Session và cookie lưu ở đâu. Các thiết lập thời gian expire cho cookie

Mothod của giao thức http

Phương thức get và post

GET request có thể được cached, bookmark và lưu trong lịch sử của trình duyệt.

GET request bị giới hạn về chiều dài, do chiều dài của URL là có hạn.

GET request không nên dùng với dữ liệu quan trọng, chỉ dùng để nhận dữ liệu.

POST không thể, cached, bookmark hay lưu trong lịch sử trình duyệt.

POST không bị giới hạn về độ dài.

HEAD:; giống như GET nhưng chỉ gửi về HTTP header.

PUT: tải lên một mô tả về URI định trước.

DELETE: xóa một tài nguyên định trước.

OPTIONS: trả về phương thức HTTP mà server hỗ trợ.

CONNECT: chuyển kết nối của HTTP request thành một kết nối HTTP tunnel.

1- PHP có mấy cách khai báo ?. Những cách nào được xem là chính thống và không ảnh hưởng khi các phiên bản update sau này ?.  
– Có 3 cách là <? ? > và <?php ?> <% %>…cách 2 là chính thống

2- Hằng trong PHP khác gì so với biến ?. Nếu 1 hằng được định nghĩa 2 lần, thì liệu có bị lỗi không ?. Cho ví dụ minh họa. Hằng có thể nội suy như biến hay không ?.  
– Khác nhau : hằng không thay đổi trong suốt chương trình, biến thì có  
– Nếu định nghĩa hăng 2 lần thì sẽ có thông báo hằng đã tồn tại, chương trình vấn chạy với hằng được định nghĩa đầu tiên  
– Không hiểu nội suy là cm gì luôn

3- Có bao nhiêu biến môi trường ?. $\_REQUEST có thể nhận được giá trị từ form hay không ?.  
– Có 4 bến môi trường là : $\_GET, $\_POST, $\_REQUEST,$\_SERVER  
– $\_REQUEST nhận được, dùng nó trong trường hợp không biết dữ liệu nên là post hay get

4- Phân biệt sự khác nhau giữa $\_POST và $\_GET.  
– $\_GET : gửi dl lên server thông qua URL, thông tin hiển thị lên url , kích thước url giới hạn  
– $\_POST : gửi dl lên server dưới dạng ẩn, không hiển thị param lên url, dữ liệu không giới hạn.  
– Get thực thi nhanh hơn post vì cơ chế  
+ get : quăng chuỗi string lên URL, server tách chuỗi lấy đc tham số  
+ Post : các tham số được đóng gói vào 1 file tạm, sau đó trình duyệt gửi file tạm đó lên server và server lưu lại file tạm đó sau đó mới phân tích  
– Post chậm hơn get

5- Mảng tuần tự là gì ?. Khác gì với bất tuần tự ?. Để duyệt mảng ta dùng vòng lặp nào ?. Tại sao vòng lặp for, while, do….while lại không ưu tiên sư dụng trong duyệt mảng ?.  
– Mảng tuần tự là mảng mà các khóa được đánh theo số, bắt đầu từ 0  
– Mảng bất tuần tự là các key của nó k phải là số mà có thể là string.  
Để duyệt mảng tuần tự dùng vòng lặp for  
Duyệt mang bất tuần tự dùng vòng lặp foreach

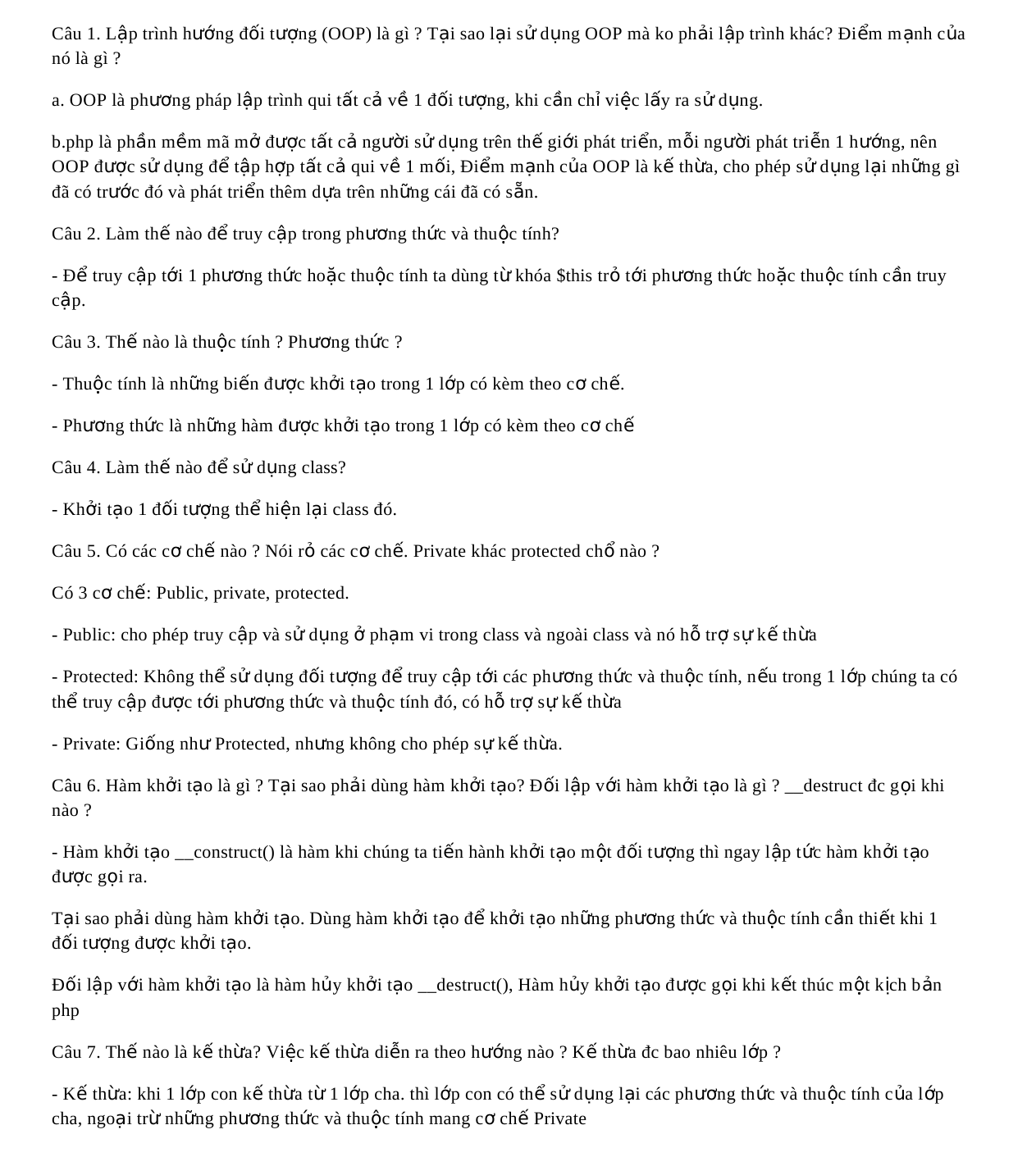
—không biết, vấn dùng for vs while duyệt hoài mà )  
6- Để chuyển mảng thành chuỗi ta dùng hàm gì ?. Để tách chuỗi thành mảng ta dùng hàm gì?. Để gộp mảng ta dùng hàm gì ? Để tách mảng ta dùng hàm gì ?.  
– Chuyển mảng thành chuỗi dùng : implode(“Chuỗi nối”,$mảng)  
– Chuyển chuỗi thành mảng dùng : explode(“Chuỗi tách”,$string)  
– Để gộp mảng dùng : array\_merge($mang1, $mang2);  
– Để tách mảng dùng : array\_slice($mang, vị trí tách, số lượng tách);  
7- Cho biết sự khác nhau giữa serialize và json\_encode. Lý giải theo cách bạn hiểu.  
– Json\_encode : để giải mã dữ liệu json đã được mã hóa bằng json\_decode : từ phiên bản php 5.2 trở đi, người ta thay thế json\_encode và decode cho serialize  
– Serialize : chuyển đổi mảng thành string và unseriallize để lấy lại mảng ban đầu  
Không biết luôn  
8- Cookie và session có gì khác nhau ?. Người ta nói bản thân của session là cookie, điều đó đúng hay sai ?. Nếu muốn tăng thời gian sống của session thì thiết lập thế nào ?.  
– Cookie lưu ở client trên browser  
– Session : lưu trên server và đồng thời ở client cũng có 1 cookie sinh ra tương ứng đúng như session id của nó trên server : PHPSESSID, các giá trị biến của session đc lưu trong file đó (dùng để quản lý đăng nhập)  
– coi session là cookie cũng đúng.  
– chả biết chỉnh ở đâu? Php.ini chăng ?  
9- Giả sử bạn là chủ của server, làm thế nào có thể biết được các file session lưu ở đâu ?. Bạn có thể can thiệp, chỉnh sửa file session này hay không ?.  
Biết được. lưu ở file php.ini (tham số session.save\_path)

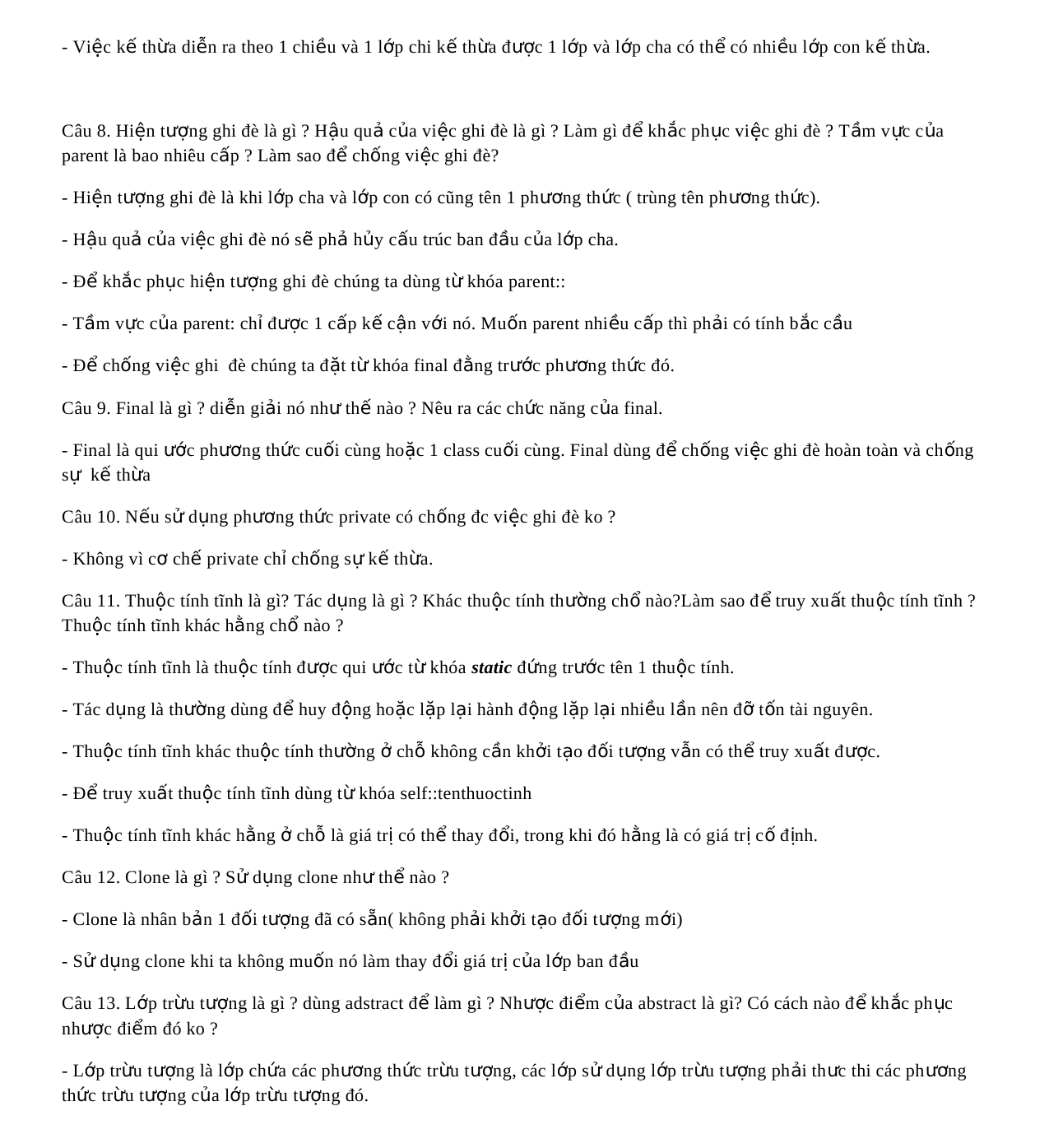
10- Theo bạn, sự khác nhau của toán tử & và && là gì ?. Hãy cho biết $a++ và ++$a khác nhau ở đâu ?. Tính nhanh kết quả của đoạn code sau trong 2 phút:  
<?php  
$a=5;  
echo $a++ + $a– + –$a + ++$a + $a++ + ++$a – $a– + $a;  
?>

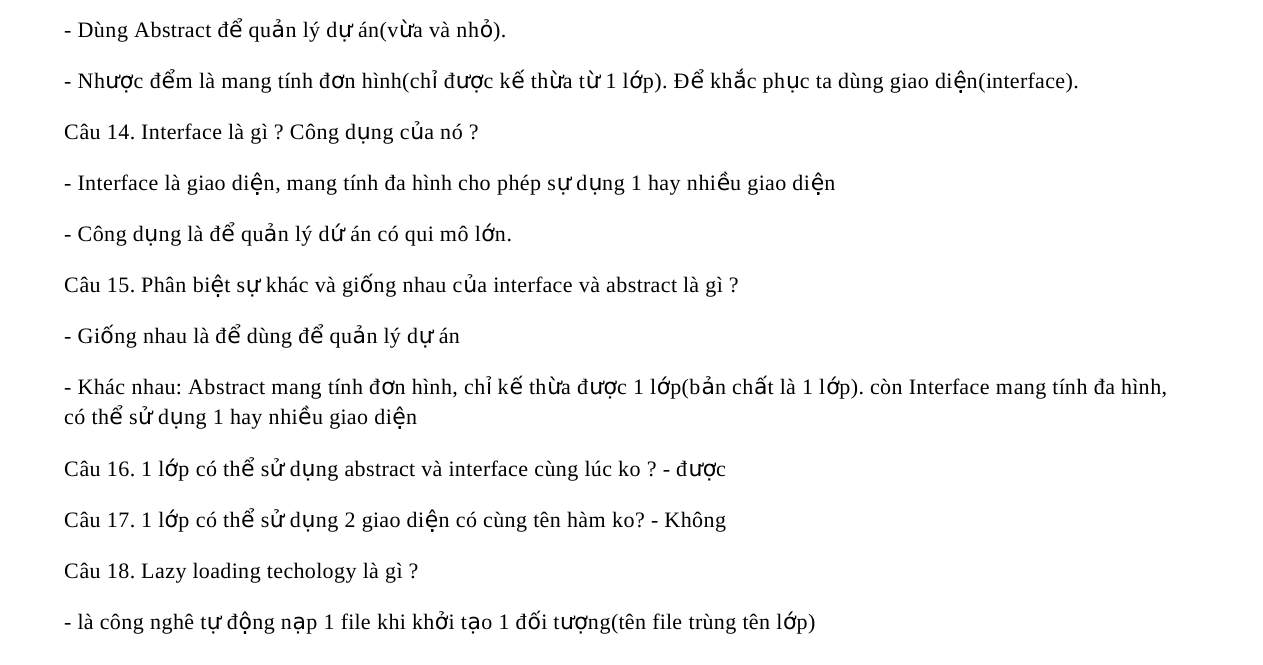
11- Mysql\_close() cần thiết như thế nào trong thực tế ?. Vì sao ít thấy người dùng áp dụng nó.  
– Chắc chắn là cần thiết mà cần ntn thì cũng k biết. K có nó vẫn chạy phà phà  
– Có lẽ có mysql\_close() chỉ để tăng tính chặt chẽ của chương trình  
12- Muốn chuẩn hóa dữ liệu về utf-8 ta phải làm gì ?.  
– Sử dụng dạng utf-8 trong meta charset  
– Sử dụng utf-8-unicode-ci trong sql  
13- Làm sao lấy được id vừa tiến hành thêm vào CSDL ngay ?.  
– Sử dụng hàm mysql\_insert\_id hoặc sử dụng limit(0,1) DESC  
– Trong các framework sẽ có các hàm khác nhau hỗ trợ  
14- Để đếm số dòng, ta nên dùng mysql\_num\_rows() trên câu truy vấn “select \* from….” hay count trực tiếp trong câu truy vấn ?.  
– Không hiểu ý câu hỏi lắm.  
15- Việc enter có giá trị hay không trong textarea bình thường (không nhúng khung soạn thảo). Và làm thế nào để có thể hiển thị ra bên ngoài thao tác xuống dòng như bình thường.  
– Ít làm nên không hiểu lắm

16- fgets() và fgetc() khác nhau như thế nào ?. fgets() và fread() khác nhau ra sao ?. feof là gì, nó có ý nghĩa như thế nào ?. file\_get\_contents() khác gì so với fread() ?.  
– File\_get\_contents() : dùng để lấy nội dung của files  
– Fread phải open xong mới đọc được, phải chỉ ra kích thước file

Đây là 1 số cầu hỏi khác



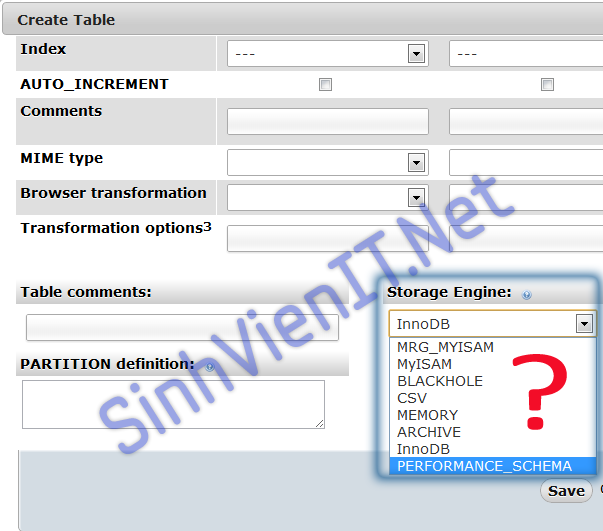




1. Phân biệt inner join với left join  
     
   *INNER JOIN*: Kết quả trả về chỉ gồm những bản ghi thỏa mãn điều kiện nối hai bảng  
   *LEFT JOIN*: select tất cả bản ghi có trong bảng thứ nhất trong mệnh đề FROM, bất kể nó có match với bảng thứ hai hay không. Những bản ghi không match với bảng thứ hai nó sẽ để giá trị NULL  
     
   *RIGHT JOIN*:select tất cả những bản ghi có trong bảng thứ hai trong mệnh đề FROM, bất kể những bản ghi đó có match với bảng thứ nhất hay không  
     
   *FULL OUTER JOIN*: là sự kết hợp của LEFT OUTER JOIN và RIGHT OUTER JOIN. Kết quả trả về tất cả những bản ghi match hoặc không match giữa hai bảng  
     
   *CROSS JOIN*  
   Cái này chính là tích Đề các. Mỗi bản ghi trong bảng thứ nhất của mệnh đề FROM sẽ kết hợp (vô điều kiện) với các bản ghi trong bảng thứ ha  
     
   *EQUI JOIN*: chính là INNER JOIN cộng thêm điều kiện: nó select toàn bộ các trường có trong hai bảng tham gia nối.  
     
   *NATURAL JOIN*:giống EQUI JOIN cộng thêm điều kiện: nếu hai bảng có hai trường giống nhau thì nó chỉ lấy ra 1 trường.  
     
   2. index trong sql  
   Index (chỉ mục) thực chất là một dạng tương tự như phần mục lục của 1 cuốn sách  
   Trong SQL, nó đóng vai trò tăng tốc truy xuất  
   index trong CSDL có hai loại: Clustered Index và Non-Clustered Index  
   Clustered index: thường được tự tạo ra khi bảng có primary key, clustered index chính là unique index  
   Non-Clustered Index: foreign key là non-clustered index  
     
   Tham khảo:  
   [http://sinhvienit.net/forum/clustere...dex.41251.html](http://sinhvienit.net/forum/clustered-index-va-non-clustered-index.41251.html)  
   [http://www.sqlviet.com/blog/index-gi...en-nhu-the-nao](http://www.sqlviet.com/blog/index-giup-tang-hieu-nang-thuc-hien-nhu-the-nao)  
     
   3. Phân biệt stored procedure ,function và trigger,view  
     
   1. Khung nhìn (view) có thể được xem như là một bảng “ảo” trong cơ sở dữ liệu có nội dung được định nghĩa thông qua một truy vấn (câu lệnh SELECT). Như vậy, một khung nhìn trông giống như một bảng với dữ liệu quan sát được là một tập bao gồm các dòng và các cột. Điểm khác biệt giữa khung nhìn và bảng là: khung nhìn không được xem là một cấu trúc lưu trữ dữ liệu tồn tại trong cơ sở dữ liệu. Thực chất dữ liệu quan sát được trong khung nhìn được lấy từ các bảng thông qua câu lệnh truy vấn dữ liệu.  
     
   2. Một thủ tục là một đối tượng trong cơ sở dữ liệu bao gồm một tập nhiều câu lệnh SQL được nhóm lại với nhau thành một nhóm (như tập tin BAT trong DOS vậy). Khi một thủ tục lưu trữ đã được định nghĩa, nó có thể được gọi thông qua tên thủ tục, nhận các tham số truyền vào, thực thi các câu lệnh SQL bên trong thủ tục và có thể trả về các giá trị sau khi thực hiện xong.  
   3. Một trigger là một đối tượng với nội dung là một tập các câu lệnh SQL (tương tự thủ tục). Mỗi một trigger gắn liền với một bảng nào đó trong CSDL và được tự động kích hoạt khi xảy ra những giao tác làm thay đổi dữ liệu trong bảng (INSERT, UPDATE, DELETE).  
     
   Điểm khác biệt giữa thủ tục lưu trữ và trigger là: các thủ tục lưu trữ được thực thi khi người sử dụng có lời gọi đến chúng còn các trigger lại được "gọi" tự động khi xảy ra những giao tác làm thay đổi dữ liệu trong các bảng  
     
   4. Function: giống như produre nhưng có kết quả trả về  
     
   4. Phân biệt JOIN and UNION  
     
   A Join: cho phép kết nối dữ liệu liên quan trong các bảng khác nhau  
   A Union: hợp dữ liệu của 2 bảng mà có các cột loại data giống nhau  
     
   5. Phân biệt UNION and UNION ALL  
   UNION và UNION ALL đều dùng để hợp hai tập bản ghi cùng cấu trúc, nhưng giữa hai mệnh đề có một khác biệt khá tinh tế: UNION loại bỏ các bản ghi trùng lặp trước khi trả lại kết quả, còn UNION ALL giữ lại tất cả các bản ghi từ hai tập ban đầu  
     
   6. WHERE clause and HAVING clause  
     
   where là câu lệnh dùng để đặt điều kiện lọc trên từng bộ.  
   Having cũng là câu lệnh đặt điều kiện nhưng là ở trên 1 nhóm xác định. thường là đi kèm với câu lệnh group by  
     
   7. How to select first 5 records from a table?  
     
   SELECT TOP 5 \* FROM EMP  
     
   8. What is difference between DELETE and TRUNCATE commands?  
     
   DELETE luôn luôn chậm hơn TRUNCATE. Càng có nhiều bản ghi DELETE càng chậm, còn TRUNCATE thì không phụ thuộc vào lượng dữ liệu. DELETE có phạm vi ứng dụng rộng hơn; còn TRUNCATE chỉ dùng được mỗi một việc, nhưng nó lại làm rất nhanh

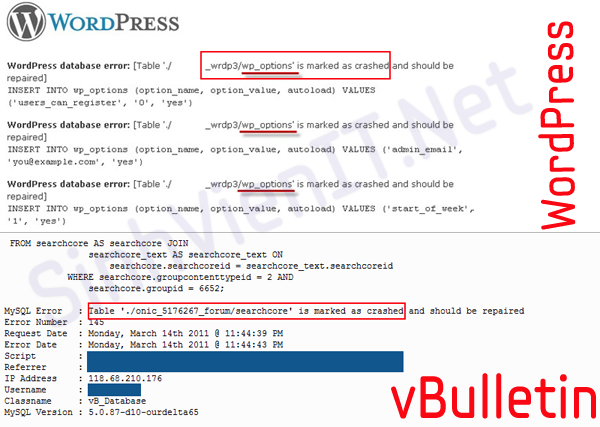
## 9. InnoDB,MyISAM và MEMORY, nên sử dụng Storage Engine nào khi dùng MySQL

MySQL là hệ quản trị cơ sở dữ liệu miễn phí được sử dụng nhiều nhất khi ứng dụng được viết bằng ngôn ngữ PHP. Tuy nhiên, khi tạo 1 bảng trong MySQL sẽ có nhiều kiểu [Storage Engine](http://sinhvienit.net/forum/tags.php?tag=storage+engine) để bạn lựa chọn. Trong đó có 3 kiểu lưu trữ bảng được dùng nhiều nhất là [InnoDB](http://sinhvienit.net/forum/tags.php?tag=innodb), [MyISAM](http://sinhvienit.net/forum/tags.php?tag=myisam) và [Memory](http://sinhvienit.net/forum/tags.php?tag=memory).   
  
Nhưng ta nên dùng InnoDB,MyISAM hay MEMORY ? Tại sao lại như thế ? Bài viết này sẽ giúp các bạn nắm được cách hoạt động của các Storage Engine này. Từ đó có thể biết được cách chọn loại phù hợp khi xây dựng website của mình để hiệu suát và độ ổn định đạt được cao nhất.

[[](http://sinhvienit.net/forum/innodb-myisam-va-memory-nen-su-dung-storage-engine-nao-khi-dung-mysql.209109.html)](http://sinhvienit.net/forum/innodb-myisam-va-memory-nen-su-dung-storage-engine-nao-khi-dung-mysql.209109.html" \o " InnoDB,MyISAM và MEMORY, nên sử dụng Storage Engine nào khi dùng MySQL" \t "_blank)

*Nên chọn MyISAM,InnoDB hay Memory ?*

1. MyISAM trong MySQL  
  
Đây là kiểu Storage Engine mặc định khi tạo bảng và được dùng phổ biết nhất. Storage Engine này cho phép lập chỉ mục toàn cột (Full Text Index). Do đó, Storage Engine này cho tốc độ truy suất (Đọc và tìm kiếm) nhanh nhất trong các Storage Engine.  
  
Tuy nhiên, Nhược điểm của MyISAM là hoạt động theo kiểu Table Level locking nên khi cập nhật (Thêm,xóa,sửa) 1 bản ghi nào đó trong cùng 1 table thì table đó sẽ bị khóa lại, không cho cập nhật (Thêm,xóa,sửa) cho đến khi thao tác cập nhật trước đó thực hiện xong.  
  
Ngoài ra, do thiết kế đơn giản và không kiểm tra ràng buộc dữ liệu nên loại Storage Engine này dễ bị hỏng chỉ mục và dễ bị Crash. Đây là cơn ác mộng của các webmaster khi table Crash là table có dung lượng lớn, khi phục hồi rất lâu và hồi hộp [http://sinhvienit.net/forum/images/smilies/LaiPro_NewYahoo/42.gif](http://sinhvienit.net/forum/innodb-myisam-va-memory-nen-su-dung-storage-engine-nao-khi-dung-mysql.209109.html)

[[](http://sinhvienit.net/forum/innodb-myisam-va-memory-nen-su-dung-storage-engine-nao-khi-dung-mysql.209109.html)](http://sinhvienit.net/forum/innodb-myisam-va-memory-nen-su-dung-storage-engine-nao-khi-dung-mysql.209109.html" \o " InnoDB,MyISAM và MEMORY, nên sử dụng Storage Engine nào khi dùng MySQL" \t "_blank)

*Table dạng MyISAM hay bị Crash*

*Làm sao để chuyển 1 bảng từ Storage Engine khác (VD: InnoDB) sang MyISAM ?*  
Bạn có thể dùng truy vấn sau:

Code:

ALTER TABLE table\_name ENGINE = MyISAM;

2. InnoDB trong MySQL  
  
Đây là kiểu Storage Engine mới hơn MyISAM. Storage Engine này không hỗ trợ Full Text Index như MyISAM (Tin mừng là sắp có hỗ trợ ở các phiên bản mới, hiện tại đã có beta rồi [http://sinhvienit.net/forum/images/smilies/LaiPro_NewYahoo/4.gif](http://sinhvienit.net/forum/innodb-myisam-va-memory-nen-su-dung-storage-engine-nao-khi-dung-mysql.209109.html)) nhưng hỗ trợ quan hệ giữa các bảng (Khóa ngoại). Do đó, kiểu Storage này kiểm tra tính toàn vẹn dữ liệu và ràng buộc rất cao => Khó sảy ra tình trạng hỏng chỉ mục và Crash như MyISAM.  
  
Ngoài ra, kiểu Storage Engine này hoạt động theo cơ chế Row Level Locking nên khi cập nhật (Thêm,xóa,sửa) 1 bảng thì chỉ có bản ghi đang bị thao tác bị khóa mà thôi, các hoạt động khác trên table này vẫn diễn ra bình thường.  
  
Vì những tính chất trên, kiểu Storage Engine này thích hợp sử dụng cho Ngân hàng và các trang web có tần suất cập nhật dữ liệu cao như Mạng xã hội, diễn đàn....  
  
Tuy nhiên, nó có nhược điểm là hoạt động tốn RAM hơn so với MyISAM [http://sinhvienit.net/forum/images/smilies/LaiPro_NewYahoo/4.gif](http://sinhvienit.net/forum/innodb-myisam-va-memory-nen-su-dung-storage-engine-nao-khi-dung-mysql.209109.html) (Nếu MyISAM mà tần suất insert hay update cao thì nếu cấu hình chưa đủ mạnh thì khéo còn tốn RAM nhiều hơn InnoDB vì hàng đợi lớn [http://sinhvienit.net/forum/images/smilies/LaiPro_NewYahoo/71.gif](http://sinhvienit.net/forum/innodb-myisam-va-memory-nen-su-dung-storage-engine-nao-khi-dung-mysql.209109.html) )  
  
=> vBulletin, mã nguồn diễn đàn lớn nhất hiện nay thật sai lầm khi chọn MyISAM làm kiểu Storage Engine cho các bảng dữ liệu forum. Vì thế mà các trang lớn mới hay bị Crash bảng như vậy (Database Error)  
  
*Làm sao để chuyển 1 bảng từ Storage Engine khác (VD: MyISAM) sang InnoDB ?*  
Bạn có thể dùng truy vấn sau:

Code:

ALTER TABLE table\_name ENGINE = InnoDB;

## (Lưu ý là nếu trước đó table này dùng MyISAM mà có cột nào đặt Full Text Index thì bạn phải xóa Full Text Index trên cột đó đi mới có thể chuyển được) 3. MEMORY trong MySQL Đây là kiểu Storage Engine được lưu trữ dữ liệu trực tiếp lên RAM nên tốc độ truy xuất và cập nhật rất nhanh. Vì thế, nó được dùng làm các table chứa dữ liệu tạm, chứa các phiên làm việc của user... Khi khởi động lại dịch vụ MySQL thì dữ liệu trên bảng có Storage Engine là MEMORY sẽ mất hết dữ liệu. Chính vì thế nên khi các bạn khởi động lại mysqld trên VPS hay Server thì sẽ thấy số người online = 0 [http://sinhvienit.net/forum/images/smilies/LaiPro_NewYahoo/4.gif](http://sinhvienit.net/forum/innodb-myisam-va-memory-nen-su-dung-storage-engine-nao-khi-dung-mysql.209109.html) MEMORY sử dụng cơ chế table-level locking như MyISAM. *Dung lượng của 1 bảng Storage Engine dạng MEMORY tối đa là bao nhiêu ?* Nó phụ thuộc vào cấu hình thông số max\_heap\_table\_size trong file my.cnf, mặc định 1 bảng kiểu MEMORY có dung lượng tối đa là 16MB. Nếu vượt quá bạn sẽ nhận được lỗi: Table xyz is full... [http://sinhvienit.net/forum/images/smilies/LaiPro_NewYahoo/4.gif](http://sinhvienit.net/forum/innodb-myisam-va-memory-nen-su-dung-storage-engine-nao-khi-dung-mysql.209109.html) Tóm lại,  -> Với 1 ứng dụng có tần suất đọc cao như trang tin tức,blog... thì bạn nên dùng MyISAM. -> Với ứng dụng có tần suất insert và update cao như: Diễn đàn, mạng xã hội.. thì bạn nên dùng InnoDB -> Bạn nên dùng MEMORY Storage Engine cho các table chứa dữ liệu tạm và thông tin phiên làm việc của người dùng (Session) -> Việc chuyển đổi 1 table từ storage engine này sang storage engine khác sẽ diễn ra tương đối lâu nếu dữ liệu trên table lớn. Do đó cần kiên nhẫn.

Git

|  |  |  |
| --- | --- | --- |
| Git task | Notes | Git commands |
| [Tell Git who you are](https://www.atlassian.com/git/tutorials/setting-up-a-repository/git-config) | Configure the author name and email address to be used with your commits.  Note that Git [strips some characters](http://stackoverflow.com/questions/26159274/is-it-possible-to-have-a-trailing-period-in-user-name-in-git/26219423#26219423) (for example trailing periods) from user.name. | git config --global user.name "Sam Smith"  git config --global user.email sam@example.com |
| [Create a new local repository](http://atlassian.com/git/tutorial/git-basics#!init) |  | git init |
| [Check out a repository](http://atlassian.com/git/tutorial/git-basics#!clone) | Create a working copy of a local repository: | git clone /path/to/repository |
| For a remote server, use: | git clone username@host:/path/to/repository |
| [Add files](http://atlassian.com/git/tutorial/git-basics#!add) | Add one or more files to staging (index): | git add <filename>  git add \* |
| [Commit](http://atlassian.com/git/tutorial/git-basics#!commit) | Commit changes to head (but not yet to the remote repository): | git commit -m "Commit message" |
| Commit any files you've added with git add, and also commit any files you've changed since then: | git commit -a |
| [Push](http://atlassian.com/git/tutorial/remote-repositories#!push) | Send changes to the master branch of your remote repository: | git push origin master |
| [Status](http://atlassian.com/git/tutorial/git-basics#!status) | List the files you've changed and those you still need to add or commit: | git status |
| [Connect to a remote repository](http://atlassian.com/git/tutorial/remote-repositories#!remote) | If you haven't connected your local repository to a remote server, add the server to be able to push to it: | git remote add origin <server> |
| List all currently configured remote repositories: | git remote -v |
| [Branches](http://atlassian.com/git/tutorial/git-branches) | Create a new branch and switch to it: | git checkout -b <branchname> |
| Switch from one branch to another: | git checkout <branchname> |
| List all the branches in your repo, and also tell you what branch you're currently in: | git branch |
| Delete the feature branch: | git branch -d <branchname> |
| Push the branch to your remote repository, so others can use it: | git push origin <branchname> |
| Push all branches to your remote repository: | git push --all origin |
| Delete a branch on your remote repository: | git push origin :<branchname> |
| [Update from the remote repository](http://atlassian.com/git/tutorial/remote-repositories) | Fetch and merge changes on the remote server to your working directory: | git pull |
| To merge a different branch into your active branch: | git merge <branchname> |
| View all the merge conflicts:  View the conflicts against the base file:  Preview changes, before merging: | git diff  git diff --base <filename>  git diff <sourcebranch> <targetbranch> |
| After you have manually resolved any conflicts, you mark the changed file: | git add <filename> |
| **Tags** | You can use tagging to mark a significant changeset, such as a release: | git tag 1.0.0 <commitID> |
| CommitId is the leading characters of the changeset ID, up to 10, but must be unique. Get the ID using: | git log |
| Push all tags to remote repository: | git push --tags origin |
| [Undo local changes](http://atlassian.com/git/tutorial/undoing-changes) | If you mess up, you can replace the changes in your working tree with the last content in head:  Changes already added to the index, as well as new files, will be kept. | git checkout -- <filename> |
| Instead, to drop all your local changes and commits, fetch the latest history from the server and point your local master branch at it, do this: | git fetch origin  git reset --hard origin/master |
| **Search** | Search the working directory for foo(): | git grep "foo()" |

Thi Javascript HTML

**1.      JavaScript là ngôn ngữ xử lý ở:**

a.       Client

b.      Server

c.      Server/client

d.      Không có dạng nào.

**2.      Javascript là ngôn ngữ thông dịch hay biên dịch**

a.       Thông dịch

b.      Diễn dịch

c.      Cả hai dạng

d.      Không có dạng nào ở trên

**3.      Phương thức viết chương trình của Javascript như thế nào?**

a.       Viết riêng một trang

b.      Viết chung với HTML

c.      Cả hai dạng

d.      Không có dạng nào.

**4.      Javascript là ngôn ngữ kịch bản có dấu được mã nguồn không?**

a.       Không dấu được vì các kịch bản chạy ở client.

b.      Dấu được vì chương trình hoạt động độc lập với trình duyệt

c.      Hai phát biểu đều sai.

**5.      JavaScript được bắt đầu bằng?**

a.       <script> …</script>

b.      <Javascript> …<Javascript>

c.      <java>   </java>

d.      Tất cả các dạng trên.

**6.      Javascript có các dạng biến?**

a.       Number, String, Boolean

b.      Number, Integer, char

c.      Number, String, Boolean, Null

d.      Tất cả các loại trên.

**7.      Trong Javascript hàm parseInt() dùng để làm gì?**

a.       Chuyển một chuỗi thành số

b.      Chuyển một chuỗi thành số nguyên

c.      Chuyển một chuỗi thành số thực

d.      Chuyển một số nguyên thành một chuỗi

**8.      Trong Javascript hàm parseFloat() dùng để làm gì?**

a.       Chuyển một chuỗi thành số

b.      Chuyển một chuỗi thành số thực

c.      Chuyển một chuỗi thành số nguyên

d.      Chuyển một số nguyên thành một chuỗi

**9.      Lệnh prompt trong Javascript để làm gì?**

a.       Hiện một thông báo nhập thông tin

b.      Hiện một thông báo dạng yes, No

c.      Cả hai dạng trên

d.      Không có lệnh nào đúng

**10. Trong Javascript sự kiện Onload thực hiện khi:**

a.       Khi bắt đầu chương trình chạy

b.      Khi click chuột

c.      Khi kết thúc một chương trình

d.      Khi di chuyển chuột qua.

**11. Trong Javascript sự kiện OnUnload thực hiện khi nào?**

a.       Khi bắt đầu chương trình chạy

b.      Khi click chuột

c.      Khi kết thúc một chương trình

d.      Khi di chuyển chuột qua.

**12. Trong Javascript sự kiện Onblur thực hiện khi nào?**

a.       Khi một đối tượng trong form mất focus.

b.      Khi một đối tượng trong form có focus

c.      Khi di chuyển con chuột qua form.

d.      Khi click chuột vào nút lệnh

**13. Trong Javascript sự kiện OnMouseOver thực hiện khi nào?**

a.       Khi một đối tượng trong form mất focus.

b.      Khi một đối tượng trong form có focus

c.      Khi di chuyển con chuột qua một đối tượng trong form.

d.      Khi click chuột vào nút lệnh

**14. Trong Javascript sự kiện Onclick thực hiện khi nào?**

a.       Khi một đối tượng trong form mất focus.

b.      Khi một đối tượng trong form có focus

c.      Khi click chuột vào một đối tượng trong form.

d.      Khi click chuột vào nút lệnh

**15. Trong Javascript sự kiện Onchange thực hiện khi nào?**

a.       Khi một đối tượng trong form mất focus.

b.      Khi một đối tượng trong form có focus

c.      Xảy ra khi giá trị của một trường trong form được người dùng thay đổi

d.      Khi click chuột vào nút lệnh

**16. Trong Javascript đoạn mã sau cho ra kết quả gì?**

<script>

function kiemtra(){

window.open("http://www.vnn.vn","Chao");

}

</script>

</head>

<body onload ="kiemtra()"></body>

a.     Khi chạy thì một trang khác (VNN) được hiện ra .

b.     Không chạy được vì sai

c.     Khi kết thúc thì một site khác hiện ra

d.     Hiện một trang vnn duy nhất.

**17. Thẻ <input type='text'> dùng để làm gì?**

a.       Tạo một ô text để nhập dữ liệu 1 dòng

b.      Tạo một ô password

c.      Tạo một  textbox cho phép nhập liệu nhiều dòng

d.      Tất cả các ý trên

**18.  Thẻ <input type='Password'> dùng để làm gì?**

a.       Tạo một ô text để nhập dữ liệu 1 dòng

b.      Tạo một ô password

c.      Tạo một  textbox cho phép nhập liệu nhiều dòng

d.      Tất cả các ý trên

**19. Thẻ <textarea ></texterea> dùng để làm gì?**

a.       Tạo một ô text để nhập dữ liệu 1 dòng

b.      Tạo một ô password

c.      Tạo một  textbox cho phép nhập liệu nhiều dòng

d.      Tất cả các ý trên

**20. Thẻ <input type=”Submit”  dùng để làm gì?**

a.       Tạo một ô text để nhập dữ liệu

b.      Tạo một nút lệnh dùng để gửi tin trong form đi

c.      Tạo một nút lệnh dùng để xóa thông tin trong form

d.      Tất cả các ý trên

**21. Thẻ <input type=”Radio” …> dùng để làm gì?**

a.       Tạo một ô text để nhập dữ liệu

b.      Tạo một nhóm đối tượng chọn nhưng  chọn duy nhất

c.      Tạo một  cùng có nhiều cột nhiều dòng

d.      Tất cả các ý trên

**22. Thẻ <input type=”checkbox” …> dùng để làm gì?**

a.       Tạo một ô text để nhập dữ liệu

b.      Tạo một nhóm đối tượng chọn được nhiều đối tượng

c.      Tạo một  cùng có nhiều cột nhiều dòng

d.      Tất cả các ý trên

**23. Thẻ <input type=”button” …> dùng để làm gì?**

a.       Tạo một ô text để nhập dữ liệu

b.      Tạo một nút lệnh lên trên form

c.      Tạo một  cùng có nhiều cột nhiều dòng

d.      Tất cả các ý trên

**24. Lệnh lặp for có dạng như thế nào?**

a.       for ( biến = Giá trị đầu; Điều kiện; Giá trị tăng)

b.      for ( biến = Giá trị đầu; Giá trị tăng; điều kiện)

c.      for ( biến = Điều kiện; Giá trị tăng; Giá trị cuối)

d.      Tất cả các dạng trên.

**25. Vòng lặp While là dạng vòng lặp?**

a.       Không xác định và xét điều kiện rồi mới lặp

b.      Không xác định và  lặp rồi mới xét điều kiện

c.      Cả hai dạng trên

d.      Không tồn tại dạng nào ở trên.

**26. Vòng lặp (Do.. while) là dạng vòng lặp?**

a.       Không xác định và xét điều kiện rồi mới lặp

b.      Không xác định và lặp rồi mới xét đ-iều kiện

c.      Cả hai dạng trên

d.      Không có dạng nào.

**27. Lệnh break kết hợp với vòng for dùng để?**

a.       Ngưng vòng for nếu gặp lệnh này

b.      Không có ý nghĩa trong vòng lặp

c.      Nhảy đến một tập lệnh khác

d.      Không thể kết hợp được.

**28. Hàm alert() dùng để làm gì?**

a.       Dùng để hiện một thông.

b.      Dùng để hiện một thông báo nhập

c.      Dùng để chuyển đổi số sang chữ

d.      Tất cả các dạng trên.

**29. Thẻ <Frameset cols> </frameset>**

a.       Dùng để chia trang web ra nhiều phần theo cột

b.      Dùng để chia trang web ra nhiều phần theo dòng

c.      Tất cả dạng trên

d.      Không có ý nào ở trên.

**30. Thẻ <Frameset rows > </frameset>**

a.       Dùng để chia trang web ra nhiều phần theo cột

b.      Dùng để chia trang web ra nhiều phần theo dòng

c.      Tất cả dạng trên

d.      Không có ý nào ở trên.

**31. Thẻ <Frame src=” duong dan  ”>**

a.       Dùng để chèn ảnh

b.      Dùng để lấy dữ liệu từ một trang khác

* 1 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 2 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 3 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 4 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 5 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 6 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 7 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 8 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 9 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 10 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 11 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 12 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 13 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 14 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 15 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 16 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 17 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 18 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 19 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 20 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 21 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 22 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 23 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 24 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 25 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 26 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 27 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 28 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 29 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 30 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)
* 31 [ABCD](http://www.studyvn.com/trac-nghiem/post/31-cau-trac-nghiem-HTML-,-Javascript-96)