

NOM :	CLASSE :
PRÉNOM :	

Condition :	<ul style="list-style-type: none"> • Travail en binôme; durée 4 heures
Matériel :	<ul style="list-style-type: none"> • un ordinateur sous Ubuntu avec LAMP installé • un Arduino Uno, un shield ethernet, un capteur de température TMP36
Documents :	<ul style="list-style-type: none"> • le sujet du TP • le cours sur les bases de données

L'objectif du TP est de découvrir l'utilisation d'une base de données.

1. Installation de phpMyAdmin

Depuis 2023 phpMyAdmin est déjà installé sur les postes.

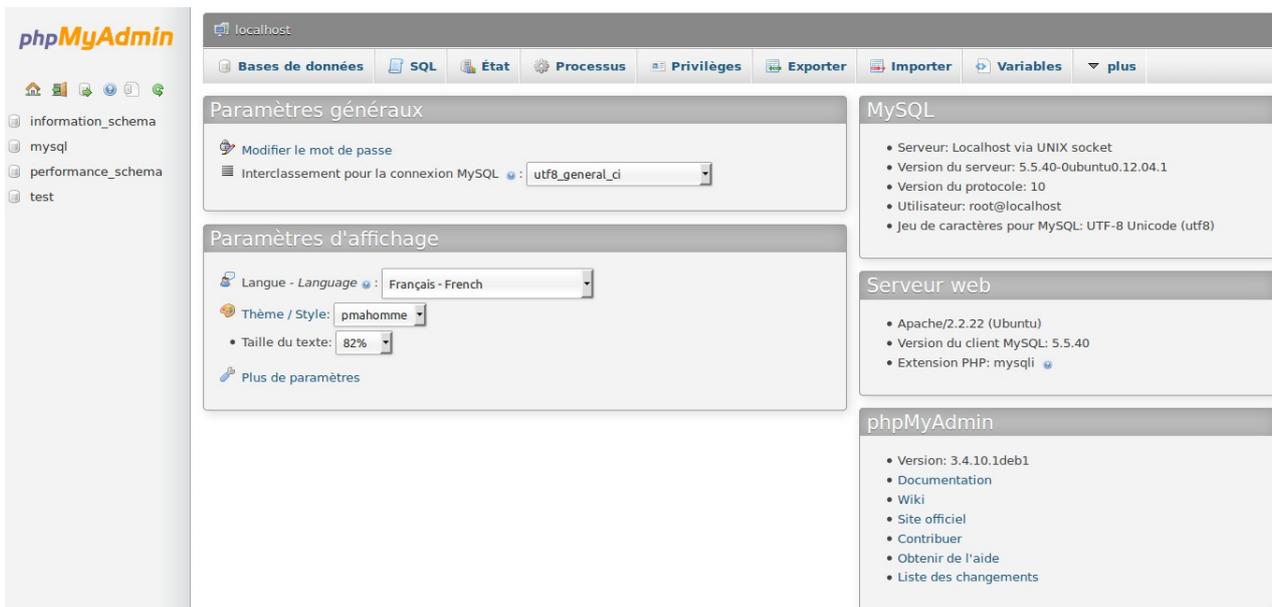
✓ Démarrez Firefox et allez à l'url *localhost/phpmyadmin/*. Connectez-vous à la base de données en utilisant le compte utilisateur admin mot de passe *sti2dsin*. Et rendez vous à la partie 2 Du TP

Pour créer nos bases de données nous utiliserons **phpMyAdmin**, un des outils les plus connus permettant de manipuler une base de données MySQL.

✓ Installez phpMyadmin en tapant sous la console la commande :

```
sudo apt-get install phpmyadmin
```

- ✓ Avec la barre espace, cochez la case *Apache 2* comme serveur web à reconfigurer.
- ✓ Répondez OUI à la question « faut-il configurer la base de données de phpmyadmin ? »
- ✓ Entrez le mot de passe *sti2dsin*.
- ✓ Entrez à nouveau le mot de passe *sti2dsin* (2 fois) pour la connexion à phpmyadmin.
- ✓ Démarrez Firefox et allez à l'url *localhost/phpmyadmin/*. Connectez-vous à la base de données en utilisant le compte utilisateur *root*, mot de passe *sti2dsin*. L'écran suivant apparaît :



2. Construire une table dans la base de données

Vous allez créer une base de données *sti2d*, composée d'une table *maTable* constituée de 5 champs, afin d'y enregistrer les données du formulaire vu dans le TP précédent.

id	nom	prenom	sexe	an_naissance

- ✓ Cliquez sur l'onglet Bases de données puis créez une nouvelle base *sti2d* :



- ✓ Dans la partie gauche de l'écran, sélectionnez maintenant cette nouvelle base *sti2d*. Puis créez une nouvelle table *maTable* composée de 5 champs (5 colonnes) :



- ✓ Créez alors les 5 champs comme ci-dessous puis cliquez sur en bas de la fenêtre sur *Sauvegarder*.

Nom de la table:
maTable

Colonne	Type	Taille/Valeurs*1
id	INT	6
nom	VARCHAR	20
prenom	VARCHAR	20
sexe	CHAR	1
an_naissance	SMALLINT	4

cochez pour cette ligne la case AI (Auto-incrémentation)

Type des variables :

- **int** (4 octets) : entier
- **varchar** : chaîne de caractères (de 0 à 65535 caractères)
- **char** : chaîne de caractères courte (de 0 à 255 caractères)
- **smallint** (2 octets) : entier de -32768 à +32767

Voilà, la table *maTable* est prête. Il est possible d'écrire, lire ou modifier des données à partir de **requêtes écrites en langage SQL**.

- ✓ Cliquez sur l'onglet *SQL* puis sur *INSERT* et tapez la ligne ci-dessous. Cliquez alors sur l'onglet *Afficher* et vérifiez que vous venez bien d'insérer une première entrée dans la table.

```
INSERT INTO `maTable`(`id`, `nom`, `prenom`, `sexe`, `an_naissance`) VALUES (1,"Dupond","pierre","M",1994)
```

Nous allons, dans le chapitre suivant, voir qu'il est possible d'écrire dans la table à partir d'un programme en PHP.

3. Écrire dans une base de données en PHP

Vous allez réutiliser le formulaire du TP PHP. On vous donne ci-dessous le fichier index.html.

- ✓ Enregistrez ce fichier dans le dossier /var/www de votre poste.

Nom:

Prénom :

Sexe : Masculin Féminin

Année de naissance :

index.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Mon super site</title>
  </head>
  <body>
    <form action="enregistrement.php" method="post">
      Nom:
      <input type="text" name="nom" size="20"/><br/>
      Prénom :
      <input type="text" name="prenom" size="20"/><br/>
      Sexe :
      <input type="radio" name="sexe" value="M" /> Masculin
      <input type="radio" name="sexe" value="F" /> Féminin
      <br/>
      Année de naissance :
      <select name="annee">
        <option value="1992">1992</option>
        <option value="1993">1993</option>
        <option value="1994">1994</option>
        <option value="1995">1995</option>
        <option value="1996">1996</option>
        <option value="1997">1997</option>
        <option value="1998">1998</option>
        <option value="1999">1999</option>
      </select>
      <br/>
      <input type="submit" value="Envoyer"/><br/>
    </form>
  </body>
</html>
```

Le formulaire appelle un fichier enregistrement.php dont le code est donné ci-dessous. **Un autre fichier identifiants.php contient les identifiants de connexion à la base de données, ce qui garantit une meilleure sécurité.**

- ✓ Enregistrez ces deux fichiers dans le dossier /var/www de votre poste.

enregistrement.php

```
<?php
/* récupération des données du formulaire */
$nom = $_POST['nom'];
$prenom = $_POST['prenom'];
$sexe = $_POST['sexe'];
$an_naissance = $_POST['annee'];

include ('identifiants.php');

try
{
  //Connexion
  $conn = new PDO("mysql:host=$servername;dbname=$dbname;charset=utf8", $username,
$password);
  echo "Base connectée <br>";
}
catch (Exception $e)
{
  die('Erreur : ' . $e->getMessage());
  echo "Erreur";
}
```

```

}
//Ajouter une entrée dans la table
$sql= "INSERT INTO Eleve (nom,prenom,sexe,an_naissance) VALUES
('$nom','$prenom','$sexe','$an_naissance)";
$conn->exec($sql);
echo "Enregistrement fait <br>";
//Deconnexion à la bdd
$conn = null;
?>
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Enregistrement</title>
  </head>
  <body>
    <p>Pour revenir à la page formulaire, <a href="index.html">cliquez ici</a></p>
  </body>
</html>

```

identifiants.php

```

<?php
/* identifiants de connexion */
$host = "localhost";
$user = "admin";
$pass = "sti2dsin";
$dbd = "sti2d";
?>

```

- ✓ Ouvrez le formulaire dans Firefox en tapant l'adresse 127.0.0.1. Remplissez le formulaire puis cliquez sur Envoyer. Allez ensuite voir dans votre base de données à l'aide de phpMyadmin que les valeurs entrées ont bien été ajoutées à la table maTable de la bdd sti2d.
- ✓ Quel est le rôle de la ligne \$nom = \$_POST['nom'] ; ?
- ✓ Quelle ligne permet de se connecter à la bdd ?
- ✓ Quel est le rôle de la requête "INSERT INTO maTable (nom,prenom,sexe,an_naissance) VALUES ('\$nom', '\$prenom', '\$sexe', '\$an_naissance)'" ?

4. Lire des éléments d'une base de données en PHP

Vous allez maintenant lire et afficher le contenu de la table dans une page web. On vous donne ci-dessous le fichier lecture.php.

- ✓ Enregistrez ce fichier dans le dossier /var/www de votre poste. Ouvrez ce fichier dans Firefox et vérifiez son bon fonctionnement.

lecture.php

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>lecture de la bdd</title>
  </head>
  <body>

<?php

  include ('identifiants.php');

  try
  {
    //Connexion à la base de données
    $conn = new PDO("mysql:host=$servername;dbname=$dbname;charset=utf8", $username,
$password);
    echo "Connexion lecture <br>";
  }
  catch (Exception $e)
  {
    die('Erreur : ' . $e->getMessage());
    echo "Erreur";
  }
  //Lire le contenu de la table
  $sql = "SELECT * FROM Eleve LIMIT 0,30"; //requête sql
  $reponse = $conn->query($sql);
  $nbLigne = $reponse->rowCount(); //nombre de lignes de la réponse
  print("La table contient $nbLigne élèves.<br>");
  echo "les $nbLigne élèves sont : <br>";
  while($ligne = $reponse->fetch()){
    echo $ligne['nom']." ".$ligne['prenom']."<br>";
  }
  $reponse->closeCursor(); //Termine le traitement de cette requête

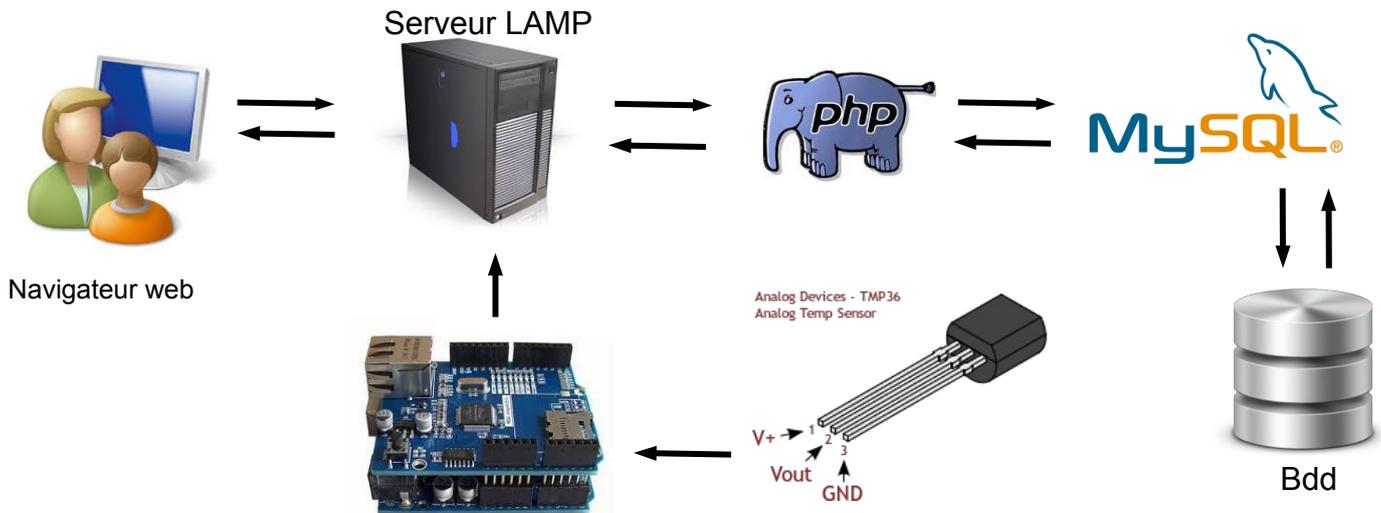
  $reponse = null;
  $conn = null; //fermeture de la connexion à la bdd
?>
</body>
</html>
```

- ✓ Quel est le rôle de la ligne `$query = "SELECT * FROM maTable ";` ?
- ✓ Quel est le rôle de la ligne `while($ligne = $reponse->fetch())` ?
- ✓ Modifiez le programme afin d'afficher à l'écran tous les champs de la table Eleve. Testez et faites valider le bon fonctionnement par le professeur.

Validation prof :

5. Arduino et base de données

L'objectif de ce chapitre est d'effectuer des mesures de température, à l'aide d'un Arduino et d'un capteur de température TMP36, toutes les 20 secondes, et d'enregistrer les mesures dans une base de données.



Pour ce faire, vous devez :

- connecter un shield ethernet sur un Arduino UNO ;
- câbler un capteur de température TMP36 sur la broche A0 de l'Arduino ;
- modifier le programme Arduino donné ci-dessous et programmer l'Arduino ;
- créer une table temp dans la base sti2d de votre poste avec 4 champs : id, date, heure, temperature ;
- écrire un programme enregistrement.php qui reçoit les données de l'Arduino et les enregistre dans la table temp de la base de données sti2d. Ce programme récupère aussi la date et l'heure du PC afin de les envoyer avec chaque valeur de température ;
- écrire une page web lecture.php qui se connecte à cette base et affiche les mesures.

ArduinoTemp.ino

```

/*
  ArduinoTemp.ino
  Web client

  Ce programme se connecte à un serveur web
  et transmet en paramètre la valeur de la température
  relevée sur un capteur TMP36 connecté à A0

```

Circuit:

- * Ethernet shield sur broches 10, 11, 12, 13
- * TMP36 sur broche A0

22 octobre 2014
Christophe Le Bris

*/

```

#include <SPI.h>
#include <Ethernet.h>

```

```

// Enter a MAC address for your controller below.
// Newer Ethernet shields have a MAC address printed on a sticker on the shield
byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };

```

```

IPAddress server(192,168,11,42); // @IP du serveur Web
IPAddress ip(192,168,11,221); // @IP du shield ethernet sur la carte Arduino

```

```

// Initialize the Ethernet client library
// with the IP address and port of the server
// that you want to connect to (port 80 is default for HTTP):
EthernetClient client;

```

```

void setup() {

```

à modifier : prendre l'@
MAC de votre shield

à modifier : le serveur web
est votre PC

à modifier : @IP que vous
attribuez à votre shield, voir
tableau sur le réseau

```

// start the serial library:
Serial.begin(9600);
pinMode(A0,INPUT);
// give the ethernet module time to boot up:
delay(1000);
// start the Ethernet connection using a fixed IP address and DNS server:
Ethernet.begin(mac, ip);
// print the Ethernet board/shield's IP address:
Serial.print("My IP address: ");
Serial.println(Ethernet.localIP());
}

void loop()
{
  httpRequest();
  delay(20000);
}

// this method makes a HTTP connection to the server
void httpRequest() {
  float temp=analogRead(A0);
  if (client.connect(server, 80))
  {
    Serial.println("Connecting...");
    Serial.print("la temperature est de : ");
    Serial.println(temp);
    // send the HTTP PUT request:
    client.print("GET http://192.168.11.42/enregistrement.php?temp="+); /*localhost à changer par l'@ IP
du serveur */
    client.print(temp);
    client.println(" HTTP/1.0" );
    client.println();
    Serial.println("fin de la requete...");
    client.stop();
  }
  else {
    // if you couldn't make a connection:
    Serial.println("connection failed");
    client.stop();
  }
}
}

```

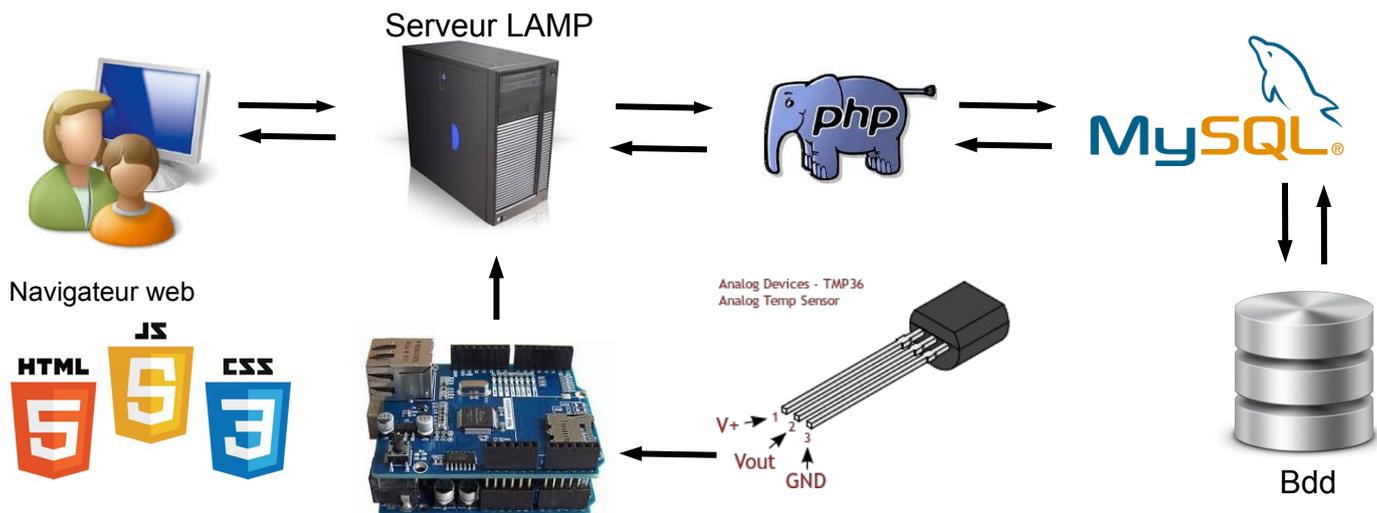
à modifier : calcul de la température

à modifier : le serveur web est votre PC

- ✓ Utilisez les travaux effectués dans ce TP pour écrire les différents programmes. Testez et faites valider le bon fonctionnement par le professeur.

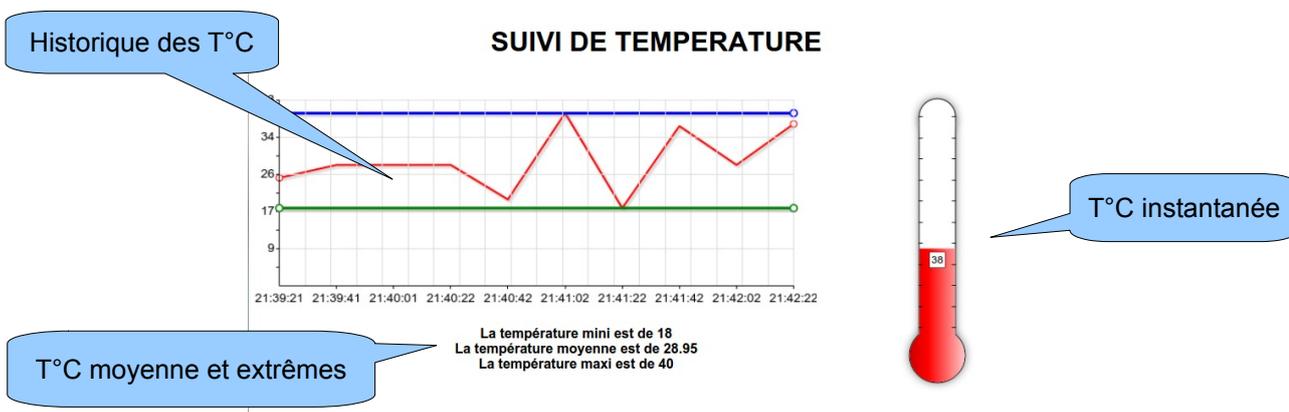
Validation prof :

6. Présenter les données en PHP et Javascript



L'objectif est de présenter les données récoltées dans la base de données sous une forme synthétique et graphique interprétable par tous navigateurs . On distingue trois zones d'information :

- la température instantanée (thermomètre) ,
- l'historique de l'évolution de la température (graphique)
- un calcul de la moyenne et des extrêmes des températures récoltées.



6.1 Calculs sur la base de données

Pour afficher les valeurs moyenne et extrêmes, trois requêtes vers la base de données doivent être formulées à partir d'un fichier PHP, puis elles seront affichées.

index.php

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Calcul sur la bdd</title>
  </head>
  <body>

<?php
  /* Calcul sur la bdd */
  include ('identifiants.php');

  try
  {
    //Connexion à la base de données
    $conn = new PDO("mysql:host=$servername;dbname=$dbname;charset=utf8", $username,
$password);
    echo "Connexion lecture <br>";
  }
  catch (Exception $e)
  {
```

```

die("Erreur : ". $e->getMessage());
    echo "Erreur";
}
//Lire le contenu de la table
$sql = "SELECT * FROM temp LIMIT 0,30"; //requête sql
$reponse = $conn->query($sql);
$nbLigne = $reponse->rowCount(); //nombre de lignes de la réponse
print("La table contient $nbLigne mesures.<br>");

echo "les mesures sont : <br>";
while($ligne = $reponse->fetch()){ //Pour toutes les lignes de la réponse
    echo $ligne['temperature']. "° à ".$ligne['heure']. " le ".$ligne['date']."<br>";
}
$reponse->closeCursor(); //Termine le traitement de cette requête
//Récupération des moyenne, maxi, mini de la colonne température
$sql = "SELECT AVG(temperature) AS temp_moy FROM temp ";
$reponse = $conn->query($sql);
$ligne = $reponse->fetch();
$temp_moy = $ligne['temp_moy'];
echo 'la température moyenne est de '.$temp_moy.'°C. <br />';
$reponse->closeCursor();

$sql = "SELECT MAX(temperature) AS temp_max FROM temp ";
$reponse = $conn->query($sql);
$ligne = $reponse->fetch();
$temp_max = $ligne['temp_max'];
echo 'la température maximum est de '.$temp_max.'°C. <br />';
$reponse->closeCursor();

$reponse = null;
$conn = null; //fermeture de la connexion à la bdd
?>
</body>
</html>

```

vérifier les noms de ces champs dans votre table

champ à vérifier

modifier par le nom de votre table

champ à vérifier

modifier par le nom de votre table

- ✓ Identifier la requête mysql nécessaire pour obtenir la valeur moyenne :

- ✓ Indiquer dans quelle variable PHP est stockée cette donnée :

- ✓ Identifier la requête mysql nécessaire pour obtenir la valeur max :

- ✓ Indiquer dans quelle variable PHP est stockée cette donnée :

- ✓ Recopier et modifier ce code en utilisant la base de données connectée avec l'Arduino. Enregistrer-le sous index.php dans var/www. Tester.

- ✓ La requête mysql nécessaire pour obtenir la valeur min est la suivante: **SELECT MIN(`temp`) AS min_temp FROM `maTable`**. Tester cette requête dans la base de donnée en l'adaptant au nom de votre base de donnée. Vérifier le résultat.

- ✓ Écrire dans votre fichier index.php, le code permettant de récupérer et d'afficher la valeur min. Vous stockerez cette valeur dans la variable PHP **\$temp_min**.

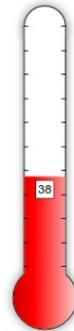
Validation prof :

6.2 Communiquer des données de la base de données au Javascript par le PHP

JavaScript est un langage de programmation de scripts principalement employé dans les pages web interactives mais aussi pour les serveurs.

Du code JavaScript peut être intégré directement au sein des pages web, pour y être exécuté sur le poste client. C'est alors le navigateur Web qui prend en charge l'exécution de ces programmes appelés scripts.

Généralement, JavaScript sert à contrôler les données saisies dans des formulaires HTML, ou à interagir avec le document HTML via l'interface Document Object Model, fournie par le navigateur (on parle alors parfois de HTML dynamique ou DHTML). Il est aussi utilisé pour réaliser des services dynamiques, parfois futiles, strictement cosmétiques ou à des fins ergonomiques.



- ✓ Télécharger sur le réseau la bibliothèque *RGRAPH* contenant entre autre la représentation du thermomètre.
- ✓ Créer un dossier *libraries* dans *var/www*. Décompresser dans ce dossier uniquement les trois fichiers ci-dessous, contenant la représentation graphique et les effets dynamiques du thermomètre (remplissage et affichage des valeurs).
 - Rgraph.common.core.js : noyau mathématiques de la bibliothèque
 - Rgraph.common.effects.js : effets dynamiques utilisés
 - Rgraph.thermometer.js : aspect graphique du thermomètre

Note : Le répertoire demos de la bibliothèque RGraph permet de visualiser l'ensemble des jauges graphes, diagrammes, Gantt et voyants disponibles... Rgraph/default pour une vision d'ensemble

On donne ci-dessous le programme partiel *index.php* que vous fusionnerez avec le programme précédent :

index.php

```
<!DOCTYPE html>
<html>
  <head>
    <META HTTP-EQUIV="Refresh" CONTENT="30; URL=index.php">
    <link rel="stylesheet" href="demo.css" type="text/css" media="screen" />
    <meta charset="utf-8" />
    <title>lecture de la température</title>
    <script src="/libraries/RGraph.common.core.js" ></script>
    <script src="/libraries/RGraph.common.effects.js" ></script>
    <script src="/libraries/RGraph.line.js" ></script>
    <script src="/libraries/RGraph.thermometer.js" ></script>
    <script src="http://ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js"></script>
  </head>
  <body>
    <h1> SUIVI DE TEMPERATURE </h1>
    <canvas id="cvs" width="100" height="400">[No canvas support]</canvas>

    <script>
      //thermomètre
      $(document).ready(func = function () {
        /*finalisation du themomètre passage des paramètres variables*/
        var value = <?php echo $temp ; ?>;
        if (typeof thermometer == 'undefined') {
          // A global on purpose
          thermometer = new RGraph.Thermometer({id: 'cvs', min: 0,
max: 100, value: value});
        } else {
          thermometer.value = value;
        }
        thermometer.grow()
        setTimeout(function(){RGraph.clear(thermometer.canvas);func();}, 5000);
      })
    </script>
  </body>
</html>
```

Code permettant d'affecter un positionnement sur la page en lien avec le fichier css et de définir la taille de la fenêtre du thermomètre

début du code javascript

- ✓ Identifier, dans le code javascript, où est réalisé le lien entre la dernière valeur de température lue dans la base de donnée par le code php. Quelle variable php est transformée en variable js ? Sous quelle nom est-elle stockée désormais dans le script js ?
- ✓ A quoi va servir cette variable PHP pour l'obtention du thermomètre ? Identifiez où cette nouvelle variable js est utilisée.
- ✓ Copier le code ci-dessus dans votre fichier index.php. Attention, veillez à insérer la partie « body » à la suite du « body » existant dans le fichier précédent.
- ✓ Code contenu dans le fichier *demo.css* que vous créez dans *var/www*.

```
body { font-family: Arial;}
#cvs {
margin-right:10%;
float: right;
width:10%;
}
h1{ text-align : center;}
```

- ✓ L'échelle actuelle du thermomètre couvre la plage de 0 à 100, modifiez votre programme pour qu'elle aille désormais de 0 à la température maxi +10 (\$temp_maxi +10) que vous avez récupéré dans votre code php.

Validation prof :

6.3 Créer un graphe représentant l'historique des données de la bdd



- ✓ Rajouter à votre répertoire librairies le fichier *Rgraph.line.js* (aspect graphique du thermomètre).
- ✓ Ajouter à la partie php de votre fichier *index.php* le code ci-dessous :

```
$sql= "SELECT heure FROM temp LIMIT 0,100";
$j=0;
$reponse = $conn->query($sql);
while($ligne = $reponse->fetch()){
    $heureinst[$j] = $ligne['heure'];
    $j++;
}
// transforme le tableau en chaîne de string de type
// exemple: '21:39:21',..., '21:42:22'
$xheure="" . implode("","",$heureinst) ."";

$query=

//transforme le tableau en chaîne d'entiers séparés pas des virgules 25,26,,,,,
$temp = implode(",",$tempinst);
```

Ecrire le code permettant de récupérer dans un tableau php les valeurs des températures dans \$tempinst[]

modifier par le nom de votre table

Code permettant de récupérer dans un tableau php les valeurs des heures des relevés de température de la bdd

Transformation des tableaux php en chaîne de caractères compréhensible par js

- ✓ Complétez ce code permettant de récupérer dans un tableau php les valeurs des températures, vous utiliserez la variable \$tempinst[].

- ✓ Identifier le nom de la nouvelle chaîne de caractères contenant les valeurs de températures :
- ✓ Insérer dans le body du code html de votre fichier index.php le code ci-dessous : il s'agit d'une superposition de 3 graphes comprenant une ligne bleu indiquant la température maxi, une rouge pour l'historique des température et une verte pour la température mini.

```

<canvas id="cvs2" width="600" height="250">[No canvas support]</canvas><br />
<script>
  var tempmaxi=new Array ();
  for (var i=0; i<php echo $nbLigne ?&gt;;i++){
    tempmaxi[i]=&lt;?php echo $temp_max ?&gt;;
  }
  $(document).ready(function ()// ouverture de la fonction créant le graphe
  {
    var line = new RGraph.Line({
      id: 'cvs2',
      data: [&lt;?php echo $ytemp; ?&gt;],
      options: {
        ymax:&lt;?php echo $temp_max; ?&gt;+3,
        labels: [&lt;?php echo $xheure; ?&gt;]
      }
    }).draw()
    var line2 = new RGraph.Line({
      id: 'cvs2',
      data: tempmaxi,
      options: {
        colors:['blue'],
        linewidth: 3,
        ylabels: false,
        ymax:&lt;?php echo $temp_max ?&gt;+3,
        labels: []
      }
    }).draw()
  })
&lt;/script&gt;
</pre

Création d'un tableau js contenant autant de fois la température maxi qu'il y a eu de prises de température



Var line est la construction de la ligne rouge du graphique historique des températures. Draw() l'affiche.



Var line2 est la construction de la ligne bleu du graphique température maxi. Draw() l'affiche.


```

- ✓ Écrire le code permettant d'obtenir la ligne verte de la température mini (vous utiliserez les mêmes options de tracé de la ligne de ce graphe que celles utilisées pour la ligne bleu hormis la couleur) .
- ✓ Attention id cvs2 doit être paramétré dans votre fichier demo.css. Complétez le fichier demo.css

```

#cvs2 {
  width:70%;
  float: left;
}
h1 {
  text-align : center;
}
h4 {
  text-align : center;
}
#text {
  float: left;
  width:80%;
}

```

Validation prof :

- ✓ Ce code peut être rajouté en fin de body du fichier *index.php* expliquez ce qu'il fait :

```

<span id="text" style="font-size: 12pt; font-weight: normal">
  <h4><?php
    echo 'La température mini est de '.$temp_min.'<br />';
    echo 'La température moyenne est de '.$temp_moy.'<br />';
    echo 'La température maxi est de '.$temp_max.'<br />';
    ?></h4>
</span>

```