

PHP NÂNG CAO

SỬ DỤNG .HTACCESS

PHẦN 1:

1. Xem thử dữ liệu:

- Xem thử file `index.php?p=cat&idLT=1`
Xem thử file `index.php?p=cat&idLT=9`
- Xem thử file `index.php?p=detail&idTin=2`
Xem thử file `index.php?p=detail&idTin=50`

2. Bật mod rewrite của apache

- Mở folder `conf` trong folder cài đặt apache (thường là `c:/appserv/Apache2.2/conf`)
- Mở file `httpd.conf` bằng Dreamweaver
- Tìm đến dòng:

```
#LoadModule rewrite_module modules/mod_rewrite.so
```

- Bỏ dấu `#` ở đầu dòng rồi lưu và đóng file lại
- Restart Apache

3. Tạo file .htaccess

Dùng Dreamweaver, Tạo file và lưu với tên `.htaccess`

4. Bật chế độ chuyển địa chỉ

```
RewriteEngine on
```

5. Chuyển địa chỉ cho file đăng ký thành viên

- Trong file `.htaccess`, gõ lệnh:

```
RewriteRule users/register/$ dangky.php [L]  
RewriteRule users/changepass/$ doipass.php [L]  
RewriteRule users/sendpass/$ goipass.php [L]  
RewriteRule users/logout/$ thoat.php [L]
```

- Test

6. Chuyển địa chỉ cho các trang hiển thị loại tin

- Trong file `.htaccess`, gõ lệnh:

```
RewriteRule news/thethao/$ index.php?p=cat&idLT=1 [L]
```

```
RewriteRule news/songdep/$ index.php?p=cat&idLT=12 [L]  
RewriteRule news/thohay/$ index.php?p=cat&idLT=28 [L]
```

b. Test

7. File home.php

a. Sau mở đầu của tag head, thêm lệnh sau:

```
<base href="http://<?=$_SERVER['SERVER_NAME'];?>/php06/" />
```

b. Test

PHẦN 2:

- Chỉnh địa chỉ các tin theo kiểu: <http://localhost/php06/details/50/>
- Chỉnh địa chỉ các loại tin theo kiểu: <http://localhost/php06/cat/9/>

1. **File .htaccess:** Gõ lệnh sau vào file .htaccess

```
RewriteRule details/(.*)/$ index.php?p=details&idTin=$1
```

Test

2. **File tintrongloai.php:** Chỉnh địa chỉ cho liên kết Xem chi tiết như sau:

```
details/<?=$row_tin['idTin'];?>/
```

3. **File tinxemnhieu.php, tinnoibat.php, tinmoi.php, ketquatimkiem.php**

Tương tự file tintrongloai.php

4. **File .htaccess:** Gõ lệnh sau vào file .htaccess

```
RewriteRule cat/(.*)/$ index.php?p=cat&idLT=$1
```

Test

5. **File menu.php:** Chỉnh địa chỉ cho liên kết của tên các loại tin như sau:

```
cat/<?=$row_loaitin['idLT'];?>/
```

Test

PHẦN 3:

- **Chỉnh địa chỉ các tin theo kiểu:**

<http://localhost/php06/chitiet/Hoang-Hon-Tren-Song-Me-Kong/>
<http://localhost/PHP03/chitiet/Mot-Goc-Dat-Phuong-Nam/>

- **Chỉnh địa chỉ các loại tin theo kiểu:**

<http://localhost/php06/listtin/The-Thao/>
<http://localhost/php06/listtin/Du-Lich/>

1. **File .htaccess:** thêm lệnh sau, rồi lưu lại:

```
RewriteRule chitiet/(.*)/$ index.php?p=details&TieuDe_KhongDau=$1 [L]
```

2. **File chitiectin.php**

- a. Trong vị trí tiếp nhận tham số , chỉnh idTin thành TieuDe_KhongDau

```
<?php
    $TieuDe_KhongDau = $_GET['TieuDe_KhongDau'];
    if (get_magic_quotes_gpc()==false)
        $TieuDe_KhongDau = mysql_real_escape_string($TieuDe_KhongDau);
?>
```

- b. Trong câu lệnh SQL lấy dữ liệu, chỉnh điều kiện lọc tin.idTin=\$idTin thành

```
tin.TieuDe_KhongDau = '{$TieuDe_KhongDau}'
```

- c. Test

<http://localhost/PHP06/chitiet/Hoang-Hon-Tren-Song-Me-Kong/>
<http://localhost/PHP06/chitiet/Mot-Goc-Dat-Phuong-Nam/>

- d. Mở classTin, bổ sung thêm cột TieuDe_KhongDau cho các hàm lấy tin

3. **Chỉnh các liên kết đến 1 tin**

Mở các file tinxemnhieu.php, tinnoibat.php, tinmoi.php, tintrongloai.php rồi chỉnh địa chỉ các liên kết sang cấu trúc mới

Chú ý :

- Khi thêm tin và chỉnh tin, nhớ phát sinh TieuDe_KhongDau (dùng hàm changTitle)
- Bỏ dấu ? khỏi TieuDe_KhongDau

```
$TieuDe_KhongDau = $t->changTitle($TieuDe);
$TieuDe_KhongDau = str_replace("?","", $TieuDe_KhongDau);
$sql="INSERT INTO tin ...
```

4. **File .htaccess:** thêm lệnh sau, rồi lưu lại:

```
RewriteRule listtin/(.*)/$ index.php?p=cat&Ten_KhongDau=$1 [L]
```

5. **Điều chỉnh file tintrongloai.php**

- Trong vị trí tiếp nhận tham số , chỉnh idLT thành Ten_KhongDau

```
<?php
    $Ten_KhongDau = $_GET['Ten_KhongDau'];
    if (get_magic_quotes_gpc()==false)
```

```
$Ten_KhongDau = mysql_real_escape_string($Ten_KhongDau);  
$idLT = $t->LayidLT($Ten_KhongDau);  
?>
```

- Vào classTin.php định nghĩa hàm LayidLT

```
function LayidLT($Ten_KhongDau){  
    $sql="select idLT from loaitin where Ten_KhongDau='$Ten_KhongDau'";  
    $kq = mysql_query($sql,$this->conn) or die (mysql_error());  
    $row_kq = mysql_fetch_assoc($kq);  
    $idLT= $row_kq['idLT'];  
    return $idLT;  
}
```

- Test

<http://localhost/PHP06/listtin/The-Thao/>
<http://localhost/PHP06/listtin/Du-Lich/>

6. Chỉnh file menu.php

- Mở file menu.php
- Chỉnh địa chỉ cho liên kết của tên các loại tin như sau:

```
listtin/<?php echo $row_loaitin["Ten_KhongDau"]?>/
```

- Test

7. Chỉnh classDB.php

- Trong hàm LoaiTinTrongTheLoai, bổ sung field Ten_KhongDau vào câu lệnh select
- Test

8. Đổi tên mở rộng php

```
RewriteRule ^(.*)\.tco$ $1.php
```

Áp dụng để request các file formdangnhap.php, ketquadangnhap.php, ketquabinhchon.php

REALLY SIMPLE SYNDICATION – RSS

1. Tạo folder RSS

2. Tạo file RSS/tinmoi.php

```
<?php
$IdLT = $_GET['idLT']; settype($IdLT, "int");
require_once "../classTin.php";
$st= new tin;
$loaitin = $st->ChiTietLoaiTin($IdLT);
$row_loaitin = mysql_fetch_assoc($loaitin);
$stin = $st->TinTrongLoai($IdLT,1,10);
?>
<?php header("Content-type: text/xml"); ?>
<?php echo "<?xml version='1.0' encoding='utf-8' ?>
        <?xml-stylesheet type='text/xsl' href='format.xsl'?>";
?>
<rss version="2.0">
<channel>
<title><?=$row_loaitin['Ten']?></title>
<description><?=$row_loaitin['Ten']?> - No Name</description>
<link>
http://<?=\$\_SERVER\['SERVER\_NAME'\]?>/PHP06/RSS/tinmoi.php?idLT=<?=\$IdLT?>
</link>
<copyright>NO Name</copyright>
<generator>No Name News </generator>
<pubDate><?=gmdate("D, d m Y H:i:s")?> GMT</pubDate>
<lastBuildDate><?=gmdate("D, d m Y H:i:s")?> GMT</lastBuildDate>
<?php while ($row_tin=mysql_fetch_assoc($stin)) { ?>
<item>
<title> <?=$row_tin['TieuDe'] ?> </title>
<description> <?=$row_tin['TomTat'] ?> </description>
<link>
http://<?=\$\_SERVER\['SERVER\_NAME'\]?>/php06/home.php?p=details&idTin=<?=\$row\_tin\['idTin'\]?>
</link>
<pubDate><?=gmdate('D, d m Y H:i:s',strtotime($row_tin['Ngay']))?> GMT</pubDate>
</item>
<?php } //while ?>
</channel>
</rss>
```

Test (nhớ Sửa dấu & thành &)

3. Tạo file RSS/rss.php

- Lấy danh sách các loại tin được hiện
- Lập để hiện tên các loại tin ra trang web
- Liên kết tên các loại tin với file tin mới ở trên

CÁC GIẢI PHÁP THỰC HIỆN WEBSITE ĐA NGÔN NGỮ

1. Interface

Bao gồm chữ, hình, hình nền, javascript, css

a. Giải pháp 1:

Với mỗi trang web, tạo file riêng cho từng ngôn ngữ

Ví dụ:

- file index.php: Tạo **index_en.php**, **index_vi.php**, **index_fr.php**
- file tintrongloai.php: Tạo **tintrongloai_en.php**, **tintrongloai_vi.php**, **tintrongloai_fr.php**
- ...

Khuyết điểm: ?

Khó phát triển thêm ngôn ngữ, chỉnh sửa rất cực (vì phải chỉnh trên mọi ngôn ngữ đang dùng)

Ưu điểm: ?

các trang web chạy nhanh vì không phải qua bước xử lý ngôn ngữ, kỹ thuật không phức tạp

Thích hợp khi nào?

Cần tốc độ nhanh, hoặc site không có nhu cầu thêm ngôn ngữ sau này

Thực hiện:

tạo mỗi folder cho 1 ngôn ngữ, mỗi folder có 1 tập các file giống tên nhau, chỉ khác ngôn ngữ thể hiện

b. Giải pháp 2:

Với mỗi trang web, chỉ cần làm 1 interface. Các chữ, địa chỉ các hình, file css, javascript sẽ lấy từ file XML hoặc file config cho ngôn ngữ đó → chỉ cần 1 trang web và nhiều file config, mỗi file cho 1 ngôn ngữ

Ví dụ:

- Tạo file **config_en.php**, **config_vi.php**, **config_fr.php**
- Các trang web trong website chỉ cần include file config thích hợp vào để dùng

Khuyết điểm: ?

trang web chạy chậm vì phải qua bước xử lý ngôn ngữ, trang web không trực quan trong quá trình thiết kế

Ưu điểm: ?

dễ phát triển thêm nhiều ngôn ngữ khác

Thích hợp khi nào?

site có nhu cầu bổ sung thêm nhiều ngôn ngữ sau này

Thực hiện:

- + Tạo các trang web bình thường, index.php, tintrongloai.php, chitiettin.php
- + Tạo file config cho ngôn ngữ đầu tiên, với các biến chứa giá trị của từng dòng text, địa chỉ hình, địa chỉ file css
- + Đưa giá trị từng giá trị trong file config vào các trang web
- + Chép file config đầu tiên ra, và chỉnh sửa giá trị cho các biến theo ngôn ngữ mới

2. Database

Các table có thể dùng chung: users, groups, config.php.

Các table khác chứa dữ liệu liên quan đến ngôn ngữ thì bạn phải trả lời câu hỏi sau đây: Mỗi một đơn vị dữ liệu (một tin, một sản phẩm...) cần lưu để hiển thị nó trên nhiều ngôn ngữ hay mỗi một đơn vị dữ liệu chỉ hiển trong một ngôn ngữ nào đó mà thôi? . Ví dụ:

- Một tin cần hiển thị trên nhiều ngôn ngữ hay mỗi tin chỉ một ngôn ngữ
- Mỗi sản phẩm nhiều ngôn ngữ hay mỗi sản phẩm 1 ngôn ngữ

Theo câu trả lời mà chọn giải pháp

- **Giải pháp 1:** mỗi ngôn ngữ 1 database. Giải pháp này không tận dụng được những table chung, mỗi 1 đối tượng dữ liệu sẽ lưu dữ liệu trong 1 ngôn ngữ nào đó. Tuy nhiên code dễ, vì tên các table, tên các field, tên các recordset hoàn toàn giống nhau, bạn có thể tận dụng code (chỉ viết 1 lần, chỉ khác connection)

- **Giải pháp 2:** Một database nhưng sẽ tạo nhiều table ứng với mỗi ngôn ngữ. Ví dụ:

Để lưu tin, bạn tạo các bảng: tin_vi, tin_en, tin_fr

Để lưu loại tin, bạn tạo các bảng: loaitin_vi, loaitin_en, loaitin_fr

Để lưu thể loại, bạn tạo các bảng: theloai_vi, theloai_en, theloai_fr

Giải pháp này tận dụng được những table chung. Mỗi một đối tượng dữ liệu sẽ lưu dữ liệu trong 1 ngôn ngữ nào đó. Code sửa lại ít vì tên các field là giống nhau, chỉ khác tên table.

- **Giải pháp 3:** Sử dụng một table để lưu dữ liệu cho nhiều ngôn ngữ, nhưng thêm nhiều field, mỗi bộ field ứng với một ngôn ngữ. Ví dụ :

table tin có các field: idTin, *TieuDe_VI*, *TieuDe_EN*, *TieuDe_FR*, *TomTat_VI*, *TomTat_EN*, *TomTat_FR*, *Content_VI*, *Content_EN*, *Content_FR* ...

table loaitin có các field idLT, *Ten_VI*, *Ten_EN*, *Ten_FR*, ...

Giải pháp này chỉ 1 database, số table không nhiều, nhưng khó phát triển khi có thêm ngôn ngữ mới. Câu lệnh SQL cho từng recordset sẽ viết theo từng ngôn ngữ → phức tạp. Nhưng có đặc điểm rất hay là một đối tượng dữ liệu cụ thể sẽ được lưu trong nhiều ngôn ngữ khác nhau.

Tham khảo: <http://www.kymdan.com>

- **Giải pháp 4:** Sử dụng một table để lưu dữ liệu cho nhiều ngôn ngữ, nhưng chỉ thêm 1 field lang để nhận biết từng dòng dữ liệu thuộc ngôn ngữ nào. Ví dụ:

idTin	TieuDe	lang
280	City traders asked to register for grand sale prog...	en
3	Chuyện ở xứ Đền thiêng Đồng Cổ	vi
75	Cơ hội lịch sử của bóng đá VN	vi
89	Cái đầu vàng của Công Vinh	vi
238	Cranes Get Into Murky Water	en
252	Colliers Int'l to market Vietnam's tallest buildin...	en

3. Format data

Là định dạng số, tiền tệ, ngày tháng, phone theo ngôn ngữ.

Trong mỗi trang PHP, ở nơi hiển thị ngày tháng, số, tiền, phone, cần định dạng tùy theo ngôn ngữ mà user chọn

4. Tổ chức folder và các file

Cần có sự tổ chức hợp lý về lưu trang web php, hình ảnh, file js, css cho từng ngôn ngữ

5. Lưu biến để biết được user đang chọn ngôn ngữ nào

Ngôn ngữ mà user chọn cần phải lưu vào Session và Cookie , để khi họ request bạn biết ngôn ngữ nào cần phải hiện.

Ví dụ: biến `$_SESSION['lang']` và `$_COOKIE['lang']` để lưu language mà user chọn.

Khi user thay đổi ngôn ngữ, thì thay đổi giá trị của biến này

Biến cookie có ưu điểm là lưu giữ lâu, khuyết điểm là user có thể xóa nó. Biến session có ưu điểm là nó hoàn toàn trong tầm kiểm soát của bạn, khuyết điểm là chỉ lưu trong 1 khoảng thời gian ngắn

6. Xác định ngôn ngữ để đưa về cho user

Ở đầu của mỗi trang PHP: kiểm tra biến này để quyết định :

```
<?php
$lang="vi";
$lang_arr=array("vi","en","fr");
if (isset($_GET['lang'])){
    if (in_array($_GET['lang'],$lang_arr)==true) $lang=$_GET['lang'];
}
elseif (isset($_COOKIE['lang'])){
    if (in_array($_COOKIE['lang'],$lang_arr)==true) $lang=$_COOKIE['lang'];
}
elseif (isset($_SESSION['lang'])){
    if (in_array($_SESSION['lang'],$lang_arr)==true) $lang=$_SESSION['lang'];
}
$_SESSION['lang']=$lang;
setcookie('lang',$lang,time()+60*60*24*30);
?>
```

Khi đã xác định được biến \$lang, thì

- + Hoặc đưa về cho user file php có ngôn ngữ tương ứng (nếu mỗi file 1 ngôn ngữ)
- + Hoặc đọc file config hoặc XML tương ứng với ngôn ngữ

7. Xử lý khi user đổi ngôn ngữ

Có nhiều cách xử lý

Giao diện:

Cách 1: Hiện hình các lá cờ ứng với mỗi ngôn ngữ

Cách 2: Hiện một list/menu, chứa tên các ngôn ngữ

Code

Cách 1: link

- Link từng hình đến 1 trang web nào đó, có thể là trang hiện hành hoặc trang chủ, đồng thời truyền tham số tương ứng với ngôn ngữ. Ví dụ `index.php?lang=vn` hoặc `index.php?lang=en ...`
- Nếu dùng List/menu thì mỗi khi user chọn một ngôn ngữ, link đến trang chủ hoặc trang hiện hành với ngôn ngữ thích hợp

Cách 2: sử dụng Ajax

- Tạo file SetLang.php để thay đổi biến \$_SESSION['lang'], \$_COOKIE['lang']

Tạo hàm ajax để gọi setlang.php khi user chọn ngôn ngữ

8. Định dạng CSS cho từng ngôn ngữ

Tạo 1 file CSS với những thuộc tính định dạng chung

Tạo từng file Css riêng cho từng ngôn ngữ:

```
$cssFile="vietnamese.css";  
if ($lang == 'en') $cssFile = 'english.css';  
if ($lang == 'fr') $cssFile = 'france.css';
```

Hoặc cũng có thể lưu biến \$cssFile trong file config của mỗi ngôn ngữ

Liên kết file CSS vào trang:

```
<link rel="stylesheet" type="text/css" href="general.css" />  
<link rel="stylesheet" type="text/css" href="<?php echo $cssFile; ?>" />
```

CÁC KỸ THUẬT TĂNG TỐC WEBSITE

1. Giảm bớt các request

a. Đơn giản thiết kế

- Sử dụng số lượng vừa phải hình ảnh, flash hay script trên trang web.
- Nếu có thể thì gom các file javascript, css lại để giảm request

b. Sử dụng CSS Sprites

- Là cách thức để gom nhiều ảnh lại với nhau thành 1.
- Nên gom các ảnh nhỏ thành 1 cột dọc, để xác định tọa độ cho dễ. Có hai website rất hay và đơn giản giúp bạn gom các ảnh và tính tọa độ các ảnh:

<http://www.csssprites.com/> hoặc

<http://spritegen.website-performance.org/>

c. Sử dụng Image Maps

- Gom các ảnh nhỏ thành 1 ảnh lớn, rồi tạo các hotspot để liên kết tới các trang web thích hợp.

2. Sử dụng Content Delivery Network(CDN)

- Đặt website ở các vùng địa lý mà phần lớn người dùng website của bạn ở đó.
- Sử dụng các CDN nổi tiếng, đang hoạt động. Ví dụ Google đã lưu trữ những framework rất nổi tiếng như [jQuery](#), [prototype](#), [MooTools](#), [Dojo](#), [script.aculo.us](#) trên hệ thống server của nó. Khi dùng JQuery, bạn chỉ việc lấy về từ Google:

```
<script src=http://ajax.googleapis.com/ajax/libs/jquery/1.2.6/jquery.min.js  
type="text/javascript"> </script>
```

Hoặc

```
<script src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.2/jquery.min.js"  
type="text/javascript"> </script>
```

3. NÉN HTML

- Cách 1: Thêm dòng sau vào file *.htaccess*:

```
php_flag zlib.output_compression On  
php_flag zlib.output_compression_level 5
```

- Cách 2: Thêm lệnh `ob_start('ob_gzhandler');` ở đầu trang

4. CACHE ON CLIENT

Dùng 2 chỉ thị http header: **Expires** và **Cache-Control**

Chỉ thị Expire báo cho browser hoặc proxy biết khi nào thì trang sẽ quá hạn trên server. Để sử dụng Expire, ngay sau chỉ thị Content-type, bạn dùng Expire như sau:

```
$offset = 3600 * 24; // calc the string in GMT not localtime and add the offset
$expire = "Expires: " . gmdate("D, d M Y H:i:s", time() + $offset) . " GMT";
header($expire);
```

Chỉ thị Cache-Control trong chuẩn HTTP 1.1. Cách dùng như sau:

```
$expire = "Cache-Control: max-age=3600, must-revalidate";
header($expire);
```

Trong đó:

- max-age=**Số giây** – là số giây trang web được cache;
- must-revalidate – bắt browser phải tuân theo những chỉ thị về trang web gọi từ server, chứ không được bỏ qua
- no-cache – bắt proxy và browser phải validate tài liệu trước khi dùng tài liệu đã cache

5. Cache dùng htaccess

Sử dụng htaccess là một giải pháp để tăng tốc độ hiển thị trang web cho user. Bạn cần phải bật mod_headers lên nếu muốn dùng

a. Cache hình, flash một khoảng thời gian dài

```
# cache images và flash 30 ngày
<FilesMatch ".(flv|gif|jpg|jpeg|png|ico|swf)$">
Header set Cache-Control "max-age=2592000"
</FilesMatch>
```

Coi cache của Firefox:

Trong thanh địa chỉ, gõ about:cache

b. Cache file text, css, javascript

```
# cache text, css, và javascript files 1 tuần
<FilesMatch ".(js|css|pdf|txt)$">
Header set Cache-Control "max-age=604800"
</FilesMatch>
```

c. Cache file htm, html

```
# cache html và htm files một ngày
<FilesMatch ".(html|htm)$">
Header set Cache-Control "max-age=43200"
</FilesMatch>
```

d. Không cache các file script

```
# không cach scripts files  
<FilesMatch "\.(pl|php|cgi|spl|scgi|fcgi)$">  
Header unset Cache-Control  
</FilesMatch>
```

6. Tối ưu các MySQL Query

Tối ưu các câu lệnh SQL cũng là yếu tố quan trọng không được bỏ qua khi bạn muốn trang web chạy nhanh hơn. Sau đây là những điều cần chú ý:

- Bớt join các bảng sẽ tốt hơn
- Index hợp lý các field
- Có thể chấp nhận dữ liệu dư thừa để tăng tốc độ xử lý
- Tuyệt đối không dùng SELECT *, chỉ select những field cần dùng.

7. PHP Cache

a. Opcode caching

Opcode caching là cách đơn giản và rất hiệu quả để tăng tốc. Opcode là code của các trang PHP đã được dịch xong và lưu trong share memory để dùng lại mà không phải thông dịch từng lệnh trong file khi có request đến từ user. Có nhiều giải pháp bạn có thể dùng:

- APC: <http://www.php.net/manual/en/ref.apc.php>
- XCache: <http://xcache.lighttpd.net/>
- eAccelerator: <http://www.eaccelerator.net/>
- Zend Platform: <http://www.zend.com/en/products/platform/>

b. File caching

Nội dung file được lưu ở 1 nơi nào đó sau khi kết xuất lần đầu tiên. Với những request đến sau, chỉ đọc file kết quả rồi trả về ngay lập tức

Có thể cache một phần của trang vào file cache.

Khi dùng cách cache này, nếu trong trang có những dữ liệu cần phải thay đổi (không muốn cache) như hitcounter, user online thì bạn dùng chúng bằng ajax

Chú ý: không được dùng trong các trang đổi pass, trang có hiện username ...

c. Memory cache

- Những dữ liệu của riêng từng user, thay vì phải chạy vào database lấy, bạn có thể lưu chúng vào session để dùng ngay như username, họ tên, email, giới tính....
- Những dữ liệu kết xuất lần đầu tiên, thay vì lưu trong file cache, bạn có thể lưu trong 1 table nào đó. Để những request sau, bạn chỉ vào table cache lấy trả về ngay mà không làm gì nữa.
- Bạn có thể lưu cả trang hoặc một phần của trang vào table cache.

8. Lab : Cache file

- a. Tạo trang **tin.php** để hiện danh sách mọi tin. Trình bày thể bào tùy bạn.
- b. Tạo trang CacheFile_Begin.php

```
<?php
$cachedir = '../cache/'; // folder lưu các file cache
$cachetime = 600; // Thời gian (Số giây) lưu files cache
$cacheext = 'cache'; // phần mở rộng của file được cache
$page = 'http://'. $_SERVER['HTTP_HOST'].$_SERVER['REQUEST_URI'];
$pathfile = $cachedir . md5($page) . '.' . $cacheext;
$file_created = (file_exists($pathfile)) ? filemtime($pathfile):0;
//clearstatcache();
if (time() - $cachetime < $file_created) {
    ob_start('ob_gzhandler');
    readfile($pathfile); //đọc 1 file và echo nội dung ra buffer
    ob_end_flush();
    exit();
}
ob_start('ob_gzhandler');
?>
```

- c. Tạo trang CacheFile_End.php

```
<?php
$cachedir = '../cache/';
$cacheext = 'cache';
$page = 'http://'. $_SERVER['HTTP_HOST'].$_SERVER['REQUEST_URI'];
$pathfile = $cachedir . md5($page) . '.' . $cacheext;

$f = fopen($pathfile, 'w');
fwrite($f, ob_get_contents());
```

```
fclose($fp);  
  
ob_end_flush();  
  
?>
```

d. Tạo trang php trống, lưu với tên tin_cacheFile.php (để test cache)

- Ở đầu trang : <?php include "CacheFile_Begin.php";?>
- Ở cuối trang: <?php include "CacheFile_End.php";?>
- Trong tag body : <?php include "tin.php";?>

e. Test

f. Tính thời gian:

- Trong file CacheFile_Begin.php, ở dòng đầu, thêm lệnh:

```
$bd = (float) array_sum(explode(' ',microtime()));
```

- Trong file CacheFile_Begin.php, sau lệnh readfile, thêm 2 lệnh:

```
$kt = (float) array_sum(explode(' ',microtime()));  
echo "Xử lý trong cache: ". sprintf("%.4f", ($kt-$bd))." giây";
```

- Trong file CacheFile_End.php, ở dòng đầu, thêm lệnh:

```
$kt = (float) array_sum(explode(' ', microtime()));  
echo "Lần đầu: ". sprintf("%.4f", ($kt-$bd))." giây<br>";
```

- Test

9. Lab : Cache DB

a. Trong database, tạo 1 table như sau:

```
CREATE TABLE `cache` (  
  `name` varchar(20) NOT NULL,  
  `content` mediumtext NOT NULL,  
  `timecache` varchar(20) NOT NULL,  
  PRIMARY KEY (`name`)  
) ENGINE=MyISAM DEFAULT CHARSET=utf8;
```

- b. Tạo trang tin1.php để hiển danh sách mọi tin. Trình bày thế nào tùy bạn.
- c. Tạo trang cachingDB.php

Định nghĩa hàm sau

```
<?php
function cachingDB( $dataname,$filename,$cachetime){

    $rs = mysql_query("      SELECT content,timecache FROM cache
                           WHERE name='$dataname'");

    $row = mysql_fetch_row($rs);

    $data = $row[0];

    $thoidiem = (int)$row[1];

    $baygio = time();

    $cachnhau = $baygio-$thoidiem;

    if ((strlen($data)<=0) || ($cachnhau>=$cachetime)) {

        ob_start();

        require $filename;

        $data = ob_get_clean();

        $data_es = mysql_real_escape_string($data);

        $sql= "INSERT INTO cache SET

Name = '$dataname', content = '$data_es', timecache=$baygio

ON DUPLICATE KEY UPDATE timecache = $baygio";

        mysql_query($sql) or die(mysql_error());

    }

    return $data;

}

?>
```

- d. Tạo trang php trống, lưu với tên tin_cacheDB.php (để test cache)
 - Ở đầu trang: <?php require_once("cachingDB.php");?>
 - Trong body:

```
<?php  
  
$cachetime=5;  
  
$dataname="listtin";  
  
$filename="listtin.php";  
  
echo cachingDB($dataname,$filename,$cachetime);  
  
?>
```

e. Test