XD;

Cách viết trên giống như cách viết sau:

Echo “Employment at ”.COMPANY. “<br/>”;

Chú ý: khi ta dùng hằng số phải cho hằn số nằm ngoài hai dấu mở và đóng nháy đôi ( “ và ”. Trường hợp: Echo “Employment at COMPANY”; thì hằng số COMPANY không có hiệu lực. Khi đó nó sẽ trả về kết quả là : “Employment at COMPANY”.

Hàm **difined()**: hàm này là hàm dùng để kiểm tra xem hằng số đã được khai báo hay chưa.

Vi vụ: if(defined(“MAU\_VANG”)){

Echo “<BODY BGCOLOR=” .MAU\_VANG. “>\n”);

}

* + - 1. Các hằng số đã được định nghĩa trong PHP (Built\_in Constants):

Để hỗ trợ cho ngươi lập trình, PHP cung cấp các hằng số như: các biến môi trường, các biến của Web server Apache...Ngươi dùng có thể dùng hàm phpinfo() để xem thông tin của version PHP đang sử dụng.

Ví dụ:

<HTML>

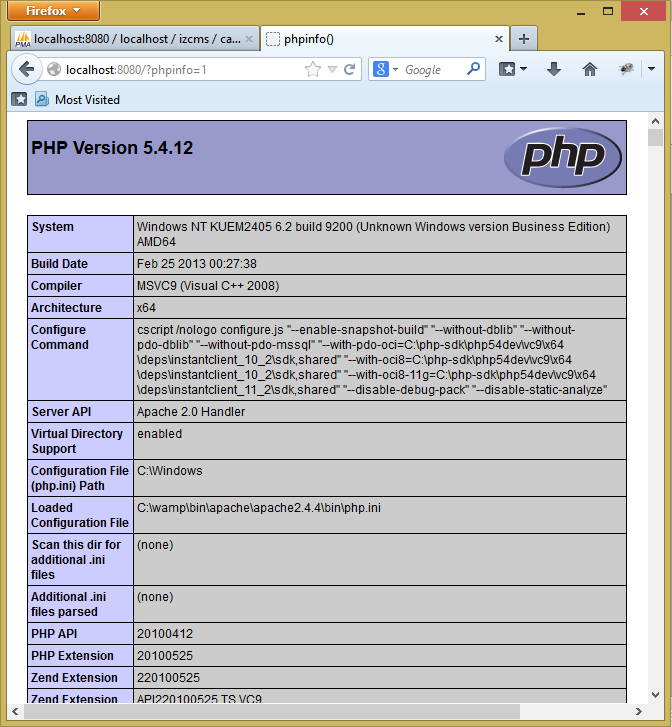
<!—phpinfo() -- >

<BODY>

<?php phpinfo() ;?>

</BODY>

</HTML>



+ Hằng số nguyên: đây là những giá trị có kiểu interger. Vi dụ: 10.

+ Hằng số thực: đây là những giá trị có kiểu số double. Ví dụ: 10.00.

+ Hằng kí tự: đây là những giá trị có kiểu string (là 1 xâu hay 1 chuỗi kí tự được đặt trong dấu nhánh kép). Ví dụ: “Chào mừng bạn đến với ngôn ngữ lập trình web tôt nhất hiện nay”.

* + 1. Toán tử

|  |  |  |  |
| --- | --- | --- | --- |
| Phép toán | Ý nghĩa | Ví dụ | Giải thích |
| + | Phép cộng | 7+3 | Thực hiện phép cộng giữa 7 và 3 kết quả: 10 |
| - | Phép trừ | 7-3 | Thực hiện phép trừ giữa 7 và 3 kết quả: 10 |
| \* | Phép nhân | 7\*3 | Thực hiện phép nhân giữa 7 và 3 kết quả: 10 |
| / | Phép chia | 7/3 | Thực hiện phép chia giữa 7 và 3 kết quả: 10 |
| % | Phép chia lấy dư | 7%3 | Thực hiện phép chia lấy phần dư giữa 7 và 3 kết quả: 1 |

Ta có thể viết các phép toán ngắn gọn như bảng sau:

|  |  |
| --- | --- |
| 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 |

Bảng các phép toán quan hệ:

|  |  |  |  |
| --- | --- | --- | --- |
| Phép toán | Ý nghĩa | Ví dụ | Giải thích |
| == | So sánh bằng | $h == $i | Kiểm tra $h và $i có bằng nhau không |
| < | So sánh nhỏ hơn | $h < $i | Kiểm tra $h và $i có nhỏ hơn nhau không |
| > | So sánh lớn hơn | $h > $i | Kiểm tra $h và $i có lớn hơn nhau không |
| <= | So sánh nhỏ hơn hoặc bằng | $h <= $i | Kiểm tra $h và $i có nhỏ hơn hay bằng nhau không |
| >= | So sánh lớn hơn hoặc bằng | $h >= $i | Kiểm tra $h và $i có lớn hơn hay bắng nhau không |
| != | So sánh khác | $h != $i | Kiểm tra $h và $i có khác nhau không |
| <> | So sánh khác | $h <> $i | Kiểm tra $h và $i có khác nhau không |

Bảng các phép toán logic: phép toán logic cùng với toán hạng tạo thành biểu thức logic. Biểu thức logic có thể có giá trị 1 (true) hoặc 0 (false).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **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 |

* Các phép toán với kiểu string:

Ta sử dụng dấu “.” Để ghép hai biến kiểu string lại với nhau.

Ví dụ: $first\_name = “Nguyễn Văn ”;

$last\_name= “A”;

$full\_name=$first\_name.$last\_name;

Echo $full\_name; //$full\_name = “Nguyễn Văn A”;

Ta có thể ghép 2 xâu như sau:

Echo $last\_name. “ đang học PHP.”;

Khi muốn chèn 1 biến vào trong 1 chuỗi nào đó thì ta cho biến đó vào ngoặc nhọn ({}):Echo “{$last\_name} đang học PHP.”;

* Các phép toán tang giảm.

+ Phép tăng :

$a++:$a được sử dụng rồi mới tăng lên 1.

++$a: $a tăng rồi mới sử dụng.

+ Phép giảm :

$a--:$a được sử dụng rồi mới giảm xuống 1.

++$a: $a giảm rồi mới sử dụng.

* Phép toán điều kiện: (?:)

Phép toán diều kiện với toán hạng tạo nên biểu thức điều kiện. Biểu thức điều kiện có dạng: e1?e2:e3

Nếu điều kiện e1 đúng thì giá trị của biểu thức là e2.

Nếu điều kiện e1 sai thì giá trị của biểu thức là e3.

Ví dụ:

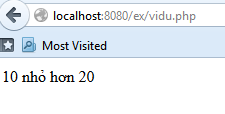
<?php

$a = 10;

$b= 20;

$a>$b ? “{$a} lớn hơn {$b}”: “{$a} nhỏ hơn {$b}”;

?>



* Toán tử sizeof(đối tượng): phép toán sizeof() cho biết kích thước ( tính bằng byte) ô nhớ mà đối tượng chiếm trong bộ nhớ. Đối tượng ở đây có kiểu là integer, double, string.

Ví dụ

$a=20;

Echo sizeof($a); //sẽ in ra ngoài trình duyệt là 4.

* + 1. Thành phần
       1. Biến, chuỗi, mảng
          1. Biến, chuỗi:

Biến được xem là vùng nhớ dữ liệu tạm thời. Và giá trị có thể thay đổi được. Biến được bắt đầu bằng ký hiệu "$". Và theo sau chúng là 1 từ, 1 cụm từ nhưng phải viết liền hoặc có gạch dưới.

* Cũng giống với C/C++, PHP không có giá trị TRUE và FALSE. Các giá trị TRUE được hiểu là giá trị bằng 1 và giá trị FALSE là những giá trị không hoặc rỗng.
* Khi xử dụng biến ta không cần khai báo kiểu dữ liệu của biến.

Vi dụ: $a=1; //$a là 1 biến có kiểu integer.

$a=1.2 ; //$a bây giờ là 1 biến kiểu double.

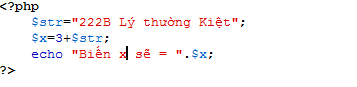
$a ="A"; // $a bây giờ là kiểu string.

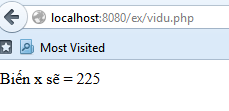
* Nếu như thực hiện phép toán giữa biến kiểu số và biến kiểu string. PHP sẽ coi chuỗi là 1 dãy số như sau:

$str= “222B Lý thường Kiệt”;

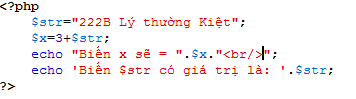
* Như ta đã thấy biến $str có kiểu string và nế cọng số 3 với giá trị nay thì

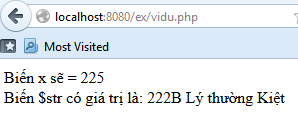
$x=3+ $str; //$x=225



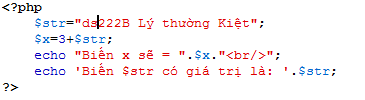


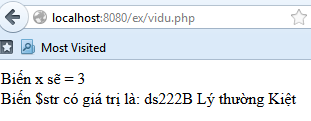
* Khi đó biến $x nhận được giá trị 225 vì PHP đã cộng số 3 với ba số đầu. Nhưng nếu ta in giá trị của biến $str thì nó có giá trị : “222B Lý thường Kiệt”.





Chú ý: phép toán giữa số và chuỗi chỉ đúng khi kí tự đầu của chuỗi là số





* Ta cũng có thể làm thay đổi kiểu giá trị của 1 biến bằng cách ép kiểu

Ví dụ:

$a=11.2; //biến $a có kiểu double

$a=(int)$a; //bây giờ biến $a có kiểu là integer, có giá trị là 11.

$a = (double)$a; //bây giờ biến $a lại có kiểu double, giá trị là 11.0

$b = (string)$a; // biến $b lại có kiểu string, giá trị là “11”.

PHP tự động chuyển đổi kiểu dữ liệu rất tốt. Nếu thật sự cần thiết thì ta mới phải dùng cách trên.

* 1 biến được xem là hợp lệ khi nó thỏa các yếu tố :

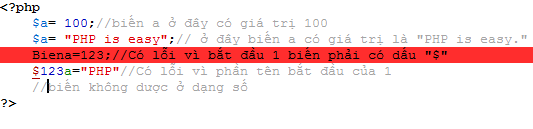
+ Tên của biến phải bắt đầu bằng dấu gạch dưới và theo sau là các ký tự, số hay dấu gạch dưới.

+ Tên của biến không được phép trùng với các từ khóa của PHP.

Trong PHP để sử dụng 1 biến chúng ta thường phải khai báo trước, tuy nhiên đối với các lập trình viên khi sử dụng họ thường xử lý cùng một lúc các công việc, nghĩa là vừa khái báo vừa gán dữ liệu cho biến.

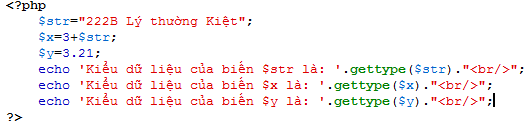
Bản thân biến cũng có thể gãn cho các kiểu dữ liệu khác. Và tùy theo ý định của người lập trình mong muốn trên chúng.

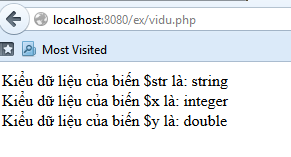
Một số ví dụ về biến :



* Các hàm làm việc với biến:

+ **gettype(Ten\_bien):** là hàm trả về kiểu của một biến nào đó. Giá tri có thể là: “integer”, “double”, “string”, “array”, “object”, “class”, “unknown type”.





+ **settype(ten\_bien,kieu\_du\_lieu)**: hàm này ép dữ liệu cho 1 biến nài đó. Nếu thành công nó trả về giá trị TRUE, ngược lại FALSE.

+ **isset(ten\_bien) và unset(ten\_bien) :** hàm **isset(ten\_bien)** kiểm tra một biến đã được gán giá trị hay chưa, hàm **unset(ten\_bien)** sẽ giải phóng bộ nhớ cho một biến nào đó.

+ **empty(ten\_bien)** : cũng giống như hàm **isset(ten\_bien)**, hàm **empty(ten\_bien)** sẽ trả về giá trị 1 (true) nếu một biến là rỗng và ngược lại 0 (false). Đối với biến là kiểu số giá trị bằng 0 là rỗng, biến string được coi là rỗng nếu xâu đó là xâu rỗng.

* + - * 1. Chuỗi
* Chuỗi là 1 tập hợp của nhiều kí tự.
* Các hàm hỗ trợ trong chuỗi là:

+ strlen($ten\_chuoi): kiểm tra chiều dài của biến $ten\_chuoi.

+ strtoupper($ten\_chuoi):in hoa tất cả kí tự trong biến $ten\_chuoi.

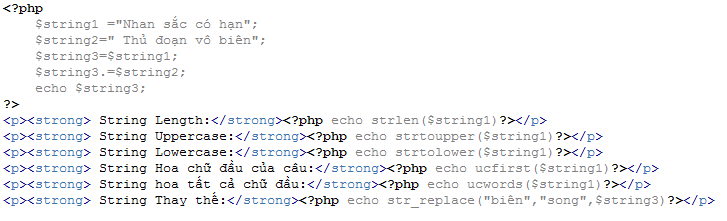
+ strtolower($ten\_chuoi): in thường tất cả các kí tự có trong biến $ten\_chuoi.

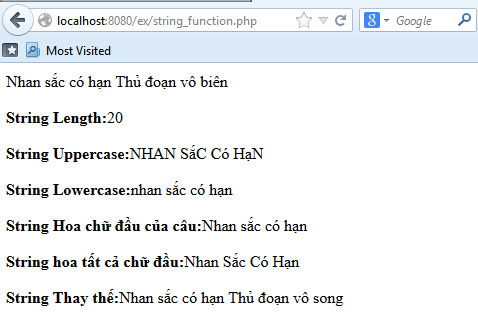
+ ucfirst($ten\_chuoi): in hoa chữ đầu trong biến $ten\_chuoi.

+ ucword($ten\_chuoi): In hoa tất cả từ đầu chữ trong biến $ten\_chuoi.

+ str\_replace(“tu\_can\_thay\_the”, “tu\_thay\_the”, $ten\_chuoi): thay thế tu\_thay\_the vào tu\_can\_thay\_the trong biến $ten\_chuoi.

Ví dụ:





* + - * 1. Mảng

Định nghĩa mảng trong PHP:

Với cú pháp giống với java ta có cách gọi 1 mảng như sau:

$tên\_biến= array("Kenny","Maria","Julia","Kenvin")

Như vậy khi gọi giá trị trong mảng ta có:

Echo $tên\_biến[2]; // Giá trị sẽ cho là Julia.

Ngoài cách khai báo ở trên chúng ta cũng có thể khai báo bằng cách khác như sau:

$tên\_biến[] = "Kenny";

$tên\_biến[] = "Gillian";

$tên\_biến[] = "Charlene";

$tên\_biến[] = "Calvin"

Chúng ta không cần thêm ký tự vào. Vì PHP đã tự xử lý cho chúng ta phần đó.

Để thêm 1 phần tử vào mảng chúng ta cũng có thể sử dụng như sau:

$tên\_biến[] = "Jiro";

Như vậy nếu chúng ta gọi giá trị: echo $tên\_biến[4] thì giá trị sẽ là jiro. Vì nó được PHP thêm vào sau cùng và ngầm hiểu là giá trị tiếp theo.

Ví dụ:

<?php

$a=array("Kenny","Maria","Julia","Kenvin");

echo $a[2]; //Xuất ra giá trị Julia

?>

Khái niệm mảng kết hợp trong PHP

Là các mảng được tạo index bằng các chuỗi , chúng được gợi là các mảng kết hợp.

Ví dụ:

<?

$a= array (name => "Kenny", job => "Teacher", age=>"45", email => "webmaster@vietchuyen.com.vn")

?>

Như vậy để truy cập giá trị của mảng trên chúng ta không thể sử dụng cách cũ được.

Ví dụ: Để lấy giá trị của tuổi tôi không thể sử dụng cách trên : echo $a[2].

Vì chúng ta đã gán giá trị của index cho 1 tên gọi khác. Do vậy để lấy giá trị của tuổi ta sử dụng như sau: echo $a[age] // Kết quả sẽ cho ra 45.

<?

$tên\_biến= array (name => "Kenny", job => "Teacher", age=>"45", email => "webmaster@vietchuyen.com.vn")

echo $a[age];

?>

Phép lặp trong mảng:

Cú pháp:

foreach($array as $temp)

{ Hành Động }

Trong đó $array là mảng mà ta muốn thực thi việc lặp dữ liệu, và $temp là một biến trong đó ta sẽ tạm thời lưu trữ mỗi phần tử.

Ví dụ:

<?php

$name= array("Kenny", "Jiro", "Calvin", "Gillian");

foreach ($name as $test)

{ echo "$test<br>"; }

?>

Lặp lại qua một mảng kết hợp:

Cú pháp:

Foreach($array as $key=>$value)

{ Hành Động }

Trong đó $array là mảng mà ta muốn thực thi việc lặp dữ liệu, $key là một biến vốn tạm thời chứa mỗi khóa, và $value là một biến vốn tạm thời chữa mỗi giá trị.

Ví dụ:

<?php

$person= array(name=>"Kenny", job=>"Teacher", email=>"whiletionvn@gmail.com", age=>"38");

foreach($person as $key=>$test)

{

echo "Key: $key. Gia Tri: $test<br>";

}

?>

Các hàm hỗ trợ trong PHP:

+ Hàm gộp mảng:

Cú pháp: array\_merge($mang1, $mang2);

+ Hàm tách mảng:

Cú pháp: array\_slice($mang, vị trí tách, số lượng tách);

+ Hàm sắp xếp mảng:

Cú pháp: sort($mảng);

+ Hàm sắp xếp mảng theo chỉ mục:

Cú pháp: ksort($mảng);

+ Hàm tìm phần tử có tồn tại trong mảng hay không ?.

Cú pháp: in\_array(giá trị tìm, $mảng);

* + - 1. Vòng lặp
         1. Mệnh đề điều kiện if-else:

Mệnh đề điều kiện:Là biểu thức dùng kiểm tra 1 sự kiện. Nếu chúng thỏa điều kiện đó thì sẽ thực thi một hành động. Ngược lại sẽ là một hành động khác.

* Mẫu 1: Áp dụng cho trường hợp có 1 điều kiện và 1 công việc xử lí.

Cú pháp:

**If(<biểu thức điều kiện>)**

**Khối lệnh 1;**

**Khối lệnh 2;**

Nguyên tắc hoạt động: Nếu biểu thức điều kiện đúng thì thực hiện khối lệnh 1, sau đó thực hiện khối lệnh 2. Ngược lại nếu biểu thức điều kiện sai thì bỏ qua khối lệnh 1 và thực hiện khối lệnh 2.

* Mẫu 2: Áp dụng cho trường hợp có 1 điều kiện và 2 lựa chọn công việc xử lí.

Cú pháp:

**If(<biểu thức điều kiện>)**

**Khối lệnh 1;**

**else**

**Khối lệnh 2;**

**Khối lệnh 3;**

Nguyên tắc hoạt động: Nếu biểu thức điều kiện đúng thì thực hiện khối lệnh 1, sau đó thực hiện khối lệnh 3 bỏ qua khối lênh 2. Ngược lại nếu biểu thức điều kiện sai thì thực hiện khối lệnh 2, sau đó thực hiện khối lệnh 3 bỏ qua khối lệnh 1.

* Mẫu 3:(if..else long nhau): Áp dụng cho trường hợp có nhiều lựa chọn khác nhau.

Cú pháp:

**If(<biểu thức điều kiện 1>)**

**Khối lệnh 1;**

**Else**

**If(<biểu thức điều kiện 2>)**

**Khối lệnh 2;**

**Else**

**….**

**Khối lệnh 3;**

Để áp dụng mẫu 3, cần xác định biểu thức điều kiện của bài toán rồi sắp xếp thứ tự lồng nhau cho phù hợp.

Ví dụ:

<?php

$gender= “nu”;

if($gender == 'nu') {

echo "Chao chi!";// $gender == 'nu' đúng

}

else {

echo "Chao anh!"; // $gender == 'nu' sai

}

Echo “Không có giới tính”;

?>

* + - * 1. Câu lệnh switch:
* Áp dụng trong trường hợp muốn chọn một trong các giá trị của biểu thức để thực hiện lệnh. Giá trị của biểu thức có thể 1 chuỗi hoặc 1 số.
* Mẫu 1:

**Switch(Biểu thức)**

**{**

**Case value1:**

**Khối lệnh 1;**

**Break;**

**Case value2:**

**Khối lệnh 2;**

**Break;**

**…..**

**Case valuek:**

**Khối lệnh k;**

**Break;**

**}**

* Mẫu 2:

**Switch(Biểu thức)**

**{**

**Case value1:**

**Khối lệnh 1;**

**Break;**

**Case value2:**

**Khối lệnh 2;**

**Break;**

**…..**

**Case valuek:**

**Khối lệnh k;**

**Break;**

**Defaut:**

**Khối lệnh k+1;**

**Break;**

**}**

Nguyên lí hoạt động:

* Trình thông dịch sẽ tính giá trị của biểu thức rồi so sánh với các value, nếu bằng giá trị nào thì thực hiện giá trị đó.
* Sự khác nhau giữa mẫu 1 và mẫu 2 là: ở mẫu 2khi so sánh giá trị của biểu thức với các value, nếu không khớp thì thực hiện lệnh trong default.
* Trong trường hợp có nhiều value khác nhau mà cùng thưc hiện một khối lệnh thì liệt kê các value liên tiếp nhau và cách nhau dấu phẩm.

Case value 1, value 2, …, value k: khối lệnh; break;

Ví dụ:

<head>

<title>Switch</title>

</head>

<body>

<ul>

<li><a href="">Home</a></li>

<li><a href="">Forum</a></li>

<?php

$user\_level = 0;

switch ($user\_level) {

case 1:

echo "<li><a href=''>User Profile</a></li>";

break;

case 2:

echo "<li><a href=''>Edit Profile</a></li>";

break;

case 3:

echo "<li><a href=''>Delete Profile</a></li>";

break;

default:

echo "<li><a href=''>Register</a></li>";

break;

}

?>

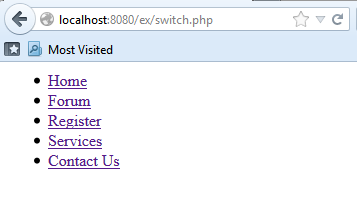
<li><a href="">Services</a></li>

<li><a href="">Contact Us</a></li>

</ul>

</body>

</html>



* + - * 1. Vòng lặp For:
* Vòng lặp for: thường áp dụng cho số lần lặp biết trước.

Cú pháp:

**For(biểu thức 1, biểu thức 2, biểu thức 3)**

**{**

**Khối lệnh 1;**

**}**

**Khối lệnh 2;**

Trong đó:

Biểu thức 1: chứa giá trị khởi tạo cảu biến điều khiển.

Biểu thức 2:chứa biểu thức điều kiện lặp

Biểu thức 1:chứa biểu thức tăng hoặc giảm biến điều khiển.

*Nguyên tắc hoạt động:*

* Trình thông dịch gán giá trị khởi tạo cho biến điều khiển, kiểm tra biểu thức 2, nếu đúng thì thực hiện khối lệnh 1, chuyển tiếp thực hiện biểu thức 3, tiếp tục kiểm tra biểu thức 2 và tiếp tục…
* Nếu biểu thức 2 có giá trị sai thì chương trình thoát khỏi vòng lặp và thực hiện khối lệnh 2.
* Nếu khối lệnh 1 chứa câu lệnh Break thì chưa trình sẽ thoát khỏi vòng lặp for và thực hiện khối lệnh 2.

Ví dụ:

<?php

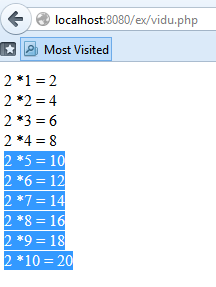
$a=2;

For($i=1;$i<=10;$i++){

Echo “$a \*$i = ”. $a \*$i.”<br/>”;

}

?>



<html>

<head>

<meta http-equiv="content-type" content="text/html; charset=utf-8"/>

<title>Strat up</title>

</head>

<body>

<table border="1">

<?php

For($i=1;$i<=10;$i++){

echo "<tr>";

for($a=1;$a<=10;$a++){

echo "<td>";

Echo "$a \*$i = ". $a \*$i."<br/>";

echo "</td>";

}

echo "</tr>";

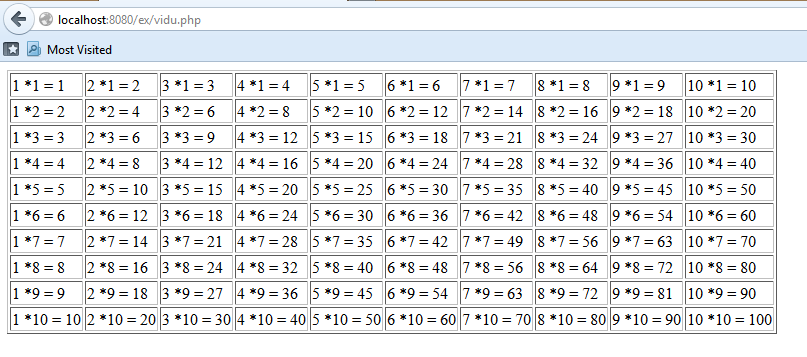
}

?>

</table>

</body>

</html>



* + - * 1. Vòng lặp while:
* Vòng lặp while thường áp dụng cho số vòng lặp không xác định. Vòng lăp while kiểm tra điều kiện trước khi thực hiện lệnh.
* Cú pháp:

**While(biểu thức điều kiện)**

**{**

**Khối lệnh 1;**

**}**

**Khối lệnh 2;**

Nguyên tắc hoạt động:

* Trình thông dịch kiểm tra biểu thức điều kiện, nếu đúng thì thực hiện khối lệnh 1, sau đó quay lại kiểm tra biểu thức điều kiện, và tiếp tục…, nếu sai thì thực hiện khối lệnh 2.
* Như vậy khối lệnh 1 có thể không thực hiện lần nào nếu ngay từ đầu biểu thức điều kiện sai.
* Thường khối lệnh 1 chứa lệnh làm thay đỏi giá trị của biểu thức điều kiện để thoát ra khỏi vòng lặp, hoặc chứa lệnh break để thoát khỏi vòng lặp *while*.

Ví dụ:

<html>

<head>

<title>While Loop</title>

<style type="text/css">

ul li {display: inline; margin: 0 5px;}

</style>

</head>

<body>

<ul class="pagination">

<?php

$count = 0;

while($count <= 10) {

if($count == 7) {

echo "...";

} else {

echo "<li><a href='{$count}'>{$count}</a></li>";

}

$count++;

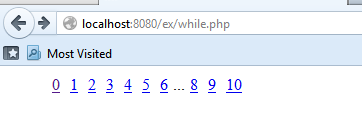
}// End While loop

?>

</ul>

</body>

</html>



* + - * 1. Vòng lặp Do..while:
* Vòng lặp do..while thường áp dụng cho số vòng lặp không xác định. Vòng lăp do…while thực hiện lệnh trước khi kiểm tra điều kiện.

Cú pháp:

**Do**

**{**

**Khối lệnh 1;**

**}While(biểu thức điều kiện);**

Khối lệnh 2;

Nguyên tắc hoạt động: trình thông dịch thực hiện khối lệnh 1 rồi kiểm tra biểu thức điều kiện, nếu đúng thì thực hiện tiếp khối lệnh 1, nếu sai thì thoát khỏi vòng lặp và thự hiện khối lệnh 2Ø

Ví dụ:

<html>

<head>

<title>While Loop</title>

<style type="text/css">

ul li {display: inline; margin: 0 5px;}

</style>

</head>

<body>

<?php

echo "<SELECT name='num'>\n";

$i=0;

$total=10;

do{

echo("\t<OPTION value=$i>$i</OPTION>\n");

}while(++$i<$total);

echo("</SELECT>\n");

?>

</body>

</html>

* + - * 1. Lệnh Break
* Là rẽ nhánh không điều kiện và thường dùng để thoát khỏi thân của switch, while, do..while, for.
* Lện break chỉ cho phép thoát khỏi thân các lệnh bên trong nhất chứa nó.
  + - * 1. Lệnh continue:
* Là lệnh rẽ nhánh không điều kiện. Lệnh thường dùng để bắt đầu lại 1 chu trình mới trong các lệnh for, while, do.. while mà không càn thực hiện hết toàn bộ thân của lệnh tạo chu trình.
  + - 1. Hàm
         1. Cách xây dựng hàm

Function tên\_hàm(danh sách tham số){

Khối lệnh trong hàm;

}

* Định nghĩa hàm không nhất thiết phải nằm ngoài thân mọi hàm, trong hàm có thể có hàm khác. Nhưng việc sử dụng một hàm không khác nhau giữa xây dựng hàm trong thân một hàm và ngoài mọi hàm.
* Tên hàm tùy đặt và khác tên hàm chuẩn.
* Hàm có thể có giá trị trả về hoặc không.
* Các câu lệnh được quyền gọi bất kỳ hàm nào đã được khai báo và đã được định nghĩa.
* Return:

+ Trả về giá trị về cho nơi gọi hàm

+ Là nơi báo kết thúc hàm.

* + - * 1. Gọi hàm
* Hàm phải được xây dựng (khai báo) trước khi gọi.
* Khi gọi hàm, nếu có giá trị trả về thường được đặt trong biểu thức.
  + - * 1. Biến toàn cục và biến cục bộ
* Thông thường PHP coi các biến được sử dụng trong thân hàm là biến cục bộ. Nghĩa là các biến trong thân hàm không làm thay đổi giá trị của biến bên ngoài hàm đó. Muốn hàm làm thay đổi giá trị của biến bên ngoài ta cần khia báo global trước biến đó trong thân hàm.

Ví dụ:

**$position= “m”;**

**Function change\_pos()**

**{**

**$position= “s”;**

**}**

**Change\_pos();**

**Echo “$position”;** //kết quả ngoài trình duyệt sẽ là “m”.

Ta thấy biến $position không thay đổi giá trị khi ta gọi hàm change\_pos().

**$position= “m”;**

**Function change\_pos()**

**{**

**Global $position;**

**$position= “s”;**

**}**

**Change\_pos();**

**Echo “$position”; //** kết quả ngoài trình duyệt sẽ là “s”.

Ta có thể viết như trên hoặc ta có thể viết :

**$position= “m”;**

**Function change\_pos()**

**{**

**Global [$position]= “s”;**

**}**

**Change\_pos();**

**Echo “$position”;**

* + - 1. Post, get, cookie, session
         1. Post
* Post là 1 biến của PHP ($\_POST): dùng để lấy giá trị trong Form.

Ví dụ:

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8"/>

<title>PHP Calendar</title>

</head>

<body>

<?php

if(isset($\_POST['submit'])) {

// Xu ly form

//if($\_POST['gender'] == 'nu') {

//echo "Chao chi!";

//}elseif($\_POST['gender'] == 'nam') {

//echo "Chao anh!";

echo ($\_POST['gender']=='nu') ? "chao chi" : "chao anh";

}

?>

<form action="" method="post">

<select name="gender">

<option>Giới tính</option>

<option value="nu">Nữ</option>

<option value="nam">Nam</option>

</select>

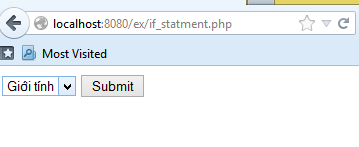
<input type="submit" name="submit" value="Submit" />

</form>

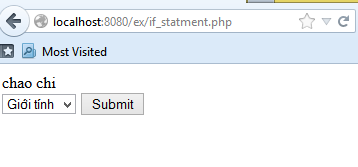
</body>

</html>

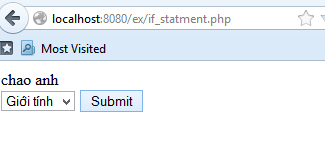
Khi trang mới load



Khi chọn option là nữ



Khi chọn option là nam



* + - * 1. Get
* GET là 1 biến trong PHP ($\_GET): dùng để lấy giá trị từ thanh địa chỉ, lây s giá trị các biến sau dấu (?).

Vi dụ: htttp//google.com?**seach=dfd&…&id=1**//seach, … và id là nhũng biến $\_GET có thể lấy được.

Chú ý: muốn khai báo nhiều biến trên địa chỉ (đường link) thi khoảng cách giữa các biến là dấu (&).

* + - * 1. Cookie
* Cookie là những mẫu tin nhỏ mà trang script có thể chứa trên các máy khách (client). Bạn có thể thiết lập một cookie trên 1 máy người dùng bằng cách gửi một “HTTP header” có chứa dữ liệu theo dạng sau:

Set-Cookie:Name=VALUE;

[expires=DATE;][path=PATH;][domain=DOMAIN\_NAME;][secure]

Câu lện này sẽ tạo ra 1 cookies có tên gọi là NAME với giá trị là VALUE. Trường expires sẽ thiết lập ngày mà cookie sẽ hết hiệu lực, path và domain có thể được sử dụng để chỉ định các URL(nơi mà cookie sẽ được gửi đi). Từ khóa secure có nghĩa là cookie sẽ không gửi đi trên quá một kết nối HTTP chuẩn.

* Khi một browser kết nối tới URL, trước tiên nó kiểm tra các cookie đã được lưu trên máy. Nếu có bất kì một cookie nào có lien quan đến địa chỉ URL vừa được kết nối, chúng sẽ được chuyển trở lại server.
  + - * 1. Session
* Session là nhũng mẫu tin hược lưu trong 1 phiên làm việc. Một cách khác quản lý người sử dụng là session. Session được hiểu là khoảng thời gian người sử dụng giao tiếp với 1 ứng dụng. Một session được bắt đầu khi người sử dụng truy cập vào ứng dụng lần đầu tiên, và kết thúc khi người sử dụng thoát khỏi ứng dụng. Mỗi session sẽ có được cấp một định danh (ID) khác nhau và nội dung được lưu trong thư mục thiết lập trong file php.ini (tham số session.save\_path).

Nhưng bước cơ bản của việc sử dụng session:

+ Bắt đầu 1 session: sử dụng hàm **Bool Session\_start()** để kiểm tra 1 session đã được tạo hay chưa.

+ Đăng kí những biến session: sử dụng hàm **Bool Session\_register(biến session cần lưu)** hàm này sẽ lưu biến session cần lưu vào. Chú ý không đượ sử dụng kid tự $ trong đăng kí một biến.

Ví dụ:

$var=5;

Session\_register(“var”);

+ Sử dụng biến session sử dụng hàm: $HTTP\_SESSION\_VARS(“tên session”), để kiểm tra session đã tồn tai hay chưa ta dùng hàm: session\_is\_registered(tên session).

+ Hủy bỏ biến Session và kết thúc Sesion để kết thúc 1 session ta dùng hàm session\_unset();//kết thúc tất cả các session.

Session\_destroy();//kết thúc 1 sesion.

* + 1. Bảo mật
    2. Khai báo tiền xử lí include và require:
* Để sử dụng các đoạn mã ở ngoài, chúng ta có thể sử dụng khai báo tiền xử lí include và require. Cho phép chúng ta xây dựng các hàm các hằng số và bất kì đoạn mã nào sau đó có thể chèn vào các đoạn cript.
* Require khác với include là: nó có thể làm thay đổi nội dung của trang hiện tại khi biên dịch, các trang này dùng để khai báo các biến, các hằng số hay các đoạn mã đơn giản giản không có vòng lặp. Khi đó include cho phép thực hiện câu lệnh phức tạp –có câu lệnh tạo chu trình. Nó chỉ sử dụng các hàm như những hàm ngoài của chương trình.
  1. Các hàm cơ bản làm việc với MYSQL:
     1. Các hàm kết nối đến MYSQL SERVER:
* PHP cung cấp hàm kết nối với CSDL MySQL: mysql\_connect và mysql\_pconnect

+ **mysql\_connect()**: hàm này tạo ra một liên kết với máy chủ MySQL.

Cú pháp:

mysql\_connect(string [hostname[:port][:/path\_to\_sockket]],string [user\_name], string [password]);

Trong đó:

* Hostname: tên máy chủ cơ sở dữ liệu, nơi trang web sẽ chứa cơ sở dữ liệu. Giá trị ngầm định là “localhost”.
* :port: địa chỉ cổng, nơi bộ máy CSDL lắng nghe yêu cầu. Giá trị ngầm định là “:3306”.
* :path\_to\_socket : cũng giống như :port nhưng chỉ cho hệ điều hành UNIX. Giá trị ngầm định là: “:/tmp/mysql.sock”.
* User\_name: tên người sử dụng được kết nối vào bộ máy CSDL.
* Password : mật khẩu người sử dụng được kết nối vào cơ sở dữ liệu.

Hàm này trả về số định dạng kết nối thành công, giá tri 0 (false) nếu kêt nối có lỗi. Mã số nhận dạng này sẽ được sử dụng cho tất cả yêu cầu tới bộ máy cơ sở dữ liệu sau này.

Sự kết nôi này đóng khi gọi hàm mysql\_close() hoặc kết thúc đoạn PHP cript.

+ **mysql\_pconnect()** : hàm này tạo ra 1 liên kết bền vững với máy chủ MySQL.

Cú pháp:

mysql\_connect(string [hostname[:port][:/path\_to\_sockket]],string [user\_name], string [password]);

Tham số và gia trị trả về của hàm này cũng giống như hàm mysql\_connect(). Sự khác biệt giữa hai hàm này là lien kết tới máy chủ MySQL không bị đóng kể cả khi kết thúc kịch bản (script) PHP hay gọi hàm mysql\_close(). Mục đích của hàm này là luôn liên kết với máy chủ MySQL do luôn có sự yêu cầu tới máy chủ, tránh cho máy chủ phải tìm mã số định dạng mới từ đó giảm thời gian truy cập.

Chú ý: Hàm chỉ thực thi được khi PHP được cấu hình như 1 module Web server.

+ **mysql\_close(int [link\_identifier]):**hàm này hủy bỏ sự kết nối tới máy chủ MySQL.

Trong đó : link\_indentifier là mã số xác nhận dạng được tạo ra bởi hàm mysql\_connect(). Hàm trả về 2 giá trị true nếu thành công, ngược lại false.

* + 1. Các hàm thao tác trên CSDL:
* **Mysql\_create\_db(string name, int [link\_identifier])**: hàm tạo CSDL.
* **Mysql\_drop\_db(string name, int [link\_identifier]):** hàm xóa CSDL.
* **Mysql\_select\_db(string name, int [link\_identifier]):** hàm cho cơ sở dữ liệu hoạt động.

Trong đó:

+ name là tên database.

+ link\_identifier : mã nhận dạng được cấp bởi hàm mysql\_connect().

Lưu ý: tất cả câu lệnh thao tác trên CSDL có thể sử dụng thông qua hàm mysql\_query(string “câu truy vấn”);

* + 1. Các hàm thao tác trên dữ liệu:
* **Mysql\_query(string query, [int link\_identifier]), Mysql\_db\_query(string database, string query, [int link\_identifier])**  là 2 hàm dùng để gửi câu lệnh SQL tới máy chủ. Khác nhau giữa hai hàm nay là: **Mysql\_db\_query()**  đã chỉ rõ câu lệnh thực hiện trên dữ liệu nào nên trước đó không cần thực hiện hàm mysql\_select\_db(), còn **Mysql\_query()** ngược lại.
* **Mysql\_insert\_id([link\_identifier])** lấy giá trị được sinh ra ở câu truy vấn INSERT trước đó.
* **Mysql\_fetch\_rows(int result\_identifier):** hàm trả về một mảng là giá trị của một bảng ghi hiện tại với chỉ số là số thứ tự của các trường (chỉ số bắt đầu từ 0). Sau đó hàm sẽ trỏ tới bảng ghi tiếp theo cho tới khi gặp babgr ghi cuối cùng hàm trả về giá trị false. Để truy xuất tới các giá trị của cột ta viết: tên\_mảng [số thứ tự ].
* **Mysql\_fetch\_array(int result\_identifier [, int result\_type]):** hàm trả về là mảng là giá trị của bảng ghi hiện tại, sau đó hàm trỏ tới bảng ghi tiếp theo cho tới khi gặp bảng ghi cuối cùng hàm trả về giá trị false.
* **Mysql\_fetch\_object(int result\_identifier):** hàm trả về một đối tượng là giá trị của 1 bảng ghi hiện thời. Sau đó hàm trỏ tới bảng ghi tiếp theo cho tới khi gặp bảng ghi cuối cùng hàm trả về giá trị false. Để truy xuất tới giá trị của cột ta viết tên\_object -> tên\_cột.
* **Mysql\_fetch\_assoc(int result\_identifier):** lấy 1 dòng kết quả như mảng liên kết.
* Trong đó:

+ query là câu lênh truy vấn SQL.

+ link\_identifier : mã nhận dạng được cấp bởi hàm mysql\_connect().

+ result\_identifier: là mã số trả về của hàm mysql\_query() hoặc mysql\_db\_query().

+ result\_type: là hằng số có thể nhận các giá trị sau:

* MYSQL\_NUM: chỉ trả về 1 mảng chứa các chỉ số là số (giồng như hàm mysql\_fetch\_row()).
* MYSQL\_ASSOC: chỉ trả giá trị mảng liên kết.
* MYSQL\_BOTH: trả lại mảng chứa đựng các chỉ số gồm cả các con số và chỉ số liên kết.

1. Dome website quản lí karaoke Nnice (Quản lí dịch vụ phát sinh)
   1. Phân tích yêu cầu nghiệp vụ (yêu cầu đề bài):

Trong quản karaoke Nnice yêu cầu quản lí những dịch vụ phát sinh của phòng đó thông qua hóa đơn của từng phòng. Để quản lí được dịch vụ phát sinh của mỗi phòng ta cần quản lí:

User: là tên của nhân viên hoặc quản lí, khi lập hóa đơn ta dựa vào id\_user để biết ai lập hóa đơn này nếu có truy cứu tránh nhiệm khi hóa đơn đó bị sai hoặc gặp vấn đề nào đó. User gồm có các thuộc tình là: id\_user, name\_user, first\_name, last\_name, pass, email, sdt, dia\_chi, so\_thich.

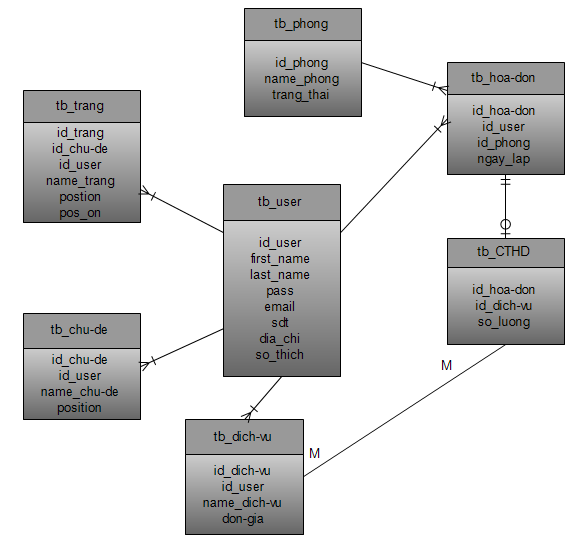
Dịch vụ phát sinh: ta quản lí các dịch vụ phát sinh có trong karaoke này. Dựa vào id\_dich\_vu ta có thể biết đó là dịch vụ nào giá bao nhiêu. Nó bao gồm những thuộc tính là: id\_dich\_vu, id\_user, name\_dich\_vu, don\_gia, dvt.

Để biết phòng đang hoạt động hay trống (không hoạt động) ta dựa vào trạng thái của nó nếu: trạng thái (no) thì phòng đang trống, ngược lại nếu trạng thái là (yes) thì phòng đang hoạt động. Phòng gồm có thuộc tính là: id\_phong, name\_phong, lau, trang\_thai.

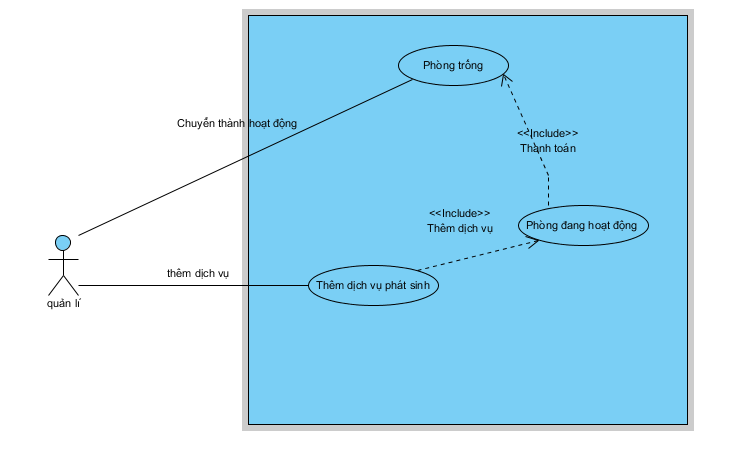
Để biết hóa đơn của phòng đã thanh toán chưa thì ta dựa vào trạng thái của hóa đơn đó nếu : trạng thái (yes) thì hóa đơn đã được thanh toán, ngược lại nếu trạng thái (no) thì hóa đơn đó chưa được thanh toán. Khi hóa đơn đơn của phòng đó được thanh toán thì trạng thái của phòng đó chuyển thành đang trống có nghĩa là trạng thái của nó là (no). Hóa đơn gồm những thuộc tính là: id\_hoa\_don, id\_user, id\_phong, ngay\_lap, trang\_thai.

Chi tiết hóa đơn: giúp ta biết chi tiết thông tin phòng đó có bao nhiêu dịch vụ phát sinh và số lượng của mỗi dịch vụ là bao nhiêu. Nó bao gồm những thuộc tính là: id\_hoa\_don, id\_dich\_vu, so\_luong.

* 1. Sơ đồ cơ sở dữ liệu:



* 1. Sơ đồ Usercase



* 1. Bảng và kiểu dữ liệu
* Dịch vụ phát sinh (tb\_DichVuPhatSinh):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Primary key | Not Null | Index |
| Id\_dv | Int | Yes | yes | Yes |
| Id\_user | Int | No | Yes | Yes |
| Name\_dv | Varchar(100) | No | Yes | No |
| Don\_gia | float | No | Yes | No |

* Bảng user (tb\_User):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Primary key | Not Null | Index |
| Id\_User | Int | Yes | Yes | Yes |
| First\_Name | Varchar(30) | No | Yes | No |
| Last\_name | Varchar(40) | No | Yes | No |
| Pass | Varchar(100) | No | Yes | Yes |
| Email | Varchar(100) | No | Yes | Yes |
| SDT | Int | No | NO | No |
| Dia\_Chi | Varchar(200) | No | NO | No |
| So\_Thich | Text | No | NO | No |

* Hóa đơn (tb\_HoaDon):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Primary key | Not Null | Index |
| Id\_HD | Int | Yes | Yes | Yes |
| Id\_user | Int | No | Yes | Yes |
| Id\_Phong | Int | No | Yes | Yes |
| Ngay\_Lap | Datetime | No | Yes | No |

* Chi tiết hóa đơn (tb\_ChiTietHoaDon):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Primary key | Not Null | Index |
| Id\_CTHD | Int | Yes | Yes | Yes |
| Id\_HD | Int | No | Yes | Yes |
| Id\_DV | Int | No | Yes | Yes |
| So\_luong | Int | No | Yes | No |

* Phòng (tb\_Phong):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Primary key | Not Null | Index |
| Id\_phong | Int | Yes | Yes | Yes |
| Name\_phong | Varchar(100) | No | Yes | NO |
| Trang\_Thai | Bool | NO | Yes | No |

* Chủ đề (tb\_ChuDe):

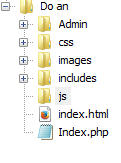
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Primary key | Not Null | Index |
| Id\_CD | Int | Yes | Yes | Yes |
| Id\_User | Int | No | Yes | Yes |
| Name\_CD | Varchar(100) | No | Yes | No |
| Position | Tinyint | No | NO | No |

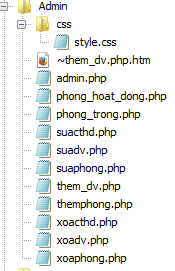
* Trang (tb\_Trang):

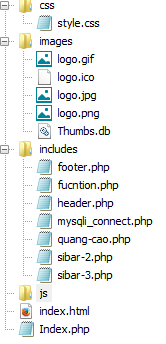
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Tên cột | Kiểu dữ liệu | Primary key | Not Null | Index |
| Id\_Trang | int | Yes | Yes | Yes |
| Id\_ChuDe | Int | No | Yes | Yes |
| Id\_User | Int | No | Yes | Yes |
| Name\_Trang | Varchar(100) | No | Yes | No |
| Position | Tinyint | No | Yes | No |
| Pos\_on | Datetime | No | Yes | No |

* 1. Dome:

* + 1. Sơ đồ cây của đồ án:







* + 1. Code:
       1. Header.php

|  |
| --- |
| <!DOCTYPE HTML>  <html>  <head>  <meta http-equiv="content-type" content="text/html" charset="utf-8"/>  <meta name="author" content="GallerySoft.info" />    <link rel="stylesheet" href="css/style.css"/>    <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.8.0/jquery.min.js"></script>  <title>KARAOKE Nnice</title>  <!--<link type="image/x-icon" href="images/logo.ico" rel="icon"/>  <link type="image/x-icon" href="images/logo.ico" rel="shortcut icon"/>-->  </head>  <body>  <div id="wrap-page">  <div id="header" >  <div id="login-logo">  <div id="logo"><a href="index.php"><h1>Karaoke Nnice</h1></a></div>  <!--<div id="login">  <form action="" method="POST">  <h3>Đăng nhập</h3>  Tên user: <input type="text" value="" name="dn"/><br />  Password: <input type="text" value="" name="pass"/>  <input type="submit" value="Đăng nhập" name="submitdn"/>  </form>  </div>-->  </div>        </div><!--end header-->  <div id="menu-search" class="group">  <div id="sibar-1" >  <ul>  <li><a href="#">Home</a></li>  <li><a href="#">About</a></li>  <li><a href="#">Services</a></li>  <li><a href="#">Content us</a></li>  </ul>  </div><!--end sibar-1-->  <div id="search">  <form action="" method="get">  <input style="text" name="s" value=""/>  <input type="submit" value="Search" name="submit"/>  </form>  </div>  </div><!--end search-menu-->  <div id="content-info"> |

* + - 1. Function.php

|  |
| --- |
| <?php  // Kiem tra xem ket qua tra ve co dung hay khong?  function confirm\_query($result, $query) {  global $dbc;  if(!$result && !LIVE) {  die("Query {$query} \n<br/> MySQL Error: " .mysqli\_error($dbc));  }  }    //lay dia chi hien tai tren thanh dia chi  function getURL(){  $dc="{$\_SERVER['HTTP\_HOST']}"."{$\_SERVER['PHP\_SELF']}"."?"."{$\_SERVER['QUERY\_STRING']}";  if(!empty($dc)){echo $dc;}  }    function getCurrentPageURL() {  $pageURL = 'http';  if (!empty($\_SERVER['HTTPS'])) {if($\_SERVER['HTTPS'] == 'on'){$pageURL .= "s";}}  $pageURL .= "://";  if ($\_SERVER["SERVER\_PORT"] != "80") {  $pageURL .= $\_SERVER["SERVER\_NAME"].":".$\_SERVER["SERVER\_PORT"].$\_SERVER["REQUEST\_URI"];  } else {  $pageURL .= $\_SERVER["SERVER\_NAME"].$\_SERVER["REQUEST\_URI"];  }  return $pageURL;  }    ?> |

* + - 1. Footer.php

|  |
| --- |
| </div><!--end center-info-->  <div id="footer">  <hr />  <p>© 2013 - Design by Nguyễn Văn Kha</p>  </div><!--end foter-->  </div><!--end wraph-page-->  </body>  </html> |

* + - 1. Sibar-3.php

|  |
| --- |
| <div id="sibar-2">  <ul class="menu">  <li><a href="phong\_hoat\_dong.php">Phòng đang hoạt động</a></li>  <li><a href="phong\_trong.php">Phòng trống</a></li>  <li><a href="themphong.php">Thêm phòng</a></li>  <li><a href="them\_dv.php">Thêm dịch vụ</a></li>  </ul>  </div><!--end sibar-2 --> |

* + - 1. Mysqli\_connect.php

|  |
| --- |
| <?php  //ket noi voi csdl  $dbc = mysqli\_connect('localhost','root','','quan-li-dich-vu-phat-sinh-karaoke');    //neu ket noi khong thanh cong thi bao loi ra trinh duyet  if(!$dbc){  trigger\_error("Not connect DB: " . mysqli\_connect\_error());  }else{  //dat phuong thuc ket noi la utf-8  mysqli\_set\_charset($dbc,'UTF8');  //mssql\_query("SET character\_set\_results=utf-8",$dbc);  //mysqli\_query("SET NAMES 'UTF8'");  //mssqli\_select\_db("KN",$dbc);  //mssqli\_query("SET character\_set\_results=utf8",$dbc);  }    ?> |

* + - 1. Quang-cao.php

|  |
| --- |
| <div id="banner-quang-cao">  ggsdgfsd  </div> |

* + - 1. Sibar-2.php

|  |
| --- |
| <div id="banner-quang-cao">  ggsdgfsd  </div> |

* + - 1. Index.php

|  |
| --- |
| <?php include('includes/header.php')?>  <?php include('includes/sibar-2.php')?>  <div id="info">  Karaoke Nnice giúp bạn giảm căng thẳng  </div><!--end info-->  <?php include('includes/quang-cao.php')?>  <?php include('includes/footer.php')?> |

* + - 1. Style.css

|  |
| --- |
| /\*\*\*\*\* MAIN SETTINGS \*\*\*\*\*\*\*/  \* {margin: 0; padding: 0}  p, label, a {color: #333333; line-height: 1.5; font-size: 13px; font-family:'lucida grande',tahoma,verdana,arial,sans-serif； }  ul{list-style: none;}  a {color: #035389; text-decoration: none;}  a:hover {color: #649805; }  h2 a, h2{ margin: 0; font-size: 19px; font-family: museo, georgia, san-serif;}  /\*\*\*\*HEADER\*\*\*\*\*/  #wrap-page{width: 1000px;margin: auto;}  #login-logo{width: 1000px;margin-top: 20px;height:173px;background: url("../images/logo.gif") no-repeat left;}  #login{width:210px;float: right;border: 1px solid;border-radius: 5px;padding: 10px 10px;margin: -130px 15px;}  #logo h1{ text-indent: -99999999999999999px;width:290px;height:173px;}  #menu-search{width: 1000px;background:#00aeef;height:40px;}  #sibar-1{float: left;}  #sibar-1 li{float:left;padding: 10px 15px;}  #sibar-1 li a {display:block;color:black;}  #sibar-1 li:hover{border: 1px solid #6dcff6;background:#6dcff6;}  #sibar-1 li a:hover{color: #959595;}  #search{float:right;margin:5px 10px;}  #sibar-2{width:200px;float:left;}  /\*\*\*\*\*CONTENT\*\*\*\*\*\*\*\*\*\*\*/  #content-info{padding-top: 80px;}  #info{float:left;}  #info fieldset{width:750px;}  #info fieldset select{width: 140px;}  #info fieldset table{margin:auto;}  #info fieldset table td,tr,th{padding:10px 15px;}  #banner-quang-cao{float:right;width:200px;}  .warning{color:red;font-weight: bold;}  .sur{color:black;font-weight: bold;}  /\*\*\*\*\*FOOTER\*\*\*\*\*\*/  #footer{width:1000px;float: left;}  #footer hr{margin-top: 40px;margin-bottom: 40px;}  #footer p {margin:0px 0px 40px 360px;} |

* + - 1. Admin.php

|  |
| --- |
| <?php include('../includes/header.php')?>  <?php include('../includes/mysqli\_connect.php')?>  <?php include('../includes/fucntion.php')?>  <?php include('../includes/sibar-3.php')?>  <div id="info">  <?php  //mysql\_query("SET NAMES 'utf-8'");  // mysql\_query("SET NAMES UTF8");  if($\_SERVER['REQUEST\_METHOD'] == 'POST'){  if(!empty($\_POST['submit'])){  $submit=$\_POST['submit'];  }    if(!empty($\_POST['thanh-toan'])){  $thanh\_toan=$\_POST['thanh-toan'];  }    if(!empty($submit)){//neu nguoi dung nhan vao nut submit  if(isset($\_POST['product'])&&filter\_var($\_POST['product'],FILTER\_VALIDATE\_INT, array('min\_range'=>1))){  $product=$\_POST['product'];  }else{  echo "<p class='warning'>Chưa chọn sản phẩm<br/></p>";  }  if(!empty($\_POST['so\_luong'])){  if(filter\_var($\_POST['so\_luong'],FILTER\_VALIDATE\_INT, array('min\_range'=>1))){  $so\_luong=$\_POST['so\_luong'];  }else{  echo "<p class='warning'>Số lượng vui lòng nhập số</p>";  }  }else{  echo "<p class='warning'>Chưa có số lượng</p>";  }    }//ket thuc kiem tra submit co ton tai khong    }  $mes=" ";  echo "<p>{$mes}</p>";  if(!empty($thongbao)){  echo $thongbao;  }  ?>    <form action="" method="post">  <fieldset>  <legend> Lựa chọn dịch vụ </legend>  <table>  <caption><span class="ten\_phong">  <?php  if(isset($\_GET['id\_phong'])&& filter\_var($\_GET['id\_phong'],FILTER\_VALIDATE\_INT,array('min\_range'=>1))){  $q10="SELECT \* FROM tb\_phong WHERE id\_phong={$\_GET['id\_phong']}";  $r10=mysqli\_query($dbc,$q10);  confirm\_query($r10,$q10);  while($p=mysqli\_fetch\_array($r10,MYSQLI\_ASSOC)){  echo "{$p['name\_phong']}"." Lầu "."{$p['lau']}";  }  }  ?>  </span></caption>  <tr>  <td><label for="Ten-dich-vu">Tên dịch vụ: <span class="required">\*</span></label></td>  <td>  <select name="product" size="5" multiple="" tabindex="1">    <!--<option value="1">Dịch vụ phụ thu 1</option>  <option value="2">Dịch vụ phụ thu 2</option>  <option value="3">Dịch vụ phụ thu 3</option>  <option value="4">Dịch vụ phụ thu 4</option>  <option value="5">Dịch vụ phụ thu 5</option>  <option value="6">Dịch vụ phụ thu 6</option>-->  <?php    $q1="select \* FROM tb\_dich\_vu ORDER BY name\_dich\_vu";  $r1=mysqli\_query($dbc,$q1);  confirm\_query($r1,$q1);    while($dv=mysqli\_fetch\_array($r1,MYSQLI\_ASSOC)){  echo "  <option value={$dv['id\_dich\_vu']}>{$dv['name\_dich\_vu']}</option>  ";  }  ?>    </select>  </td>  </tr>  <tr>  <td><label for="so-luong">Số lượng: <span class="required">\*</span></label></td>  <td><input type="text" name="so\_luong" size="20" maxlength="3" value="" tabindex="2"/></td>  </tr>  <tr>  <td></td>  <td><input type="submit" name="submit" value="OK"/><input type="reset" name="reset" value="Xóa"/></td>  </tr>  </table>  </fieldset>  </form>  <fieldset>  <legend> Thông tin hóa đơn </legend>  <table border=1>  <thead>  <tr>  <th><a href="<?php getURL();?>&id=dv">Tên dịch vụ</a></th>  <th><a href="<?php getURL();?>&id=dg">Đơn giá</th>  <th><a href="<?php getURL();?>&id=sl">Số lượng</th>  <th><a href="<?php getURL();?>&id=tt">Tổng tiền</th>  <th>Edit</th>  <th>Delete</th>  </tr>  </thead>  <tbody>  <?php        $hien\_tai=date('Y-m-d H:i:s');  if(isset($\_GET['id\_phong'])&& filter\_var($\_GET['id\_phong'],FILTER\_VALIDATE\_INT,array('min\_range'=>1))){  $id\_phong=$\_GET['id\_phong'];    //select tb\_hoa\_don khi id\_phong =$id\_phong  $q2="SELECT \* FROM tb\_hoa\_don WHERE id\_phong=$id\_phong and trang\_thai LIKE 'no'";  $r2=mysqli\_query($dbc,$q2);  confirm\_query($r2,$q2);  //kiem tra san pham va so luong da duoc chon hay chua  if(!empty($product)&&!empty($so\_luong)){  while($hd=mysqli\_fetch\_array($r2,MYSQLI\_ASSOC)){  //echo "{$hd['id\_hoa\_don']}";  $q5="INSERT INTO tb\_cthd(id\_hoa\_don, id\_dich\_vu, so\_luong) VALUES ({$hd['id\_hoa\_don']},{$product},{$so\_luong})";  $r5=mysqli\_query($dbc,$q5);  confirm\_query($r5,$q5);  }  }else{  $thongbao="<br/>Vui lòng chọn sản phẩm và điền số lượng.";  }    if(!empty($thanh\_toan)){  while($hd=mysqli\_fetch\_array($r2,MYSQLI\_ASSOC)){  $q6="UPDATE tb\_phong as p, tb\_hoa\_don as hd SET p.trang\_thai='no', hd.trang\_thai='yes' where p.id\_phong={$id\_phong} and id\_hoa\_don={$hd['id\_hoa\_don']}";  $r6=mysqli\_query($dbc,$q6);  confirm\_query($r6,$q6);    }  $mes= "Phòng đã thanh toán";  }    //kiem tra hoa don co ton tai khong  if(mysqli\_num\_rows($r2)>0){    $q4="SELECT dv.don\_vi\_tinh as dvt,hd.id\_hoa\_don as mahd,dv.id\_dich\_vu as madv, dv.name\_dich\_vu as dvu, dv.don\_gia as dg, ct.so\_luong as sl,(dv.don\_gia \* ct.so\_luong) AS tong\_tien ";  $q4.="FROM tb\_hoa\_don AS hd ";  $q4.="JOIN tb\_cthd AS ct ";  $q4.="USING ( id\_hoa\_don ) ";  $q4.="JOIN tb\_dich\_vu AS dv ";  $q4.="USING ( id\_dich\_vu ) ";  $q4.="WHERE hd.id\_phong ={$id\_phong} ";  $q4.="AND hd.trang\_thai LIKE 'no' ";  $r=mysqli\_query($dbc,$q4);  confirm\_query($r,$q4);  $tong=0;  while($cthd=mysqli\_fetch\_array($r,MYSQLI\_ASSOC)){  $tong+="{$cthd['tong\_tien']}";    echo "  <tr>  <td>{$cthd['dvu']}</td>  <td>{$cthd['dg']}/{$cthd['dvt']}</td>  <td>{$cthd['sl']}</td>  <td>{$cthd['tong\_tien']}</td>  <td><a class='edit' href='";  echo "suacthd.php?iddv={$cthd["madv"]}&idhd={$cthd["mahd"]}'>Edit</a></td>  <td><a class='delete' href='";  echo "xoacthd.php?iddv={$cthd["madv"]}&idhd={$cthd["mahd"]}'>Delete</a></td>  </tr>  ";    }//end while loop  }else{    $q7="select \* FROM tb\_phong as p where p.trang\_thai='yes' ";  $r7=mysqli\_query($dbc,$q7);  confirm\_query($r7,$q7);  if(isset($r7)){  $q3="INSERT INTO tb\_hoa\_don(id\_user,id\_phong,ngay\_lap,trang\_thai) VALUES ('1','{$id\_phong}','{$hien\_tai}','no')";  $r3=mysqli\_query($dbc,$q3);  confirm\_query($r3,$q3);  }else{  $mes= "Phòng đã thanh toán";  }    }//ket thuc else kiem tra hoa don co ton tai khong    }  else{  $id\_phong="Không có phòng.";  $mes=$id\_phong;  }  ?>  <?php  /\*$now = getdate();  $currentTime = $now["hours"] . ":" . $now["minutes"] . ":" . $now["seconds"];  $currentDate = $now["mday"] . "." . $now["mon"] . "." . $now["year"];  $currentWeek = $now["wday"] . ".";  echo $now["hours"] .":" . $now["minutes"] . ":" . $now["seconds"];  date\_default\_timezone\_set('Asia/Ho\_Chi\_Minh');  echo date('Y-m-d H:i:s')."<br/>";  \*/    ?>  <tr>  <td></td>    <td colspan="5"><form action="" method="post"><input type="submit" name="thanh-toan" value="Thanh toán"/> <label for="tong"> Tổng: </label>  <input type="text" name="total" size="20" maxlength="3" value="<?php if(!empty($tong)){echo $tong;} ?>" tabindex="2" readonly="true"/></form></td>    </tr>  </tbody>  </table>  </fieldset>  </div><!--end info-->  <?php include('../includes/footer.php')?> |

* + - 1. Phong\_hoat\_dong.php

|  |
| --- |
| <?php include('../includes/header.php')?>  <?php include('../includes/mysqli\_connect.php')?>  <?php include('../includes/fucntion.php')?>  <?php include('../includes/sibar-3.php')?>  <div id="info">  <fieldset>  <legend> Phòng đang hoạt động</legend>  <table border=1 width=750>  <tr>  <th>Tầng trệt</th>  <th>Tầng 1</th>  </tr>  <tr>  <td>  <table border=1>  <tr>    <th>Tên phòng</th>    <th>Xem hóa đơn</th>  </tr>  <?php  if(isset($\_GET['id\_phong'])&& filter\_var($\_GET['id\_phong'],FILTER\_VALIDATE\_INT,array('min\_range'=>1))){  $id\_phong=$\_GET['id\_phong'];  $q="UPDATE tb\_phong SET trang\_thai='yes' where id\_phong=$id\_phong ";  $r=mysqli\_query($dbc,$q);  confirm\_query($r,$q);  }    $q1="select \* FROM tb\_phong as p where p.trang\_thai='yes' and lau = 0";  $r1=mysqli\_query($dbc,$q1);  confirm\_query($r1,$q1);    while($dv=mysqli\_fetch\_array($r1,MYSQLI\_ASSOC)){  echo "  <tr>    <td>{$dv['name\_phong']}</td>    <td><a href='admin.php?id\_phong={$dv['id\_phong']}'>Xem hóa đơn</a><br/></td>  </tr>    ";  }  ?>  </table>  </td>  <td>  <table border=1>  <tr>    <th>Tên phòng</th>    <th>Xem hóa đơn</th>  </tr>  <?php    $q3="select \* FROM tb\_phong as p where p.trang\_thai='yes' and lau = 1";  $r3=mysqli\_query($dbc,$q3);  confirm\_query($r3,$q3);    while($dv=mysqli\_fetch\_array($r3,MYSQLI\_ASSOC)){  echo "  <tr>    <td>{$dv['name\_phong']}</td>    <td><a href='admin.php?id\_phong={$dv['id\_phong']}'>Xem hóa đơn</a><br/></td>  </tr>    ";  }  ?>  </table>  </td>  </tr>  <tr>  <th>Tầng 2</th>  <th>Tầng 3</th>  </tr>  <tr>  <td>  <table border=1>  <tr>    <th>Tên phòng</th>    <th>Xem hóa đơn</th>  </tr>  <?php      $q5="select \* FROM tb\_phong as p where p.trang\_thai='yes' and lau = 2";  $r5=mysqli\_query($dbc,$q5);  confirm\_query($r5,$q5);    while($dv=mysqli\_fetch\_array($r5,MYSQLI\_ASSOC)){  echo "  <tr>    <td>{$dv['name\_phong']}</td>    <td><a href='admin.php?id\_phong={$dv['id\_phong']}'>Xem hóa đơn</a><br/></td>  </tr>    ";  }  ?>  </table>  </td>  <td>  <table border=1>  <tr>    <th>Tên phòng</th>    <th>Xem hóa đơn</th>  </tr>  <?php    $q11="select \* FROM tb\_phong as p where p.trang\_thai='yes' and lau = 3";  $r11=mysqli\_query($dbc,$q11);  confirm\_query($r11,$q11);    while($dv=mysqli\_fetch\_array($r11,MYSQLI\_ASSOC)){  echo "  <tr>    <td>{$dv['name\_phong']}</td>    <td><a href='admin.php?id\_phong={$dv['id\_phong']}'>Xem hóa đơn</a><br/></td>  </tr>    ";  }  ?>  </table>  </td>  </tr>  <tr>  <th>Tầng 4</th>  <th>Tầng 5</th>  </tr>  <tr>  <td>  <table border=1>  <tr>    <th>Tên phòng</th>    <th>Xem hóa đơn</th>  </tr>  <?php    $q12="select \* FROM tb\_phong as p where p.trang\_thai='yes' and lau = 4";  $r12=mysqli\_query($dbc,$q12);  confirm\_query($r12,$q12);    while($dv=mysqli\_fetch\_array($r12,MYSQLI\_ASSOC)){  echo "  <tr>    <td>{$dv['name\_phong']}</td>    <td><a href='admin.php?id\_phong={$dv['id\_phong']}'>Xem hóa đơn</a><br/></td>  </tr>    ";  }  ?>  </table>  </td>  <td>  <table border=1>  <tr>    <th>Tên phòng</th>    <th>Xem hóa đơn</th>  </tr>  <?php      $q13="select \* FROM tb\_phong as p where p.trang\_thai='yes' and lau = 5";  $r13=mysqli\_query($dbc,$q13);  confirm\_query($r13,$q13);    while($dv=mysqli\_fetch\_array($r13,MYSQLI\_ASSOC)){  echo "  <tr>    <td>{$dv['name\_phong']}</td>    <td><a href='admin.php?id\_phong={$dv['id\_phong']}'>Xem hóa đơn</a><br/></td>  </tr>    ";  }  ?>  </table>  </td>  </tr>  <tr>  <th>Tầng 6</th>    </tr>  <tr>  <td>  <table border=1>  <tr>    <th>Tên phòng</th>    <th>Xem hóa đơn</th>  </tr>  <?php    $q14="select \* FROM tb\_phong as p where p.trang\_thai='yes' and lau = 6";  $r14=mysqli\_query($dbc,$q14);  confirm\_query($r14,$q14);    while($dv=mysqli\_fetch\_array($r14,MYSQLI\_ASSOC)){  echo "  <tr>    <td>{$dv['name\_phong']}</td>    <td><a href='admin.php?id\_phong={$dv['id\_phong']}'>Xem hóa đơn</a><br/></td>  </tr>    ";  }  ?>  </table>  </td>  <td></td>  </tr>  </table>  </fieldset>      <?php          while($dv=mysqli\_fetch\_array($r1,MYSQLI\_ASSOC)){  echo "  ";  }  ?>  </div><!--end info-->  <?php include('../includes/footer.php')?> |

* + - 1. Phong\_trong.php

|  |
| --- |
| <?php include('../includes/header.php')?>  <?php include('../includes/mysqli\_connect.php')?>  <?php include('../includes/fucntion.php')?>  <?php include('../includes/sibar-3.php')?>  <div id="info">  <fieldset>  <legend> Phòng trống</legend>  <table border=1 width=750>  <tr>  <th>Tầng trệt</th>  <th>Tầng 1</th>  </tr>  <tr>  <td>  <table border=1>  <tr>    <th>Tên phòng</th>    <th>Xem hóa đơn</th>  </tr>  <?php      $q1="select \* FROM tb\_phong as p where p.trang\_thai='no' and lau = 0";  $r1=mysqli\_query($dbc,$q1);  confirm\_query($r1,$q1);    while($dv=mysqli\_fetch\_array($r1,MYSQLI\_ASSOC)){  echo "  <tr>    <td>{$dv['name\_phong']}</td>    <td><a href='phong\_hoat\_dong.php?id\_phong={$dv['id\_phong']}'>Hoạt động</a><br/></td>  </tr>    ";  }  ?>  </table>  </td>  <td>  <table border=1>  <tr>    <th>Tên phòng</th>    <th>Xem hóa đơn</th>  </tr>  <?php    $q3="select \* FROM tb\_phong as p where p.trang\_thai='no' and lau = 1";  $r3=mysqli\_query($dbc,$q3);  confirm\_query($r3,$q3);    while($dv=mysqli\_fetch\_array($r3,MYSQLI\_ASSOC)){  echo "  <tr>    <td>{$dv['name\_phong']}</td>    <td><a href='phong\_hoat\_dong.php?id\_phong={$dv['id\_phong']}'>Hoạt động</a><br/></td>  </tr>    ";  }  ?>  </table>  </td>  </tr>  <tr>  <th>Tầng 2</th>  <th>Tầng 3</th>  </tr>  <tr>  <td>  <table border=1>  <tr>    <th>Tên phòng</th>    <th>Xem hóa đơn</th>  </tr>  <?php      $q5="select \* FROM tb\_phong as p where p.trang\_thai='no' and lau = 2";  $r5=mysqli\_query($dbc,$q5);  confirm\_query($r5,$q5);    while($dv=mysqli\_fetch\_array($r5,MYSQLI\_ASSOC)){  echo "  <tr>    <td>{$dv['name\_phong']}</td>    <td><a href='phong\_hoat\_dong.php?id\_phong={$dv['id\_phong']}'>Hoạt động</a><br/></td>  </tr>    ";  }  ?>  </table>  </td>  <td>  <table border=1>  <tr>    <th>Tên phòng</th>    <th>Xem hóa đơn</th>  </tr>  <?php    $q11="select \* FROM tb\_phong as p where p.trang\_thai='no' and lau = 3";  $r11=mysqli\_query($dbc,$q11);  confirm\_query($r11,$q11);    while($dv=mysqli\_fetch\_array($r11,MYSQLI\_ASSOC)){  echo "  <tr>    <td>{$dv['name\_phong']}</td>    <td><a href='phong\_hoat\_dong.php?id\_phong={$dv['id\_phong']}'>Hoạt động</a><br/></td>  </tr>    ";  }  ?>  </table>  </td>  </tr>  <tr>  <th>Tầng 4</th>  <th>Tầng 5</th>  </tr>  <tr>  <td>  <table border=1>  <tr>    <th>Tên phòng</th>    <th>Xem hóa đơn</th>  </tr>  <?php    $q12="select \* FROM tb\_phong as p where p.trang\_thai='no' and lau = 4";  $r12=mysqli\_query($dbc,$q12);  confirm\_query($r12,$q12);    while($dv=mysqli\_fetch\_array($r12,MYSQLI\_ASSOC)){  echo "  <tr>    <td>{$dv['name\_phong']}</td>    <td><a href='phong\_hoat\_dong.php?id\_phong={$dv['id\_phong']}'>Hoạt động</a><br/></td>  </tr>    ";  }  ?>  </table>  </td>  <td>  <table border=1>  <tr>    <th>Tên phòng</th>    <th>Xem hóa đơn</th>  </tr>  <?php      $q13="select \* FROM tb\_phong as p where p.trang\_thai='no' and lau = 5";  $r13=mysqli\_query($dbc,$q13);  confirm\_query($r13,$q13);    while($dv=mysqli\_fetch\_array($r13,MYSQLI\_ASSOC)){  echo "  <tr>    <td>{$dv['name\_phong']}</td>    <td><a href='phong\_hoat\_dong.php?id\_phong={$dv['id\_phong']}'>Hoạt động</a><br/></td>  </tr>    ";  }  ?>  </table>  </td>  </tr>  <tr>  <th>Tầng 6</th>    </tr>  <tr>  <td>  <table border=1>  <tr>    <th>Tên phòng</th>    <th>Xem hóa đơn</th>  </tr>  <?php    $q14="select \* FROM tb\_phong as p where p.trang\_thai='no' and lau = 6";  $r14=mysqli\_query($dbc,$q14);  confirm\_query($r14,$q14);    while($dv=mysqli\_fetch\_array($r14,MYSQLI\_ASSOC)){  echo "  <tr>    <td>{$dv['name\_phong']}</td>    <td><a href='phong\_hoat\_dong.php?id\_phong={$dv['id\_phong']}'>Hoạt động</a><br/></td>  </tr>    ";  }  ?>  </table>  </td>  <td></td>  </tr>  </table>  </fieldset>  </div><!--end info-->  <?php include('../includes/footer.php')?> |

* + - 1. Suacthd.php

|  |
| --- |
| <?php include('../includes/header.php')?>  <?php include('../includes/mysqli\_connect.php')?>  <?php include('../includes/fucntion.php')?>  <?php include('../includes/sibar-3.php')?>  <div id="info">  <?php  if(isset($\_GET['iddv'])&&isset($\_GET['idhd'])){    $iddv=$\_GET['iddv'];  $idhd=$\_GET['idhd'];  $q1="SELECT \*FROM tb\_dich\_vu where id\_dich\_vu='{$iddv}'";  $r1=mysqli\_query($dbc,$q1);  confirm\_query($r1,$q1);  while($dv1=mysqli\_fetch\_array($r1,MYSQLI\_ASSOC)){  $namedv="{$dv1['name\_dich\_vu']}";    }  if($\_SERVER['REQUEST\_METHOD']=="POST"){  if(!empty($\_POST['tendv'])){  $tendv=$\_POST['tendv'];  //echo $tenphong;  }else{  echo "<p class='warning'>Vui lòng nhập tên dịch vụ.<br/></p>";  }  if(!empty($\_POST['gia'])){  if(filter\_var($\_POST['gia'],FILTER\_VALIDATE\_INT,array('min\_range'=>1))){  $gia=$\_POST['gia'];  //echo $trang\_thai;  }  else{  echo "<p class='warning'>Vui lòng nhập đơn giá bằng số.<br/></p>";  }    }else{  echo "<p class='warning'>Vui lòng nhập đơn giá.<br/></p>";  }  if(!empty($tendv)&&!empty($gia)){  $q="UPDATE tb\_cthd SET so\_luong='{$gia}' where id\_dich\_vu={$iddv} and id\_hoa\_don={$idhd}" ;  $r=mysqli\_query($dbc,$q);  confirm\_query($r,$q);  if(mysqli\_affected\_rows($dbc)==1){  //url\_stat("them\_dv.php");  //redirect\_to("");    echo "<p class='sur'>Sửa thành công.</p>";  }else{  echo "<p class='warning'>Sửa không được.</p>";  }  }    }  }    ?>  <form action="" method="POST">  <fieldset>  <legend>Form sửa chi tiết hóa đơn</legend>  <table>  <tr>  <td><label for="tendv">Tên dịch vụ: <span class="required">\*</span></label></td>  <td><input type="text" name="tendv" value="<?php if(!empty($namedv)) echo $namedv;?>" readonly="true"/></td>  </tr>  <tr>  <td><label for="gia">Số lượng<span class="required">\*</span></label></td>  <td><input type="text" name="gia" value=""/></td>  </tr>  <tr>  <td></td>  <td><input type="submit" value="Add" name="add"/><input type="reset" name="reset" value="Reset"/></td>  </tr>  </table>  </fieldset>  </form>    </div><!--end info-->  <?php include('../includes/footer.php')?> |

* + - 1. Suadv.php

|  |
| --- |
| <?php include('../includes/header.php')?>  <?php include('../includes/mysqli\_connect.php')?>  <?php include('../includes/fucntion.php')?>  <?php include('../includes/sibar-3.php')?>  <div id="info">  <?php  if(isset($\_GET['iddv'])){    $iddv=$\_GET['iddv'];  $q1="SELECT \*FROM tb\_dich\_vu where id\_dich\_vu='{$iddv}'";  $r1=mysqli\_query($dbc,$q1);  confirm\_query($r1,$q1);  while($dv1=mysqli\_fetch\_array($r1,MYSQLI\_ASSOC)){  $namedv="{$dv1['name\_dich\_vu']}";  $dongia="{$dv1['don\_gia']}";  }  if($\_SERVER['REQUEST\_METHOD']=="POST"){  if(!empty($\_POST['tendv'])){  $tendv=$\_POST['tendv'];  //echo $tenphong;  }else{  echo "<p class='warning'>Vui lòng nhập tên dịch vụ.<br/></p>";  }  if(isset($\_POST['dvt'])){  $dvt=$\_POST['dvt'];  //echo $tenphong;  }else{  echo "<p class='warning'>Vui lòng chọn đơn vị tính.<br/></p>";  }  if(!empty($\_POST['gia'])){  if(filter\_var($\_POST['gia'],FILTER\_VALIDATE\_INT,array('min\_range'=>1))){  $gia=$\_POST['gia'];  //echo $trang\_thai;  }  else{  echo "<p class='warning'>Vui lòng nhập đơn giá bằng số.<br/></p>";  }    }else{  echo "<p class='warning'>Vui lòng nhập đơn giá.<br/></p>";  }  if(!empty($tendv)&&!empty($gia)&&!empty($dvt)){  $q="UPDATE tb\_dich\_vu SET name\_dich\_vu='{$tendv}',don\_gia='{$gia}',don\_vi\_tinh='{$dvt}' where id\_dich\_vu={$iddv}";  $r=mysqli\_query($dbc,$q);  confirm\_query($r,$q);  if(mysqli\_affected\_rows($dbc)==1){  //url\_stat("them\_dv.php");  //header("Location : http://localhost:8080/Do an/admin/them\_dv.php");  //exit();  echo "<p class='sur'>Sửa thành công.</p>";  }else{  echo "<p class='warning'>Sửa không được.</p>";  }  }    }  }    ?>  <form action="" method="POST">  <fieldset>  <legend>Form sửa Dịch vụ</legend>  <table>  <tr>  <td><label for="tendv">Tên dịch vụ: <span class="required">\*</span></label></td>  <td><input type="text" name="tendv" value="<?php if(!empty($namedv)) echo $namedv;?>"/></td>  </tr>  <tr>  <td><label for="gia">Đơn giá<span class="required">\*</span></label></td>  <td><input type="text" name="gia" value="<?php if(!empty($dongia)) echo $dongia;?>"/></td>  </tr>  <tr>  <td><label for="dvt">Đơn vị tính<span class="required">\*</span></label></td>  <td>  <select name="dvt">  <option value="Lon">Lon</option>  <option value="Chai">Chai</option>  <option value="Két">Két</option>  <option value="Bì">Bì</option>  <option value="Kg">Kg</option>  <option value="Dĩa">Dĩa</option>  <option value="Suất">Suất</option>  <option value="Thùng">Thùng</option>    </select>  </td>  </tr>  <tr>  <td></td>  <td><input type="submit" value="Add" name="add"/><input type="reset" name="reset" value="Reset"/></td>  </tr>  </table>  </fieldset>  </form>    </div><!--end info-->  <?php include('../includes/footer.php')?> |

* + - 1. Suaphong.php

|  |
| --- |
| <?php include('../includes/header.php')?>  <?php include('../includes/mysqli\_connect.php')?>  <?php include('../includes/fucntion.php')?>  <?php include('../includes/sibar-3.php')?>  <div id="info">  <?php  if(isset($\_GET['idphong'])){    $iddv=$\_GET['idphong'];    $q1="SELECT \*FROM tb\_phong where id\_phong='{$iddv}'";  $r1=mysqli\_query($dbc,$q1);  confirm\_query($r1,$q1);  while($dv1=mysqli\_fetch\_array($r1,MYSQLI\_ASSOC)){  $namedv="{$dv1['name\_phong']}";    }  if($\_SERVER['REQUEST\_METHOD']=="POST"){  if(!empty($\_POST['tendv'])){  $tenphong=$\_POST['tendv'];  //echo $tenphong;  }else{  echo "<p class='warning'>Vui lòng nhập tên dịch vụ.<br/></p>";  }  if(isset($\_POST['lau'])){  $lau=$\_POST['lau'];  //echo $trang\_thai;  }else{  echo "<p class='warning'>Vui lòng chọn lầu.<br/></p>";  }  if(!empty($tenphong)&&!empty($lau)){  $q="UPDATE tb\_phong SET name\_phong='{$tenphong}',lau='{$lau}' where id\_phong={$iddv}" ;  $r=mysqli\_query($dbc,$q);  confirm\_query($r,$q);  if(mysqli\_affected\_rows($dbc)==1){  //url\_stat("them\_dv.php");  echo "<p class='sur'>Sửa thành công.</p>";  }else{  echo "<p class='warning'>Sửa không được.</p>";  }  }    }  }    ?>  <form action="" method="POST">  <fieldset>  <legend>Form sửa chi tiết hóa đơn</legend>  <table>  <tr>  <td><label for="tendv">Tên phòng: <span class="required">\*</span></label></td>  <td><input type="text" name="tendv" value="<?php if(!empty($namedv)) echo $namedv;?>" /></td>  </tr>  <tr>  <td><label for="lau">Lầu<span class="required">\*</span></label></td>  <td>  <select name="lau">  <option value="0">Tầng trệt</option>  <option value="1">Lầu 1</option>  <option value="2">Lầu 2</option>  <option value="3">Lầu 3</option>  <option value="4">Lầu4</option>  <option value="5">Lầu 5</option>  <option value="6">Lầu6</option>    </select>  </td>  </tr>  <tr>  <td></td>  <td><input type="submit" value="Add" name="add"/><input type="reset" name="reset" value="Reset"/></td>  </tr>  </table>  </fieldset>  </form>    </div><!--end info-->  <?php include('../includes/footer.php')?> |

* + - 1. Them\_dv.php

|  |
| --- |
| <?php include('../includes/header.php')?>  <?php include('../includes/mysqli\_connect.php')?>  <?php include('../includes/fucntion.php')?>  <?php include('../includes/sibar-3.php')?>  <div id="info">  <?php  if($\_SERVER['REQUEST\_METHOD']=="POST"){  if(!empty($\_POST['tendv'])){  $tendv=$\_POST['tendv'];  //echo $tenphong;  }else{  echo "<p class='warning'>Vui lòng nhập tên dịch vụ.<br/></p>";  }  if(isset($\_POST['dvt'])){  $dvt=$\_POST['dvt'];  //echo $tenphong;  }else{  echo "<p class='warning'>Vui lòng chọn đơn vị tính.<br/></p>";  }  if(!empty($\_POST['gia'])){  if(filter\_var($\_POST['gia'],FILTER\_VALIDATE\_INT,array('min\_range'=>1))){  $gia=$\_POST['gia'];  //echo $trang\_thai;  }  else{  echo "<p class='warning'>Vui lòng nhập đơn giá bằng số.<br/></p>";  }    }else{  echo "<p class='warning'>Vui lòng nhập đơn giá.<br/></p>";  }  if(!empty($tendv)&&!empty($gia)&&!empty($dvt)){  $q="INSERT INTO tb\_dich\_vu(id\_user, name\_dich\_vu,don\_gia,don\_vi\_tinh) VALUES ('1','{$tendv}','{$gia}','{$dvt}')";  $r=mysqli\_query($dbc,$q);  confirm\_query($r,$q);  if(mysqli\_affected\_rows($dbc)==1){  echo "<p class='sur'>Thêm thành công.</p>";  }else{  echo "<p class='warning'>Thêm không được.</p>";  }  }    }  ?>  <form action="" method="POST">  <fieldset>  <legend>Form Thêm Dịch vụ</legend>  <table>  <tr>  <td><label for="tendv">Tên dịch vụ: <span class="required">\*</span></label></td>  <td><input type="text" name="tendv" value=""/></td>  </tr>  <tr>  <td><label for="gia">Đơn giá<span class="required">\*</span></label></td>  <td><input type="text" name="gia" value=""/></td>  </tr>  <tr>  <td><label for="dvt">Đơn vị tính<span class="required">\*</span></label></td>  <td>  <select name="dvt">  <option value="Lon">Lon</option>  <option value="Chai">Chai</option>  <option value="Két">Két</option>  <option value="Bì">Bì</option>  <option value="Kg">Kg</option>  <option value="Dĩa">Dĩa</option>  <option value="Suất">Suất</option>  <option value="Thùng">Thùng</option>    </select>  </td>  </tr>  <tr>  <td></td>  <td><input type="submit" value="Add" name="add"/><input type="reset" name="reset" value="Reset"/></td>  </tr>  </table>  </fieldset>  </form>  <fieldset>  <legend>Thông tin các dịch vụ trong Karaoke</legend>  <table border=1>  <tr>  <th>Mã dịch vụ</th>  <th>Tên dịch vụ</th>  <th>Đơn giá</th>  <th>Đơn vị tính</th>  <th>Sửa</th>  <th>Xóa</th>  </tr>  <?php    $q1="select \* FROM tb\_dich\_vu ORDER BY name\_dich\_vu";  $r1=mysqli\_query($dbc,$q1);  confirm\_query($r1,$q1);    while($dv=mysqli\_fetch\_array($r1,MYSQLI\_ASSOC)){  echo "  <tr>  <td>{$dv['id\_dich\_vu']}</td>  <td>{$dv['name\_dich\_vu']}</td>  <td>{$dv['don\_gia']}</td>  <td>{$dv['don\_vi\_tinh']}</td>  <td><a href='";  echo "suadv.php?iddv={$dv['id\_dich\_vu']}'>Edit</a></th>  <td><a href='";  echo "xoadv.php?iddv={$dv['id\_dich\_vu']}'>Delete</a></th>  </tr>    ";  }  ?>  </table>  </fieldset>  </div><!--end info-->  <?php include('../includes/footer.php')?> |

* + - 1. Them\_phong.php

|  |
| --- |
| <?php include('../includes/header.php')?>  <?php include('../includes/mysqli\_connect.php')?>  <?php include('../includes/fucntion.php')?>  <?php include('../includes/sibar-3.php')?>  <div id="info">  <?php  if($\_SERVER['REQUEST\_METHOD']=="POST"){  if(!empty($\_POST['ten\_phong'])){  $tenphong=$\_POST['ten\_phong'];  //echo $tenphong;  }else{  echo "<p class='warning'>Vui lòng điền tên phòng.<br/></p>";  }  if(isset($\_POST['trang\_thai'])){  $trang\_thai=$\_POST['trang\_thai'];  //echo $trang\_thai;  }else{  echo "<p class='warning'>Vui lòng chọn trạng thái.<br/></p>";  }  if(isset($\_POST['lau'])){  $lau=$\_POST['lau'];  //echo $trang\_thai;  }else{  echo "<p class='warning'>Vui lòng chọn lầu.<br/></p>";  }  if(!empty($tenphong)&&!empty($trang\_thai)){  $q="INSERT INTO tb\_phong(name\_phong,lau,trang\_thai) VALUES ('{$tenphong}','{$lau}','{$trang\_thai}')";  $r=mysqli\_query($dbc,$q);  confirm\_query($r,$q);  if(mysqli\_affected\_rows($dbc)==1){  echo "<p class='sur'>Thêm thành công.</p>";  }else{  echo "<p class='warning'>Thêm không được.</p>";  }  }    }  ?>  <form action="" method="POST">  <fieldset>  <legend>Form Thêm Phòng</legend>  <table>  <tr>  <td><label for="ten-phong">Tên phòng<span class="required">\*</span></label></td>  <td><input type="text" name="ten\_phong" value=""/></td>  </tr>  <tr>  <td><label for="trang-thai">Trạng thái<span class="required">\*</span></label></td>  <td>  <select name="trang\_thai">  <option value="yes">Đang hoạt động</option>  <option value="no">Trống</option>  </select>  </td>  </tr>  <tr>  <td><label for="lau">Lầu<span class="required">\*</span></label></td>  <td>  <select name="lau">  <option value="0">Tầng trệt</option>  <option value="1">Lầu 1</option>  <option value="2">Lầu 2</option>  <option value="3">Lầu 3</option>  <option value="4">Lầu 4</option>  <option value="5">Lầu 5</option>  <option value="6">Lầu 6</option>    </select>  </td>  </tr>  <tr>  <td></td>  <td><input type="submit" value="Add" name="add"/><input type="reset" name="reset" value="Reset"/></td>  </tr>  </table>  </fieldset>  </form>  <fieldset>  <legend>Thông ti các phòng Karaoke</legend>  <table border=1>  <tr>  <th>Mã phòng</th>  <th>Tên phòng</th>  <th>Lầu</th>  <th>Trạng thái</th>  <th>Sửa</th>  <th>Xóa</th>  </tr>  <?php    $q1="select \* FROM tb\_phong as p ";  $r1=mysqli\_query($dbc,$q1);  confirm\_query($r1,$q1);    while($dv=mysqli\_fetch\_array($r1,MYSQLI\_ASSOC)){  echo "  <tr>  <td>{$dv['id\_phong']}</td>  <td>{$dv['name\_phong']}</td>  <td>{$dv['lau']}</td>  <td>{$dv['trang\_thai']}</td>  <td><a href='";  echo "suaphong.php?idphong={$dv['id\_phong']}'>Edit</a></th>  <td><a href='";  echo "xoaphong.php?idphong={$dv['id\_phong']}'>Delete</a></th>  </tr>    ";  }  ?>  </table>  </fieldset>  </div><!--end info-->  <?php include('../includes/footer.php')?> |

* + - 1. Xoacthd.php

|  |
| --- |
| <?php include('../includes/header.php')?>  <?php include('../includes/mysqli\_connect.php')?>  <?php include('../includes/fucntion.php')?>  <?php include('../includes/sibar-3.php')?>  <div id="info">  <?php  if(isset($\_GET['iddv'])&&isset($\_GET['idhd'])){  $iddv=$\_GET['iddv'];  $idhd=$\_GET['idhd'];  $q="DELETE FROM tb\_cthd WHERE id\_dich\_vu={$iddv} and id\_hoa\_don={$idhd}" ;  $r=mysqli\_query($dbc,$q);  confirm\_query($r,$q);  if(mysqli\_affected\_rows($dbc)==1){  //url\_stat("them\_dv.php");  echo "<p class='sur'>Xóa thành công.</p>";  }else{  echo "<p class='warning'>Xóa không được.</p>";  }  }    ?>      </div><!--end info-->  <?php include('../includes/footer.php')?> |

* + - 1. Xoadv.php

|  |
| --- |
| <?php include('../includes/header.php')?>  <?php include('../includes/mysqli\_connect.php')?>  <?php include('../includes/fucntion.php')?>  <?php include('../includes/sibar-3.php')?>  <div id="info">  <?php  if(isset($\_GET['iddv'])){  $iddv=$\_GET['iddv'];    $q="DELETE FROM tb\_dich\_vu WHERE id\_dich\_vu={$iddv}";  $r=mysqli\_query($dbc,$q);  confirm\_query($r,$q);  if(mysqli\_affected\_rows($dbc)==1){  //url\_stat("them\_dv.php");  echo "<p class='sur'>Xóa thành công.</p>";  }else{  echo "<p class='warning'>Xóa không được.</p>";  }  }    ?>      </div><!--end info-->  <?php include('../includes/footer.php')?> |

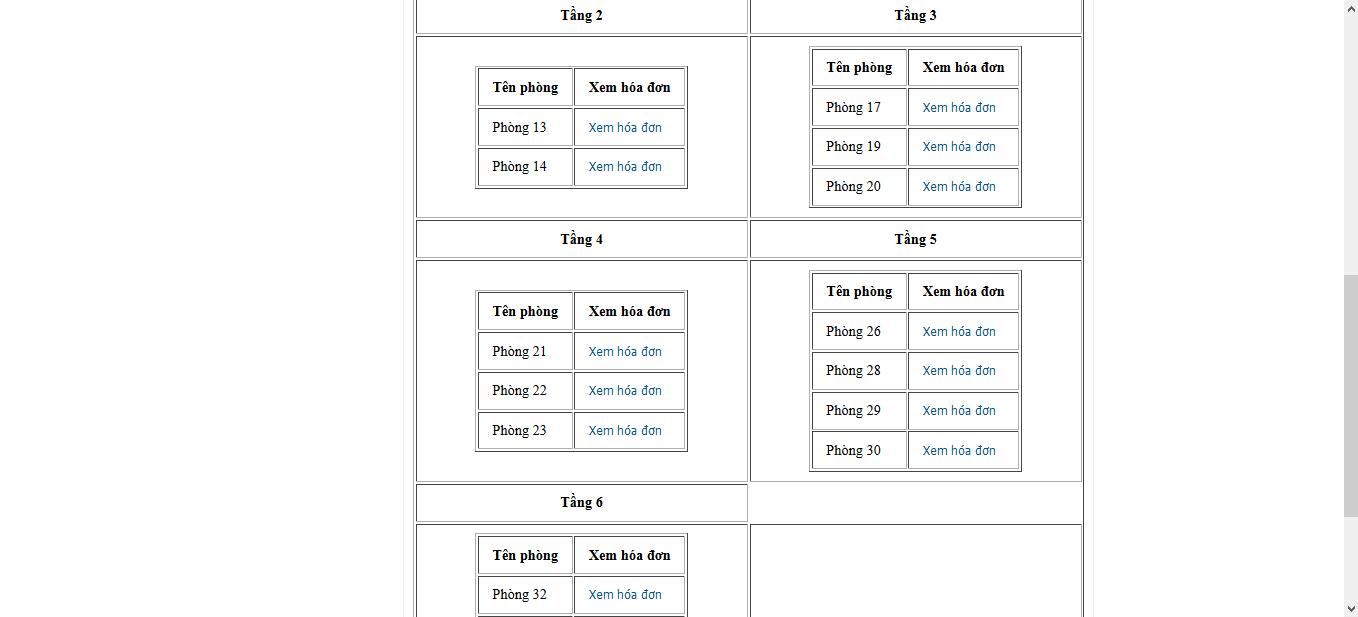
* + - 1. Xoaphong.php

|  |
| --- |
| <?php include('../includes/header.php')?>  <?php include('../includes/mysqli\_connect.php')?>  <?php include('../includes/fucntion.php')?>  <?php include('../includes/sibar-3.php')?>  <div id="info">  <?php  if(isset($\_GET['idphong'])){  $iddv=$\_GET['iddv'];    $q="DELETE FROM tb\_phong WHERE id\_phong={$idphong}" ;  $r=mysqli\_query($dbc,$q);  confirm\_query($r,$q);  if(mysqli\_affected\_rows($dbc)==1){  //url\_stat("them\_dv.php");  echo "<p class='sur'>Xóa thành công.</p>";  }else{  echo "<p class='warning'>Xóa không được.</p>";  }  }    ?>      </div><!--end info-->  <?php include('../includes/footer.php')?> |

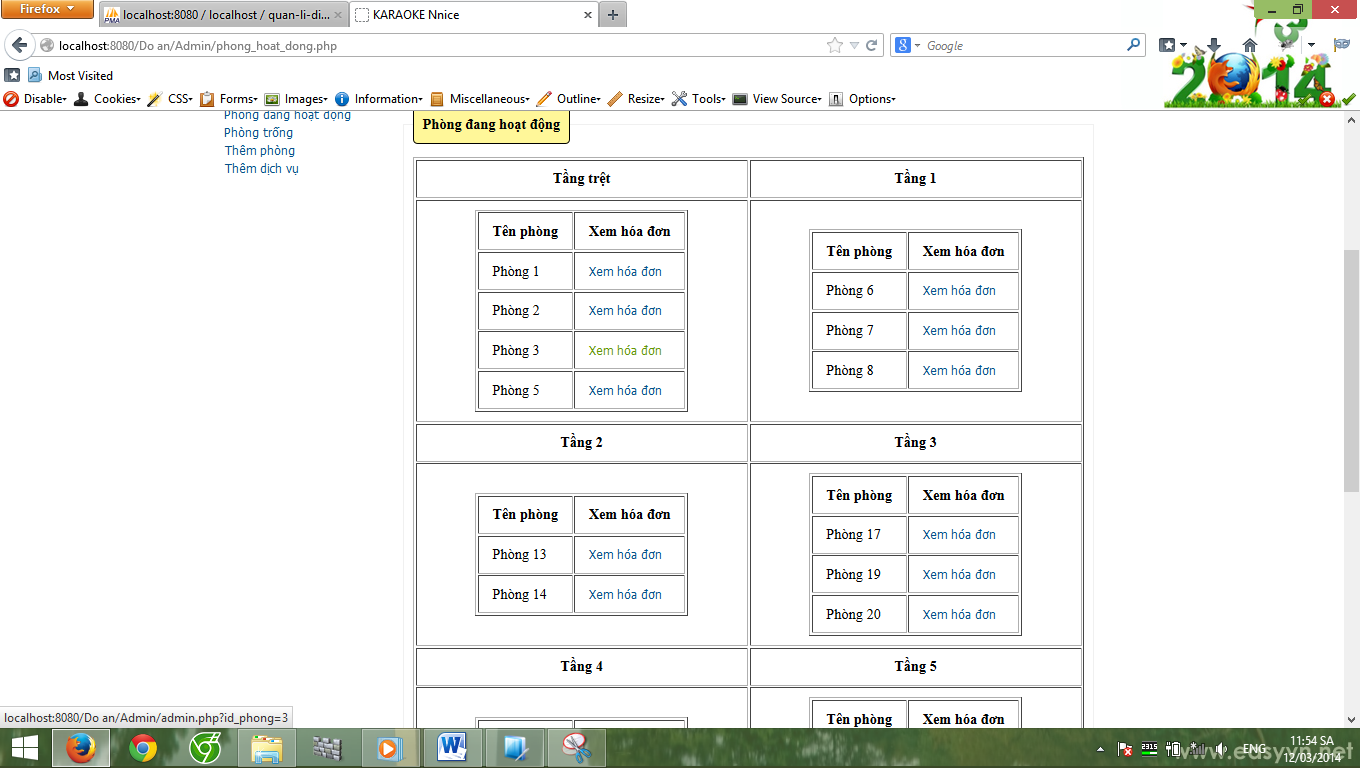
* + 1. Dome bài làm:

Giao diện phòng đang hoạt động

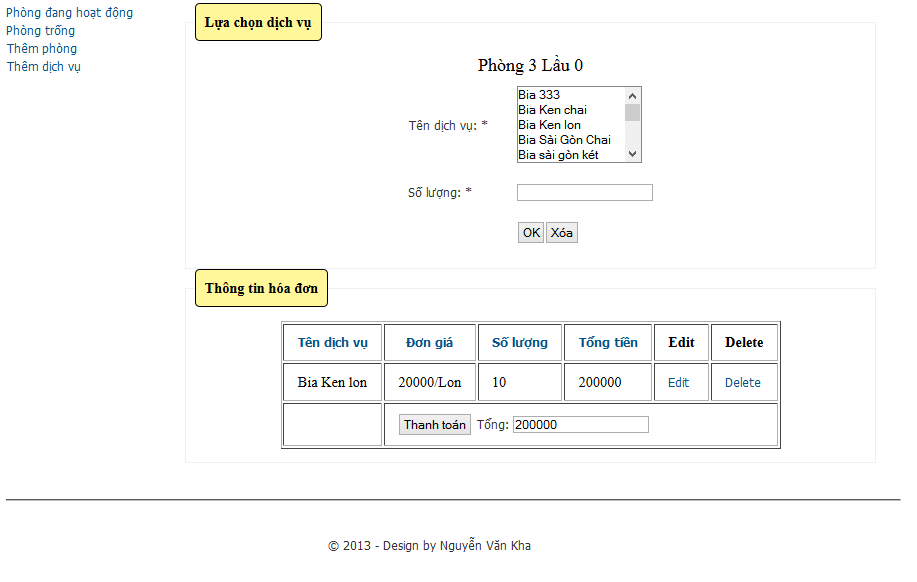






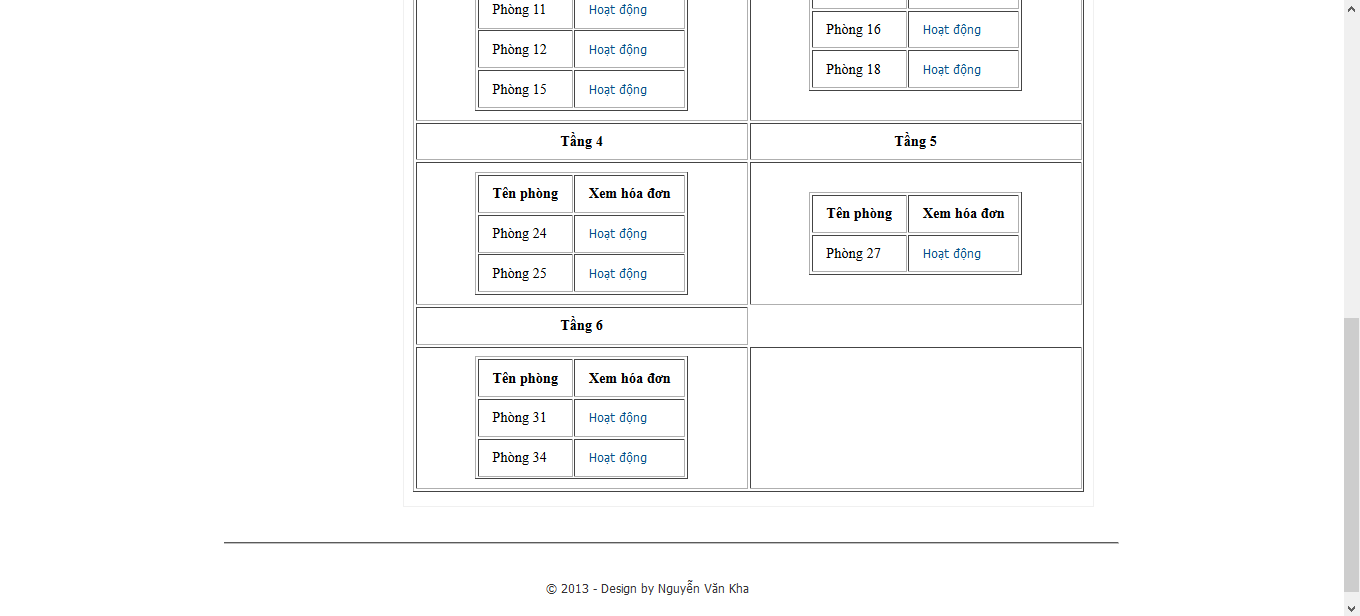


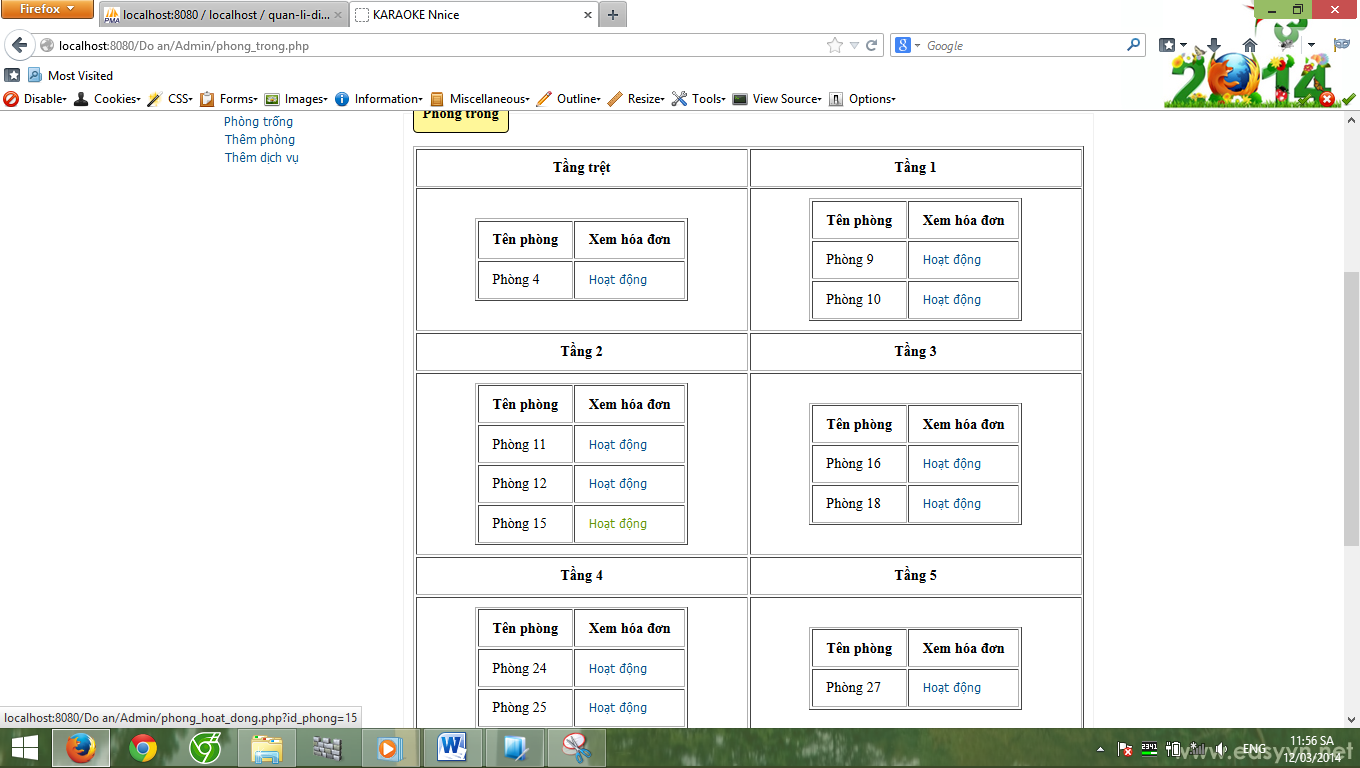
Giao diện chi tiết hóa đơn khi ta chọn vào 1 trong những phòng đang hoạt động:



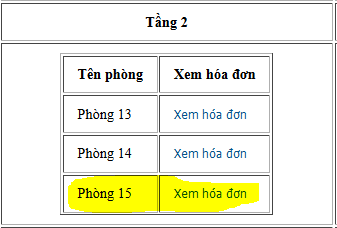
Giao diện phòng trống:



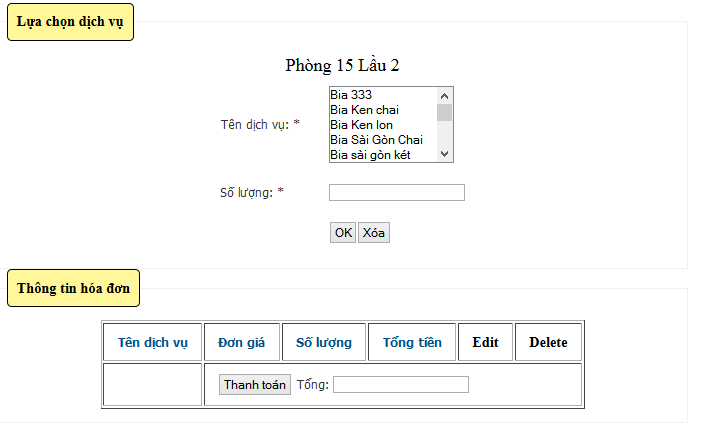




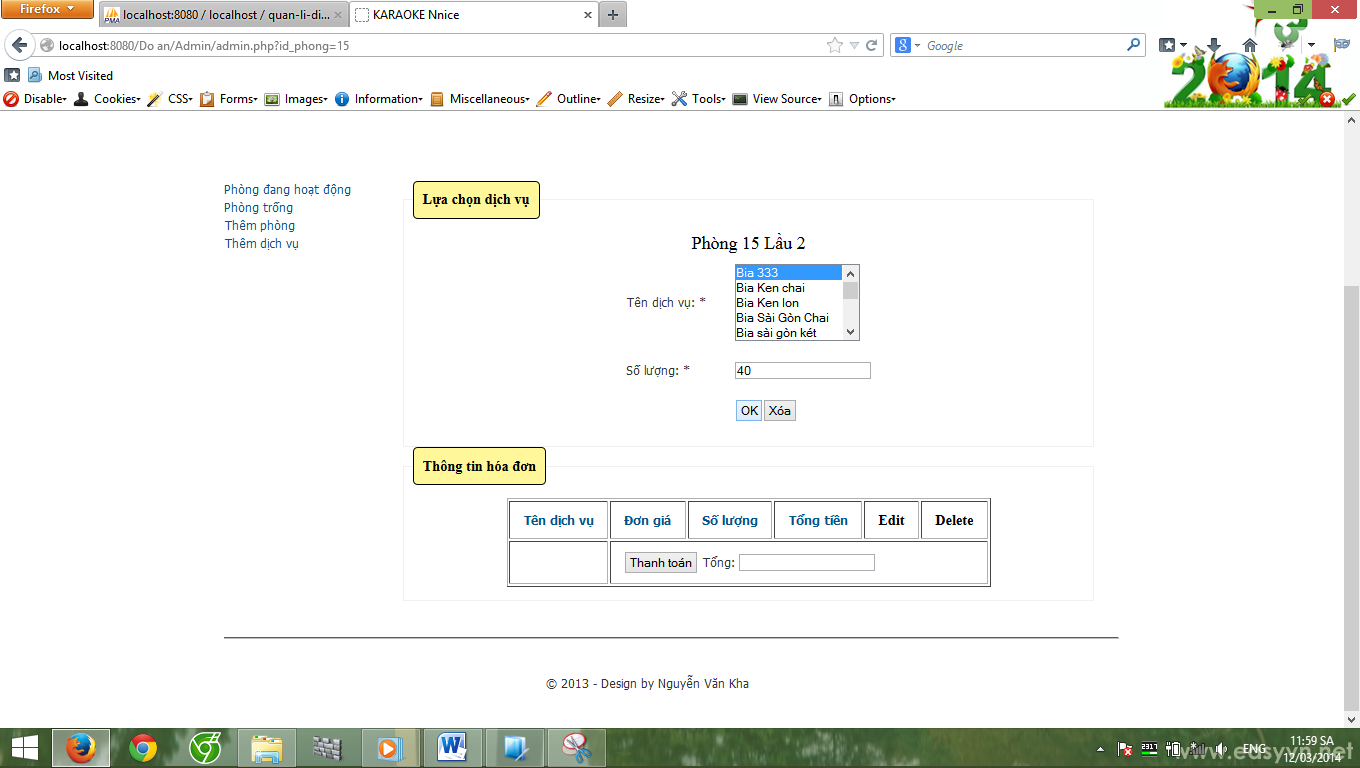
Khi chọn phòng trống thành hoạt động thi phòng đó sẽ chuyển sang phòng đang hoạt động

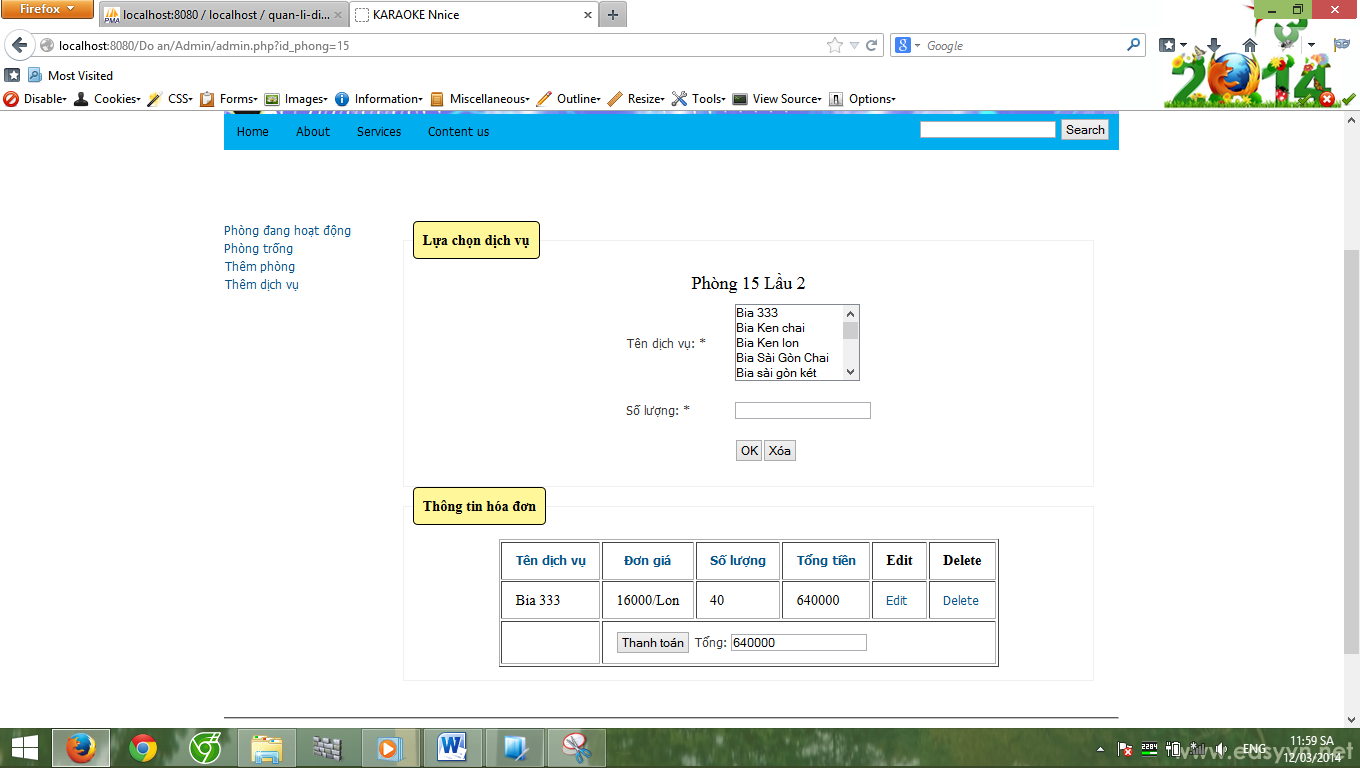


Khi phòng trống chuyển sang phòng hoạt động nó sẽ phát sinh ra 1 hóa đơn mới không có chi tiết hóa đơn nào trong đó

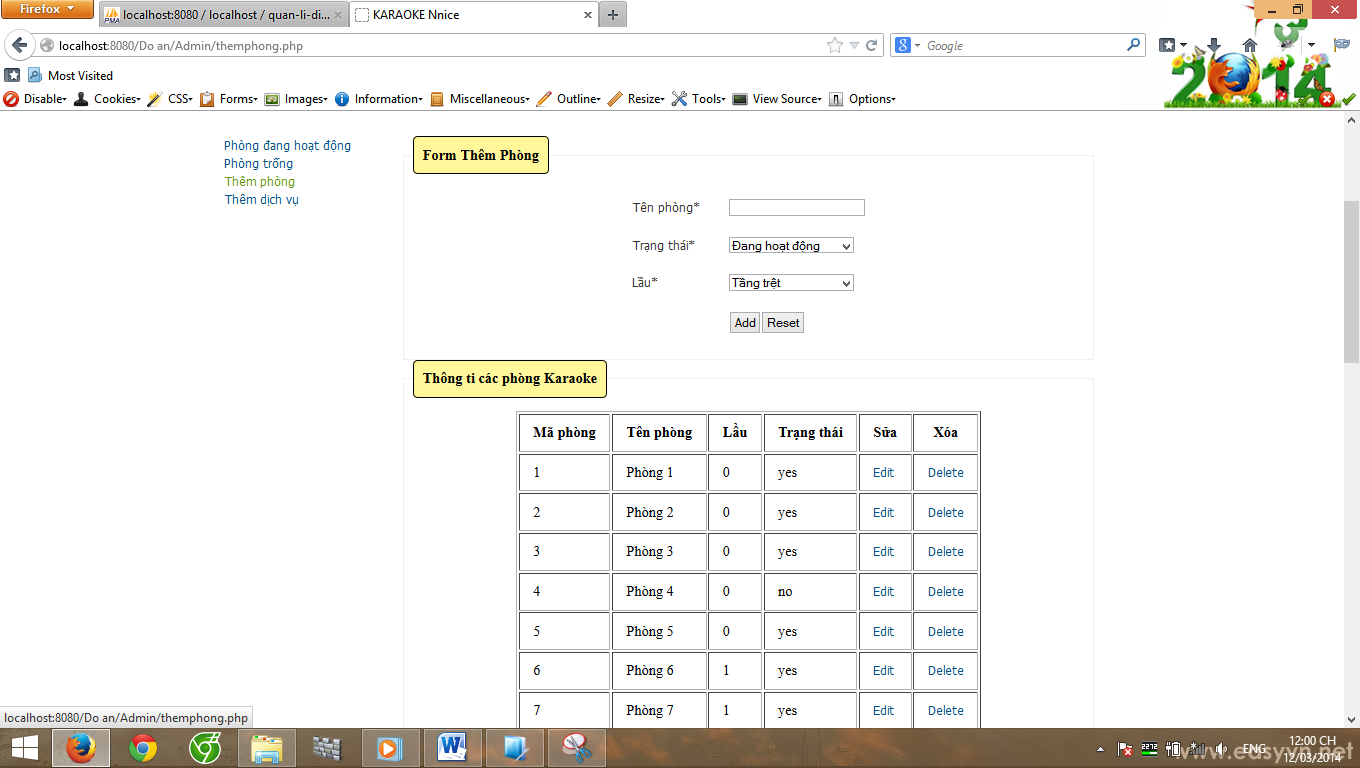


**Chọn** dịch vụ choh phòng

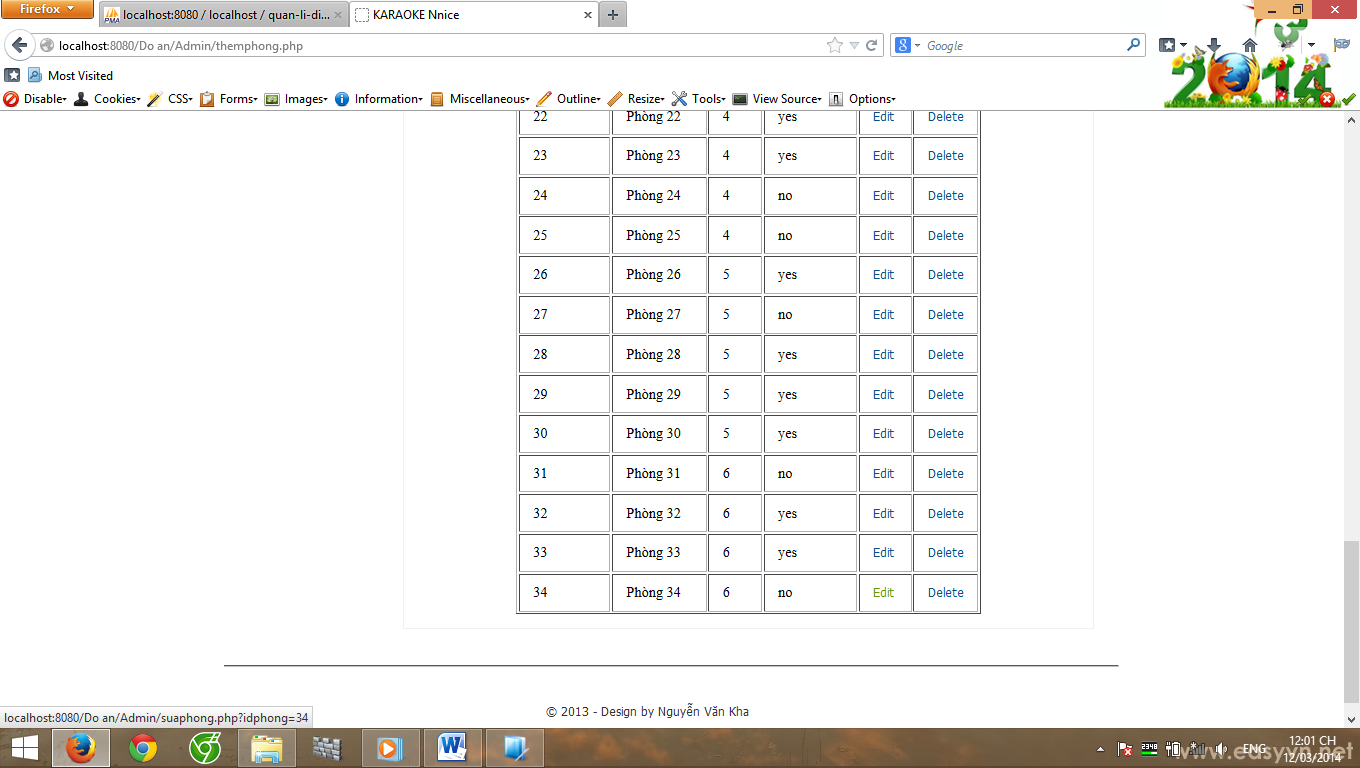


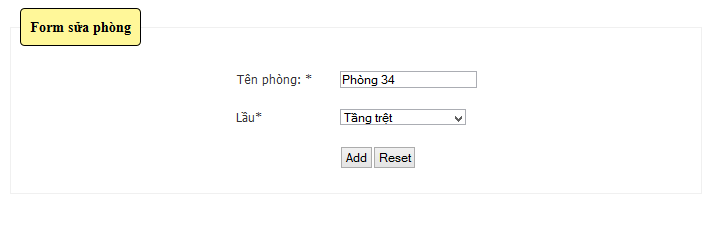


Giao diện them phòng

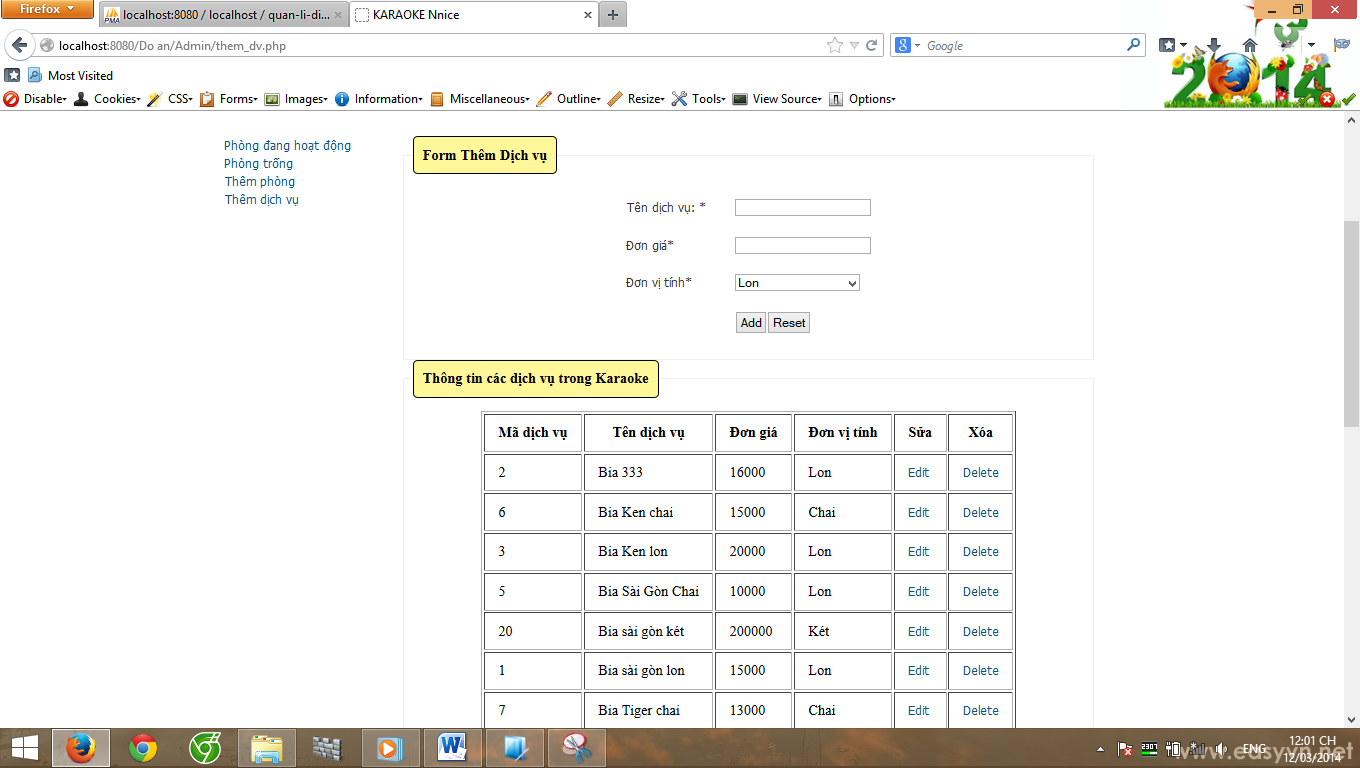


Chọn edit 1 phòng nào đó

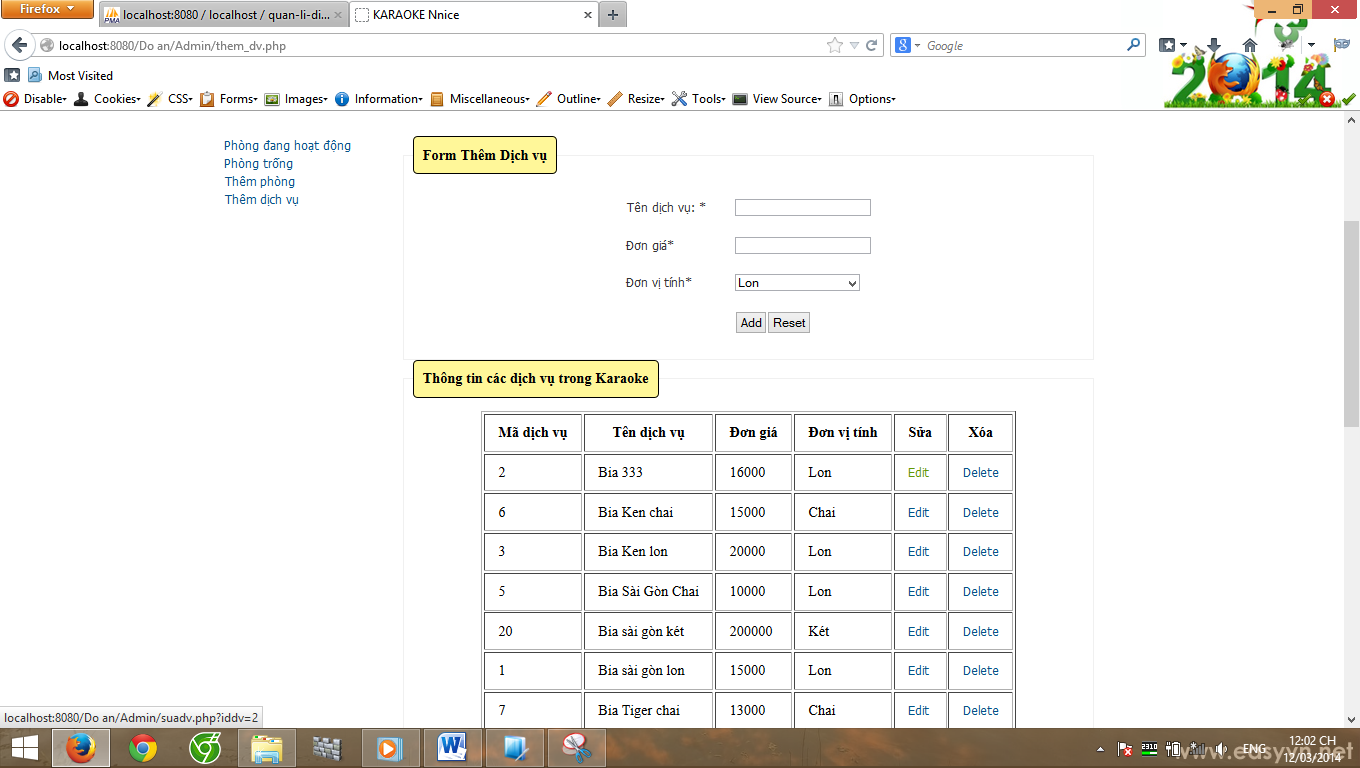




Giao diện thêm dịch vụ

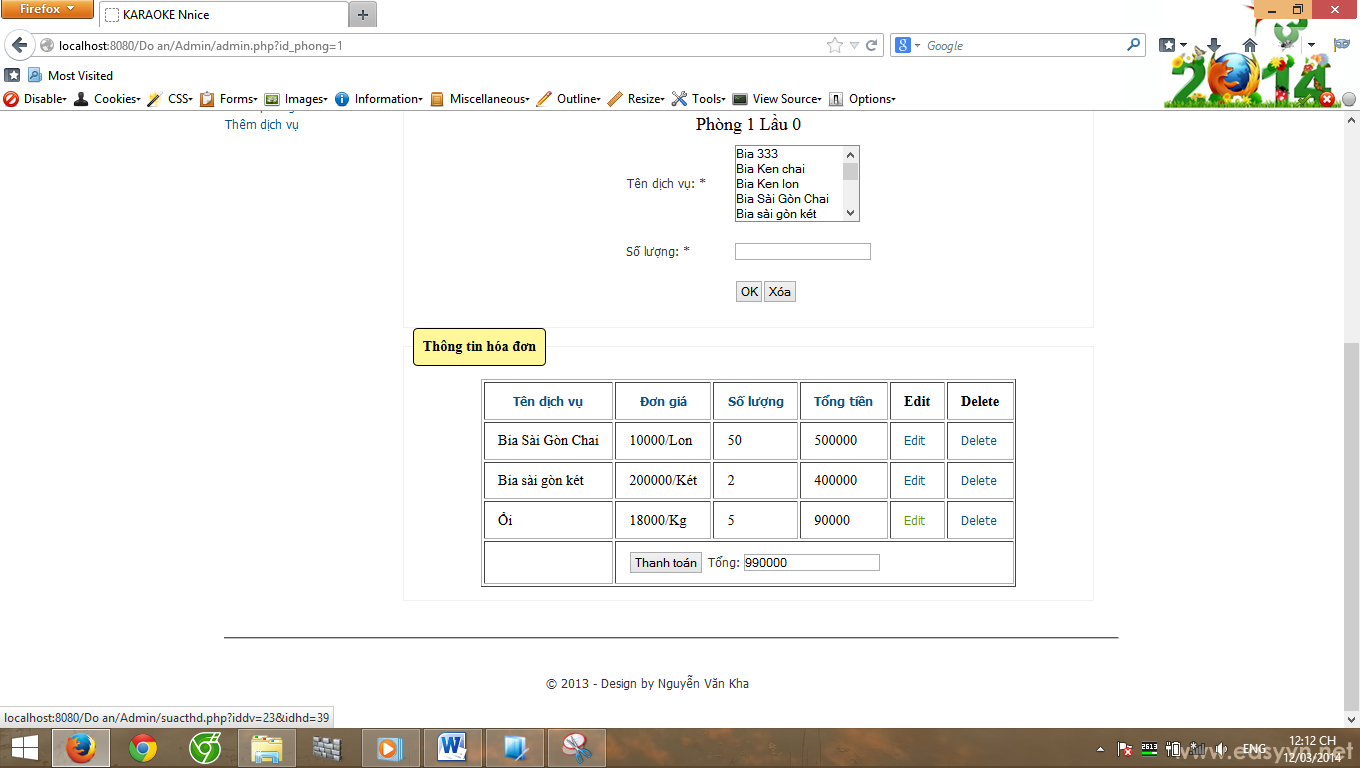


**Edit 1 dich vụ**





Sưa chi tiết hóa đơn của phòng đang chọn:





Khi thanh toán

