

Langage PHP

*Accès aux bases de données
avec PDO*

Plan

2

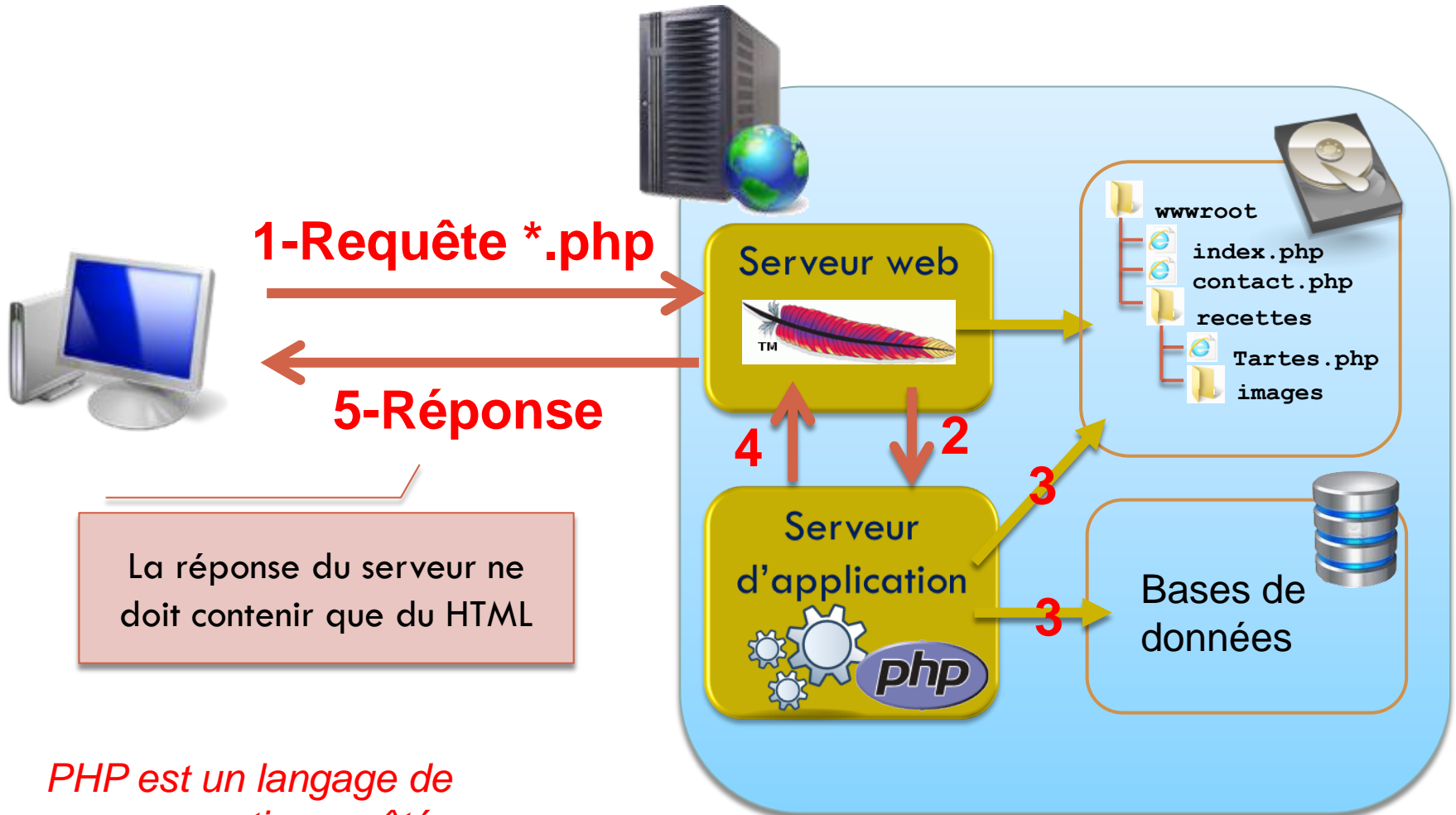
- Base de données
- Connexion aux bases de données
- Lire des données
- Requêtes HTTP avec données
- Les requêtes préparées

Base de données

idTemp	temp	date	idBassin
1	11.3	2015-03-21 08:05:20	1
2	11.7	2015-03-21 12:00:00	1
3	11.8	2015-03-21 16:00:00	1
4	9.8	2015-03-21 04:00:00	2
5	10.1	2015-03-21 08:00:00	2
6	13.5	2015-03-21 12:00:00	3

Site web dynamique PHP

4



PHP est un langage de programmation « côté serveur »

Architecture de WampServer

5



Adresse du site web
http://localhost/truites/

Alias



WampServer

Serveur web



Apache

Alias : **truites**



D:/temp/web/Pisciculture3/

Serveur
d'application



Utilisateur mysql : **bts / snir**

SGBDR

MySQL



Base de données
bddtruites

Installer la base de données



6

- Utilisez Phpmyadmin pour importer le fichier **bbtruites.sql**
 - ▣ Clic gauche sur la tray icône de wampserver
 > phpMyAdmin
 - ▣ Page d'accueil de phpmyadmin > Importer
 > Choisissez un fichier : bddtruites.sql
 - ▣ Exécuter

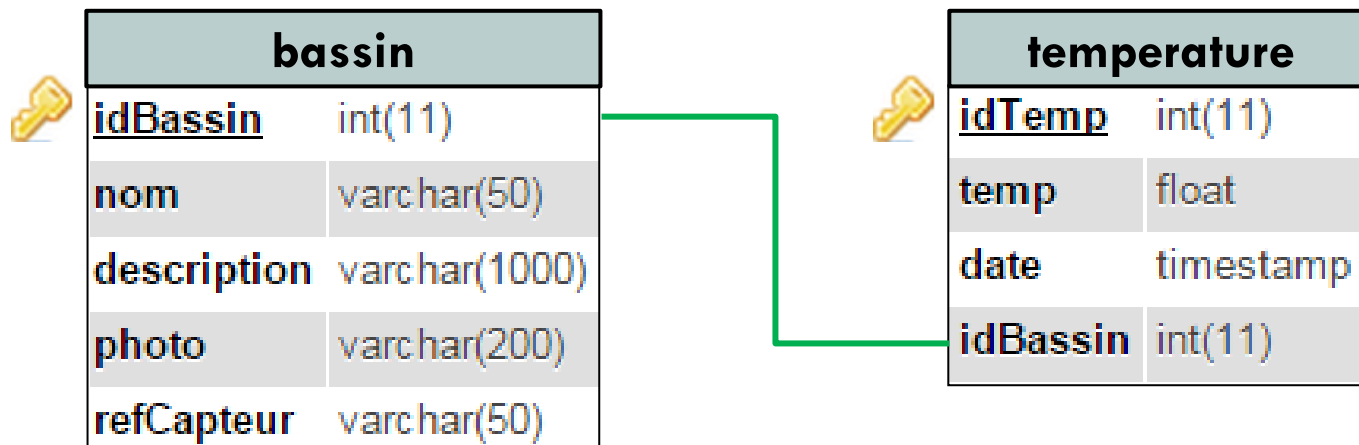
Structure de la base de données

7

□ Schéma conceptuel



□ Schéma relationnel



Les données

8

□ Table « bassin »

idBassin	nom	description	photo	refCapteur
1	Bassin du Hem	20 hectares dans le centre d'Armentières.	bassins1.jpg	E4FD
2	Bassin du Héron	15 hectares au centre de Villeneuve-d'Ascq.	bassins2.jpg	842A
3	Bassin de l'arc en ciel	5 hectares au coeur des flandres.	bassins3.jpg	850E

□ Table « temperature »

idTemp	temp	date	idBassin
1	11.3	2015-03-21 08:05:20	1
2	11.7	2015-03-21 12:00:00	1
3	11.8	2015-03-21 16:00:00	1
4	9.8	2015-03-21 04:00:00	2
5	10.1	2015-03-21 08:00:00	2
6	13.5	2015-03-21 12:00:00	3



3 étapes pour accéder à MySQL

10



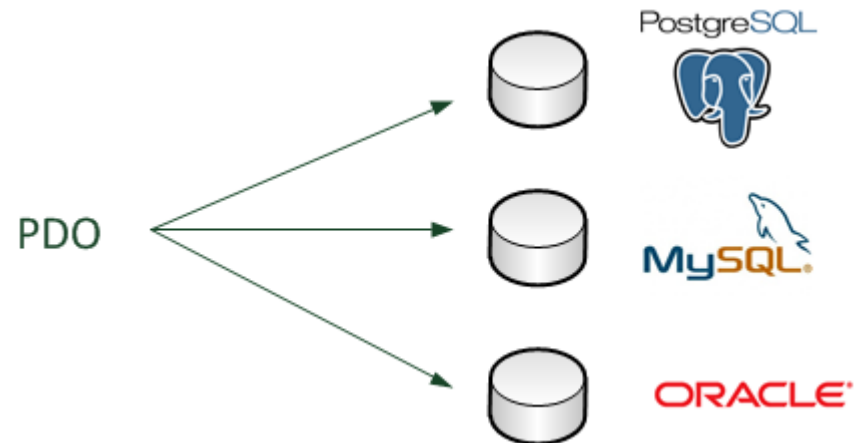
- **SQL** (*Structured Query Language*) est un langage de requêtage standard pour les SGBDR.
- 4 fonctions pour manipuler les données
 - ▣ Insérer **INSERT**
 - ▣ Lire **SELECT**
 - ▣ Mettre à jour **UPDATE**
 - ▣ Supprimer **DELETE**

CRUD
Create
Read
Update
Delete

PDO

11

- Depuis la version 5.1, PHP propose l'extension **PDO (*PHP Data Objects*)**
- C'est un outil complet qui permet d'accéder à n'importe quel type de base de données.
- On peut donc l'utiliser pour se connecter aussi bien à MySQL que PostgreSQL ou Oracle



Pour se connecter à MySQL

12

- Il faut donner 4 informations :
 - ▣ Le nom d'hôte : adresse IP de la machine hébergeant MySQL : Comme PHP et MySQL s'exécutent sur la même machine c'est « **localhost** »
 - ▣ Le nom de la base de données : « **bddtruites** »
 - ▣ Le nom d'utilisateur MySQL : « **bts** »
 - ▣ Son mot de passe : « **snir** »

Connexion et déconnexion

13

□ Connexion à MySQL

```
$objBdd = new PDO("mysql:host=$bddserver;  
                  dbname=$bddname;  
                  charset=utf8",  
                  $bddlogin, $bddpass);
```

□ Déconnexion

```
$objBdd = null;
```

Si vous ne le faites pas explicitement, PHP fermera automatiquement la connexion lorsque le script arrivera à la fin.

Gestion des erreurs de connexion

14

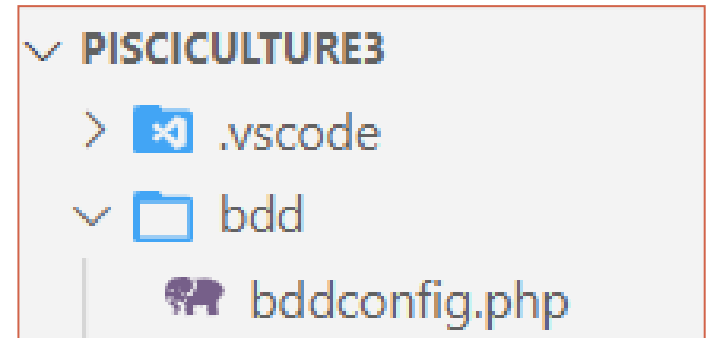
```
try {  
    $objBdd = new PDO("mysql:host=$bddserver;  
                      dbname=$bddname;  
                      charset=utf8",  
                      $bddlogin, $bddpass);  
  
    $objBdd->setAttribute(PDO::ATTR_ERRMODE,  
                          PDO::ERRMODE_EXCEPTION);  
  
} catch (Exception $prme) {  
    die('Erreur : ' . $prme->getMessage());  
}
```

Connexion à la base



15

- Créez un fichier **bddconfig.php** qui contiendra les variables de connexion
- Codez une connexion à la base de données dans la page **bassins.php**
- Dans Phpmyadmin, vérifiez que la connexion est bien établie / supprimée (VScode en debug pas à pas)
 - ▣ Phpmyadmin > Etat > Serveur > Processus



Les bassins

Bassin du Hem

20 hectares dans le centre d'Armentières.



Lire des données avec fetch

17


Envoyer la requête SQL

```
<?php
$bassins = $objBdd->query("select nom from bassin");
?>
```

Afficher les résultats ligne par ligne

```
<?php
while ($bassin = $bassins->fetch()) {
    <h2><?php echo $bassin['nom']; ?></h2>

    <?php
} //fin du while
$bassins->closeCursor(); //libère les ressources de la bdd
?>
```



Lire des données avec foreach

18


Envoyer la requête SQL

```
<?php
$bassins = $objBdd->query("select nom from bassin");
?>
```

Afficher les résultats ligne par ligne

```
<?php
foreach ($bassins as $bassin) {
    <h2><?php echo $bassin['nom']; ?></h2>

    <?php
} //fin du foreach
$bassins->closeCursor(); //libère les ressources de la bdd
?>
```





La page bassins.php

19

- Affichez les données des bassins en les récupérant depuis la base de données

Les bassins

Bassin du Hem

20 hectares dans le centre d'Armentières.



Requêtes HTTP avec données

Bassin du Hem

20 hectares dans le centre d'Armentières.



[Voir les températures](#)

Les températures du Bassin du Hem

Date	Température (°C)
21/03/2015 8:00:00	10.4
21/03/2015 12:00:00	11.1

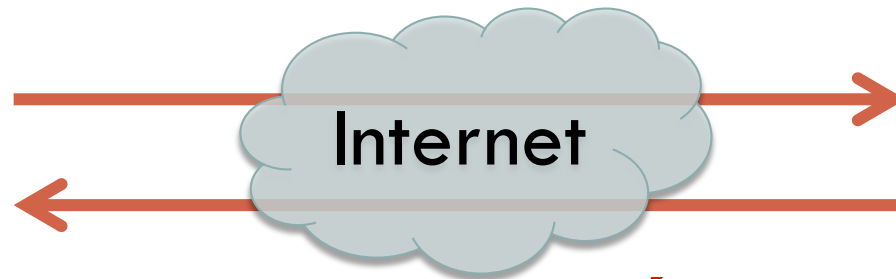


Requêtes HTTP avec données

21

Requête HTTP +

```
idbassin=1  
nombassin=Bassin du Hem
```



Réponse HTTP

- Les requêtes HTTP peuvent transporter des données suivant plusieurs méthodes
 - ▣ GET, POST, PUT et DELETE

HTTP

TCP:80

IP

Accès réseau

Requête HTTP GET



Requête HTTP



Les données apparaissent dans l'URL

?idbassin=1&nombassin=Bassin%20du%20Hem

Nom de la page web désirée suivi des données

Méthode de
requêtage

Requête GET

En-tête

~~Corps
(optionnel)~~

```
GET /truites/temperatures.php?idbassin=1&nombassin=Bassin%20du%20Hem HTTP/1.1
Host: localhost:82
Connection: keep-alive
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko)
Referer: http://localhost:82/truites/bassins.php
Accept-Encoding: gzip, deflate, sdch
Accept-Language: fr-FR;q=0.8,en-US;q=0.6,en;q=0.4
```

Requête HTTP POST



Requête HTTP



Méthode de
requête

Nom de la page web désirée

Requête POST

En-tête

Corps

POST /truites/temperatures.php HTTP/1.1

Host: localhost:82

Connection: keep-alive

Content-Length: 34

Cache-Control: max-age=0

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;

Origin: http://localhost:82

User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/41.0.2226.150 Safari/537.36

Content-Type: application/x-www-form-urlencoded

Referer: http://localhost:82/truites/test.html

Accept-Encoding: gzip, deflate

idbassin=1&nombassin=Bassin+du+Hem

Les données sont passées dans le
corps de la requête

Création d'un lien <a> avec données

24

- En utilisant la méthode GET

Toujours un ? après le nom de la page



```
<a href="temperatures.php?idbassin=1&nombassin=Hem">  
Voir les températures  
</a>
```



Les paires nom=valeur sont séparées par un &

Récupérer les données d'URL

25

- PHP fournit un tableau associatif des valeurs passées via les paramètres d'URL

```
$_GET["nombassin"]
```

- Pour éviter les erreurs éventuelles :

`isset()` vérifie si la variable existe

```
if (isset($_GET["nombassin"])) {  
    $nomBassin = strval($_GET["nombassin"]);  
}
```

`strval()` convertit la variable en String

pour convertir en entier : `intval()`

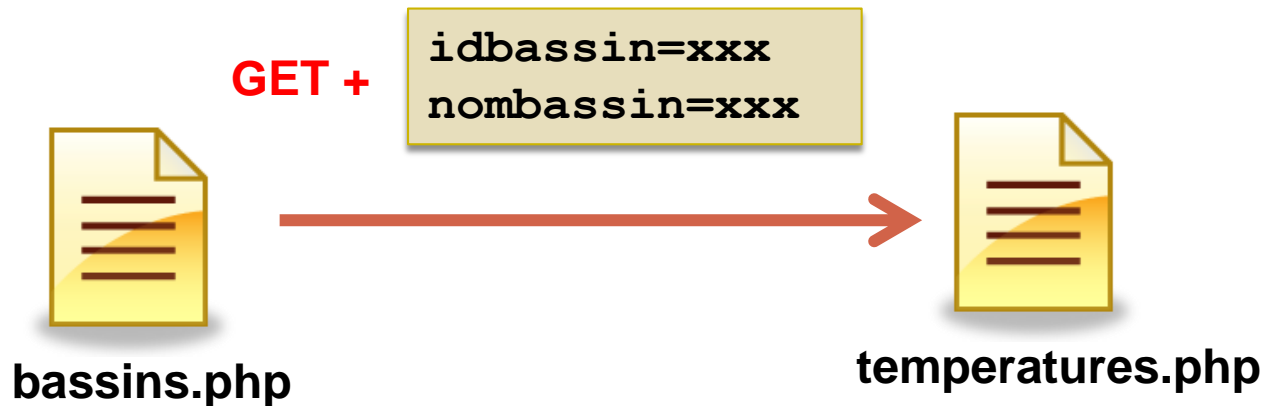
Liens vers la page temperatures.php



26

- Dans la page **bassins.php**, créez les liens vers la page **temperatures.php** en passant les données suivantes :

- ▣ idbassin
- ▣ nombassin



- Créez la nouvelle page **temperatures.php** sur le modèle des autres et affichez le nom du bassin grâce à la variable `$_GET`

Les températures du Bassin du Héron

Les requêtes préparées

SQL : critères de sélection

28

- Dès qu'une requête SQL contient un paramètre d'entrée, il faut utiliser une technique pour se prémunir des injections de code

```
"select * from temperature where idBassin = $id"
```



paramètre d'entrée

Requête préparée (marqueurs ?)

29

- Avec des marqueurs interrogatifs (?)

```
<?php
$idbassin = intval($_GET["idbassin"]);
$objBdd = new PDO("mysql:host=$bddserver;
                  dbname=$bddname;
                  charset=utf8",
                  $bddlogin, $bddpass);

$temps = $objBdd->prepare("select * from temperature
                           where idBassin = ?");
$temps->execute(array($idbassin));
?>
```

↑
tableau de valeurs (marqueurs)

Requête préparée (marqueurs :nom)

30

- Avec des marqueurs nommés (**:nom**)

```
<?php
$idbassin = intval($_GET["idbassin"]);
$objBdd = new PDO("mysql:host=$bddserver;
                  dbname=$bddname;
                  charset=utf8",
                  $bddlogin, $bddpass);

$temps = $objBdd->prepare("select * from temperature
                           where idBassin = :id");
$temps->execute(array(':id' => $idbassin));
?>
```

↑
tableau de valeurs (paramètres de nom)

Requête préparée (bindParam)

31

- Avec des variables liées (**bindParam**)

```
<?php
$idbassin = intval($_GET["idbassin"]);
$objBdd = new PDO("mysql:host=$bddserver;
                  dbname=$bddname;
                  charset=utf8",
                  $bddlogin, $bddpass);

$stmts = $objBdd->prepare("select * from temperature
                           where idBassin = :id");
$stmts->bindParam(':id', $idbassin, PDO::PARAM_INT);
$stmts->execute();?>
```

Vérification du type du paramètre



Afficher les températures

32

- Récupérez le paramètre d'URL **idbassin**
- Affichez les températures du bassin reçu en paramètre en les récupérant depuis la base de données (utilisez des marqueurs nommés)

Les températures du Bassin de l'arc en ciel

Date	Température (°C)
2015-03-21 16:00:00	10.6
2015-03-21 12:00:00	10.5
2015-03-21 08:00:00	10.3
2015-03-21 04:00:00	9.8

Les tableaux HTML5

33

Article	Prix
Nexus 5	349
Nexus 7	229
Total	578

<thead>

<tbody>

<tfoot>

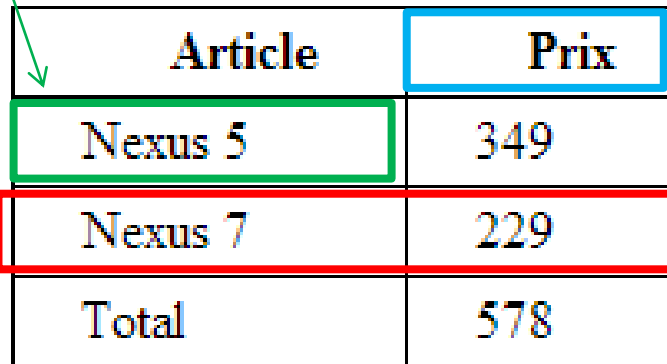
```
<table>
  <thead>
    <tr>
      <th>Article</th>
      <th>Prix</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Nexus 5</td>
      <td>349</td>
    </tr>
    <tr>
      <td>Nexus 7</td>
      <td>229</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>Total</td>
      <td>578</td>
    </tr>
  </tfoot>
</table>
```

Les tableaux HTML5

34

TD (cellule)

TH (cellule d'en-tête)



Article	Prix
Nexus 5	349
Nexus 7	229
Total	578

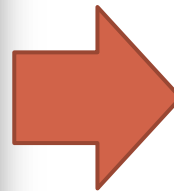
TR (ligne)

```
<table>
  <thead>
    <tr>
      <th>Article</th>
      <th>Prix</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Nexus 5</td>
      <td>349</td>
    </tr>
    <tr>
      <td>Nexus 7</td>
      <td>229</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>Total</td>
      <td>578</td>
    </tr>
  </tfoot>
</table>
```

CSS des tableaux HTML5

35

```
table
{
    border-collapse: collapse;
    width: 200px;
}
th,td
{
    border:1px solid black;
    padding: 3px 0px 3px 20px;
}
thead
{
    background-color: red;
    color:white;
}
tfoot
{
    background-color: silver;
    color:red;
}
```



Article	Prix
Nexus 5	349
Nexus 7	229
Total	578



Améliorations

36

- Affichez les températures en ordre anti chronologique

```
"select * from temperature where idBassin = :id  
order by date desc"
```

- Affichez les 3 dernières températures

```
"select * from temperature where idBassin = :id  
order by date desc  
limit 0,3"
```

3 valeurs en commençant à 0

Documentation

37

- <http://php.net/>
- <https://www.w3schools.com/php/DEFAULT.asp>

