

Project report

1. XHTML - CSS
 - 1.1. Bonus part
2. Lamp configuration
 - 2.1. Bonus part
3. PHP Basic
4. PHP Advanced
5. Organize your code
6. Database
 - 6.1. Database : select & insert
 - 6.2. Database
7. Member management
8. XML / RSS
9. Post exam & javascript
10. Ajax & API
11. File management

1. XHTML - CSS

- *What does the property "float" stand for ?*

The property "float" stands for not fix a block but to put it on the left or the right of the page. It's often used to images.

The possible values are :

- left or right
- none : the element is not float, as default
- inherit : the property is inherited from the parent

- *What is the property "overflow" ?*

This property is to manage the contents which are out the view or out a block. The most used value is scroll. But there is also :

- visible : to show all which is outside
- hidden : to hide all which is outside
- auto : to put a scrollbar if it's necessary
- inherit : to herit the property from its parent

Bonus part

- *Which web browser are supporting CSS3 and HTML5 today ?*

CSS3 is supported by Chrome 30, Firefox 25, Opera 12, Safari 6 and Internet Explorer 10.

HTML5 is supported by Chrome 30, Firefox 25, Opera 12, Safari 6 and Internet Explorer 10.

In fact, we can say that most browsers support these two languages. Indeed, the statistics by Statcounter, in september 2013 and in the world, show us that Chrome is used by 41% of users, Internet Explorer by 29%, Firefox by 18%, Safari by 9% and Opera by 1%. So these 5 browsers are used by 98% of users.

We found that the website <http://html5readiness.com/> shows us very simply the browsers supporting and the fonctionnalities details.

- *What is the framework "modernizr" for ?*

Modernizr is a javascript that detects the html5 and css3 features in the browsers and warns if some of features are possible to use or not. It permits to take advantage from the last browsers and to not worry about the olds.

2. Lamp configuration

- *What is the command to know if apache is running or not*

service apache2 status

But we can also going to <http://localhost> with a browser, if the page is load, that means that apache is running, else it's not.

- *Describe what you see and what mean each lines of*

```
<Directory ${HOME}/public_html>
    Options Indexes FollowSymLinks MultiViews
    AllowOverride None
    Order allow,deny
    allow from all
</Directory>
```

It declares a folder which will be watches by Apache in the personal file in the folder public_html. In this one, we allowed :

- the indexing of files meadow in the folder (in the absence of the index file)
- the follow-up of symbolic links
- the research of files not indexed

So, none .htaccess file will be read and all queries from all hotes will be accepted.

- *What are "logs" ? Is it specific to apache ?*

Most of softwares has logs to permit to draw the error conditions or just for legal reasons (a webmaster has to keep the IP address from the users who access to the site during one year minimum).

- *Change the "max_execution_time". Is it working ? Why ?*

This variable defines the maximum time during which a script can turn before being automatically killed. The values changed in the php.ini allow to show the errors received by the server and usually hided to the users in an environment of production. It is impossible to change this value to "safe mode".

Bonus part

- *What is the difference between apache & nginx ?*

Apache is a process-based server and nginx is an event-based web server.

- *Same question with node.js ?*

Node.js is a software which permits to use javascript from the server while apache is a server..

- *What is the goal of less ? Try to write LESS : <http://lesscss.org/>*

LESS is a dynamic stylesheet language to make the css work easier. It permits to use variables, mixins, operations and functions.

The procedure is :

- you write a .less file with all your variables, functions, ...
- you compile the file and a new .css file is created to permit the browser to read the commands that you want

There is also SASS, another stylesheet language, similar as LESS.

- *Test your apache install with : <http://en.wikipedia.org/wiki/ApacheBench> What is the result ?*

```
This is ApacheBench, Version 2.3 <$Revision: 655654 $>  
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/  
Licensed to The Apache Software Foundation, http://www.apache.org/
```

3. PHP Basic

- *Try ending this code snippet by adding lines in order to have printed "Hello terminator ! The time is [time]"*

```
<?php
    $date      = date ('H:i:s');
    $username = 'Terminator';

    echo "Hello ".$username." ! ";
    echo "The time is ".$date;

?>
```

- *Thanks to this new structure, improve your last exercise by adding a new cool feature. Thanks to the date of movie creation, create cases :*
 - *The year is in future : print "The movie "Back to the future" will be released in 2 years"*
 - *The year is in the past : print the same thing as before*
 - *The year is the current one : print "The movie "Back to the future" will be released in soon"*

```
<?php
    $date = date("Y");
    switch ($date) {
        case 2014:
            echo 'The movie "Back to the future" will be released in 2 years';
            break;
        case 2012:
            echo 'The movie "Back to the future" will be released in 2 years';
            break;
        case 2013:
            echo 'Back to the future" will be released in soon';
            break;
    }

?>
```

- *Describe each loop*
 - *While*

The while loop is to repeat an instruction while a condition is respected. The loop is not running if the condition is not respected as the beginning. So if the condition is never respected, the loop never starts.

- *Do ... while*

The do...while loop is to do an instruction while a condition is respected. The loop is started even the condition is not respected but it stops if the condition is not any more.

- *For()*

The for loop is to repeat an instruction for a number of time known.

- *Foreach*

The foreach loop is to repeat an instruction foreach values of an array or an object.

4. PHP Advanced

- *What is the purpose of print_r function ? (see php.net)*

The print_r function permits to print the content in a understandable way to an human.

- *Print all elements from \$pizzaToppings array thanks to "for() []" loop. All you need to do is access the appropriate element of the array using its index number. You can use "count()" to get the number of elements in an array*

```
<?php
    $pizzaToppings = array('onion', 'tomato', 'cheese', 'anchovies', 'ham', 'pepperoni');

    for($i=0 ; $i<count($pizzaToppings) ; $i++)
    {
        echo $i." ".$pizzaToppings[$i]."<br>";
    }

?>
```

- *Using "foreach" syntax, print the content of movie array.*

```
<?php
    $pizzaToppings = array('onion', 'tomato', 'cheese', 'anchovies', 'ham', 'pepperoni');

    foreach($pizzaToppings as $j)
    {
        echo $j." <br>";
    }

?>
```

- *What is the difference between GET or POST ?*

The difference is on the way to send the datas :

- with GET, datas are sent by the url, so they are visible to the user.

With you example, we obtain : <http://localhost:8888/add.php?name=peter&age=18>.

- with POST, datas are send by the sql query, so they are not visible to the user, but still easily accessible for a person who wants to get them.

- *How do we call vars beginning by "\$_"*

The vars beginning by "\$_" are called predefined variables.

5. Organize your code

- *What do the XHTML represent ? The view, the model or the controller ?*

The xhtml represents the view because it permits to show datas on the webpages.

- *What is a "Front controller pattern" ?*

It's a design pattern which aims to simplify the site management by imposing an unique entry point to check easily the flow of navigation.

6. Database

```
<?php
    $pdo_options[PDO::ATTR_ERRMODE] = PDO::ERRMODE_EXCEPTION;
    $bdd = new PDO('mysql:host=sql-users.ece.fr;dbname=DBNAME;port=3305', 'username',
    'password', $pdo_options);
    $answer = $bdd->query('SELECT pseudo, isAdmin FROM Member');
    while ($data = $answer->fetch())
    {
        echo $data['pseudo'] . ' , isAdmin ' . $data['isAdmin'] . '<br />';
    }
    $answer->closeCursor();
?>
```

- *Explain, line by line, this code, in your report.*

`$pdo_options[PDO::ATTR_ERRMODE] = PDO::ERRMODE_EXCEPTION;`

it's the way to monitor the error signal : an alert or an exception (here, it's an exception)

the variable `$pdo_options` is an array of `PDO::ATTR_ERRMODE`

`$bdd = new PDO('mysql:host=sql-users.ece.fr;dbname=DBNAME;port=3305', 'username', 'password', $pdo_options);`

it's the declaration of a PDO object connected to a certain database. We will then use `$bdd` in order to do the requested queries

`$answer = $bdd->query('SELECT pseudo, isAdmin FROM Member');`

We store the result of the query into the variable "answer"

```
while ($data = $answer->fetch())
{
    echo $data['pseudo'] . ' , isAdmin ' . $data['isAdmin'] . '<br />';
}
```

`fetch` returns the next line in the query result. So while we haven't reach the last line of the query result, we echo the pseudo and the situation (isAdmin or not)

`$answer->closeCursor();`

It's the means to free the server, and permit other queries to be executed

`fetchObject` -> return the next line as an object

`fetchAll` -> return an array of the result rows

6.1. Database : select & insert

```
<?php

$pdo_options[PDO::ATTR_ERRMODE] = PDO::ERRMODE_EXCEPTION;
$bdd = new PDO('mysql:host=sql-users.ece.fr;dbname=zysman;port=3305', 'zysman-rw', $pdo_options);

$bdd->exec("INSERT INTO `update`(`idUpdate`,`content`,`date`,`service`,`idMember`)
VALUES ('1','mollis nec, cursus nim.','1271934352','flickr','34')");
echo 'Inserted';
?>
```

- *What does the method do?*

The first 3 lines of the method are the same as above, it is the establishment of the connection between the page and the SQL server.

exec specifies the execution of a command in between brackets, in our case the INSERT command.

- *What does exec return?*

After hours of searching, it is not possible to access the school database from outside! Therefore we used a local connection. **It seems impossible to have a connection through MAMP using the PDO**, even though the syntax is correct and tested into phpmyadmin, the query below adds the update in the data base!

**INSERT INTO updates (idUpdate,content,date,service,idMember)
VALUES ('1','mollis nec, cursus nim.','1271934352','flickr','34');**

```
<?php

$pdo_options[PDO::ATTR_ERRMODE] = PDO::ERRMODE_EXCEPTION;
$bdd = new PDO('mysql:127.0.0.1;dbname=web;port=3305', 'root', 'root', $pdo_options);

$bdd->exec("INSERT INTO updates (idUpdate,content,date,service,idMember)
VALUES ('1','mollis nec, cursus nim.','1271934352','flickr','34')");

echo 'Inserted';
?>
```

At the beginning, we decided to use the PDO function proposed in the lesson, unfortunately, several services don't recognize such a function : the Free sql service, the school one and Mamp let users know that PDO class doesn't exist. To tackle this issues, we use another class called mysqli_connect which is also used to connect a php file to a database, and the same error appeared :

Fatal error: Class 'mysqli_connect' not found in
**/Users/thibault/Documents/Documents/ECE/WebTechnology/TechnoWebRepo/test
 Mamp.php on line 4**

6.2. Database

- *What is the query to delete all Members with isAdmin equals to 1 ?*

After connecting to the database, we use the following syntax to insert a row in a database :

```
$bdd->exec (DELETE * FROM member WHERE isAdmin ='1');
```

- *What is the code to delete rows with an empty email ?*

```
$bdd->exec (DELETE FROM member WHERE email="");
```

- *The query to change the name of all female patient ?*

```
$bdd->exec (UPDATE member SET pseudo = 'bitch' WHERE sex = 'F');
```

- *The query to change the password of people registered after the 1st january 2013 ?*

```
$bdd->exec (UPDATE member LEFT JOIN updates ON member.idMember=updates.idMember SET password='newpassword' WHERE date >= '2013-01-01');
```

- 7. Member management**
- 8. XML / RSS**
- 9. Post exam & javascript**
- 10. Ajax & API**
- 11. File management**