



Lycée Jacquard

CAUDRY

Formation : BTS
Cybersécurité, Informatique et réseaux, Électronique
Informatique et Réseaux

Web dynamique côté serveur
Partie 2



Objectif de ce TP :

- Découvrir le langage php

Activités pratiques :

- Ecriture de page web en php, interprété par le serveur
- Lecture de données venant du client.
- Enregistrement et lecture dans un fichier des données du client,
- Ecriture et lecture de données dans une base

Petit projet de mise en place d'une sortie


Nous allons gérer les informations relatives à un évènement à afficher dans le calendrier. Une sortie a des paramètres bien définie tel que la date, le thème, etc. Les élèves sont également repérés de manière individuel. Voici les 2 classes pour représenter l'ensemble.



Il faut créer les tables dans la base de données relatives à : Sortie et Eleve. Pour ce faire, utilisons phpMyAdmin avec MySQL

Si phpMyAdmin n'est pas installé, faire en ligne de commande, en mode root : apt-get install phpmyadmin
Suivre la procédure d'installation.

L'installation terminée, ouvrir un navigateur web et écrire dans la barre d'adresse : 127.0.0.1/phpmyadmin
Voir ci-dessous :

 127.0.0.1/phpmyadmin/

Projet de création d'une base et de tables

Vous arrivez normalement sur cette page

1 – Adresse entrée précédemment

127.0.0.1/phpmyadmin/

Google

phpMyAdmin

Bienvenue dans phpMyAdmin

Langue - Language

Français - French

Connexion

Utilisateur :

root

Mot de passe :

Exécuter

1 - Ecrire root

2 – Entrer le mot de passe

Projet de création d'une base et de tables

Vous arrivez sur cette page

Menu principal

Nom des bases de données créées précédemment

The screenshot shows the phpMyAdmin interface for a localhost connection. The main menu on the left includes icons for home, SQL, status, processes, privileges, export, and a 'plus' dropdown. Below the menu, a list of databases is shown: DB2_pti, DB_MetierWeb, information_schema, mysql, performance_schema, and phpmyadmin. The 'DB_MetierWeb' database is selected. The main content area displays the 'Paramètres généraux' (General Settings) panel, which includes options to 'Modifier le mot de passe' (Change password) and 'Interclassement pour la connexion MySQL' (Collation for MySQL connection), currently set to 'utf8_general_ci'. Below this is the 'Paramètres d'affichage' (Display Settings) panel, showing 'Langue - Language' set to 'Français - French', 'Thème / Style' set to 'pmahomme', and 'Taille du texte' (Text size) set to '82%'. A 'Plus de paramètres' (More settings) link is also present. On the right side, there are three summary panels: 'MySQL' (Server: Localhost via UNIX socket, Version: 5.5.38-0+wheezy1, Protocol version: 10, User: root@localhost, Character set: UTF-8 Unicode (utf8)), 'Serveur web' (Web server) (Apache/2.2.22 (Debian), MySQL client version: 5.5.38, PHP extension: mysqli), and 'phpMyAdmin' (Version: 3.4.11.1deb2+deb7u1, with links to documentation, wiki, official site, contribute, get help, and changelog).

Projet de création d'une base et de tables

vous arrivez sur cette page

1 - Choisir : Bases de données

Nom des bases de données créées précédemment

The screenshot displays the phpMyAdmin web interface for a local MySQL instance. The 'Bases de données' (Databases) tab is active, showing a list of existing databases on the left: DB2_pti, DB_MetierWeb, information_schema, mysql, performance_schema, and phpmyadmin. The main panel is divided into sections for general and display parameters. The 'Paramètres généraux' (General) section includes options to change the password and the MySQL character set (currently utf8_general_ci). The 'Paramètres d'affichage' (Display) section allows setting the language (Français - French) and theme (pmahomme). On the right, three summary boxes provide system information: MySQL server details (localhost via UNIX socket, version 5.5.38-0+wheezy1, root@localhost user, UTF-8 character set), web server details (Apache/2.2.22 on Debian, MySQL client 5.5.38, PHP extension), and phpMyAdmin version (3.4.11.1deb2+deb7u1) with links to documentation and the official site.

Projet de création d'une base et de tables

Vous arrivez sur cette page

localhost

Bases de données SQL État Processus Privilèges

Bases de données

Créer une base de données

Interclassement

1-Entrer le nom de la base de données : DB_Calendrier

2-Cliquer pour créer la base

Projet de création d'une base et de tables

vous arrivez sur cette page



Bases de données

Créer une base de données

DB_Calendrier

Interclassement

2-Cliquer pour créer la base

3-Cliquer sur la nouvelle base créée

Base de données

<input type="checkbox"/>	DB2_pti	Vérifier les privilèges
<input type="checkbox"/>	DB_Calendrier	Vérifier les privilèges
<input type="checkbox"/>	DB_College	Vérifier les privilèges
<input type="checkbox"/>	DBM_MetierWeb	Vérifier les privilèges
<input type="checkbox"/>	information_schema	Vérifier les privilèges
<input type="checkbox"/>	mysql	Vérifier les privilèges
<input type="checkbox"/>	performance_schema	Vérifier les privilèges
<input type="checkbox"/>	phpmyadmin	Vérifier les privilèges

Total: 8

Projet de création d'une base et de tables

1- écrire le nom de la table : Eleve

Créer une nouvelle table sur la base DB_Calendrier

Nom: Nombre de colonnes:

6-Cliquer pour sauver la table

Sauvegarder

Ou Ajouter

1

colonne(s)

Exécuter

3-Cliquer pour ajouter des colonnes dans la table

2-Cliquer ici pour ajouter une nouvelle table à la base

5-Compléter tous les champs

5-Compléter tous les champs

5-Compléter tous les champs

4-Ajouter le nom de la table

5-Compléter tous les champs

Nom de la table:

Eleve

Colonne

Type

Interclassement

Attributs

Null

Index

A.I

id

INT

Aucune

UNSIGNED

☐

INDEX

☒

prenom

VARCHAR

50

Aucune

☐

☐

nom

VARCHAR

50

Aucune

☐

☐

classe

VARCHAR

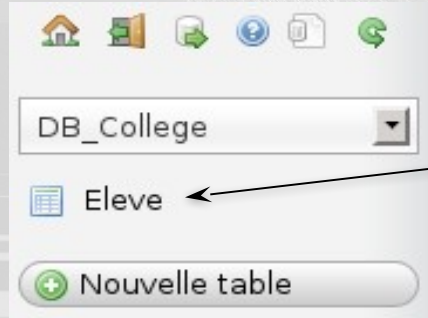
20

Aucune

☐

☐

Projet de création d'une base et de tables



La table est ainsi créée

Suivre la même procédure pour créer l'autre table Sortie.

Connexion à une base de données

Ecrire le code suivant, à l'aide d'un simple éditeur de texte :

```
<?php
/*Définition des constantes*/
define('DB_ADDRESS','localhost');
define('DB_NAME','DB_Calendrier');
define('DB_USER','root');
define('DB_PASS','votre_mot_de_passe');

/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);
echo "Connexion réussie !";

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

Tester le fichier dans un navigateur web.
Que lisez-vous ?

Écriture de données dans une table

Écrire le code suivant, à l'aide d'un simple éditeur de texte :

```
<?php
/*Définition des constantes*/
define('DB_ADDRESS','localhost');
define('DB_NAME','DB_Calendrier');
define('DB_USER','root');
define('DB_PASS','votre_mot_de_passe');

/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);

$mydb->query("INSERT INTO Eleve(prenom, nom, classe) VALUES ('Pierre','DURAND','2nd2')");

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

Exécuter 1 fois le fichier dans un navigateur web
Que lisez-vous dans phpmyadmin ?

The screenshot shows the phpMyAdmin interface. On the left, the database 'DB_College' is selected, and the table 'Eleve' is highlighted. A button 'Cliquez dessus' points to the 'Eleve' table. The main area shows the table structure and data. The SQL query 'SELECT * FROM `Eleve` LIMIT 0,30' is displayed. Below the query, the table structure is shown with columns: id, prenom, nom, classe. The data shows one record: Pierre Durand, 2nd2.

id	prenom	nom	classe
1	Pierre	Durand	2nd2

Ecriture de données dans une table

Ajouter la ligne manquante pour écrire une donnée dans la table Eleve

```
/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);

/*lecture du contenu de la table*/
$resultat = $mydb->query("SELECT * FROM Eleve");

$obj_lu = $resultat->fetch_object(); //tri par objet

echo 'Première ligne lue : ' . $obj_lu->prenom . ' ' . $obj_lu->nom . ' ' . $obj_lu->classe;

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

Tester le fichier dans un navigateur web. Vous ne devriez rien lire à l'écran.

Visualiser le contenu de la table Eleve dans la base de données

Ecriture de données dans une table

Ajouter 4 autres nom, prénom et classe dans la table élèves comme ci-dessous

```
/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);

$mydb->query("INSERT INTO Eleve(prenom, nom, classe) VALUES ('Jean','Valgeant','1sti2d3')");
$mydb->query("INSERT INTO Eleve(prenom, nom, classe) VALUES ('Romain','Dupont','tsi1')");
$mydb->query("INSERT INTO Eleve(prenom, nom, classe) VALUES ('Julie','Dominguez','tsi1')");
$mydb->query("INSERT INTO Eleve(prenom, nom, classe) VALUES ('Sophie','Chloé','btsnir')");

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

puis visualiser le résultat dans la table avec phpmyadmin.

Visualiser le contenu de la table Eleve dans la base de données. Ce qui est écrit est-il cohérent avec ce qui est entré dans le fichier php ? Visualiser la colonne « id ». Pourquoi la valeur est différente à chaque ligne, alors que nous ne lui donnons pas de valeur ?

	id	prenom	nom	classe
<input type="checkbox"/> Modifier Éditer en place Copier Effacer	1	Pierre	Durand	2nd2
<input type="checkbox"/> Modifier Éditer en place Copier Effacer	2	Jean	Valgeant	1sti2d3
<input type="checkbox"/> Modifier Éditer en place Copier Effacer	3	Romain	Dupont	tsi1
<input type="checkbox"/> Modifier Éditer en place Copier Effacer	4	Julie	Dominguez	tsi1
<input type="checkbox"/> Modifier Éditer en place Copier Effacer	5	Sophie	Chloé	btsnir

Lire les lignes dans la table

Ecriture de données dans une table

Nous allons maintenant lire 2 tables ensemble. Ecrire les lignes ci-dessous pour remplir la table Sortie

Modifier simplement la lignes de la requête sql :

```
<?php
/*Définition des constantes*/
define('DB_ADDRESS','localhost');
define('DB_NAME','DB_Calendrier');
define('DB_USER','root');
define('DB_PASS','votre_mot_de_passe');

/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);
$mydb->query("INSERT INTO Sortie(titre,date,classe) VALUES('Theatre','2014-12-10 10:00:00','2nd2')");
$mydb->query("INSERT INTO Sortie(titre,date,classe) VALUES('Musee','2014-12-18 8:30:00','tsi1')");

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

puis visualiser le résultat à l'aide d'un navigateur web et dans la base de données avec phpmyadmin

Visualiser le contenu de la table Sortie dans la base de données. Ce qui est écrit est-il cohérent avec ce qui est entré dans le fichier php ?
Visualiser la colonne « id ». Pourquoi la valeur est différente à chaque ligne, alors que nous ne lui donnons pas de valeur ?

Lecture de données dans une table

Modifier le fichier pour ajouter les lignes manquantes ci-dessous :

```
/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);

/*lecture du contenu de la table*/
$resultat = $mydb->query("SELECT * FROM Eleve");

$obj_lu = $resultat->fetch_object(); //tri par objet

echo 'Première ligne lue : '.$obj_lu->prenom.' '.$obj_lu->nom.' '.$obj_lu->classe;

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

puis visualiser le résultat à l'aide d'un navigateur web.

Lecture de données dans une table

Maintenant pour lire toutes les lignes du tableau, il faut faire une boucle :

```
/*ouverture de la connexion à la base de données*/  
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);  
  
/*lecture du contenu de la table*/  
$resultat = $mydb->query("SELECT * FROM Eleve");  
  
while($obj_lu = $resultat->fetch_object())  
    echo 'Ligne '.$obj_lu->id.' lue : '.$obj_lu->prenom.' '.$obj_lu->nom.' '.$obj_lu->classe."<br/>";  
  
/*fermeture de la connexion à la base de données*/  
$mydb->close();  
?>
```

puis visualiser le résultat à l'aide d'un navigateur web.

Le problème de lire toutes les lignes de la table est que lorsque celle-ci est de taille importante, le résultat retourné est de très grande taille et met beaucoup de temps.

Lecture de données dans une table

Nous allons utiliser des filtres de lectures afin de spécifier les données que nous souhaitons récupérer.

Modifier simplement la lignes de la requête sql :

```
<?php
/*Définition des constantes*/
define('DB_ADDRESS','localhost');
define('DB_NAME','DB_Calendrier');
define('DB_USER','root');
define('DB_PASS','votre_mot_de_passe');

/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);

/*lecture du contenu de la table*/
$resultat = $mydb->query("SELECT id, prenom, nom, classe FROM Eleve");

echo 'Ligne : ' . $obj_lu->id . ' lue : ' . $obj_lu->prenom . ' ' . $obj_lu->nom . ' ' . $obj_lu->classe . '<br/>';

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

puis visualiser le résultat à l'aide d'un navigateur web.
Justifier le résultat obtenu à l'écran.

Lecture de données dans une table

Nous pouvons trier les données par ordre croissant.

Modifier simplement la lignes de la requête sqli :

```
<?php
/*Définition des constantes*/
define('DB_ADDRESS','localhost');
define('DB_NAME','DB_Calendrier');
define('DB_USER','root');
define('DB_PASS','votre_mot_de_passe');

/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);

/*lecture du contenu de la table*/
$resultat = $mydb->query("SELECT id, prenom, nom, classe FROM Eleve");

while($obj_lu=$resultat->fetch_object())
    echo 'Ligne : '.$obj_lu->id.' lue : '.$obj_lu->prenom.' '.$obj_lu->nom.' '.$obj_lu->classe.'  
';

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

puis visualiser le résultat à l'aide d'un navigateur web.
Justifier le résultat obtenu à l'écran.

Lecture de données dans une table

Nous pouvons trier les données par ordre décroissant.

Modifier simplement la lignes de la requête sql :

```
<?php
/*Définition des constantes*/
define('DB_ADDRESS','localhost');
define('DB_NAME','DB_Calendrier');
define('DB_USER','root');
define('DB_PASS','votre_mot_de_passe');

/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);

/*lecture du contenu de la table*/
$resultat = $mydb->query("SELECT id, prenom, nom, classe FROM Eleve ORDER BY DESC");

while($obj_lu=$resultat->fetch_object())
    echo 'Ligne : '.$obj_lu->id.' lue : '.$obj_lu->prenom.' '.$obj_lu->nom.' '.$obj_lu->classe.'  
';

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

puis visualiser le résultat à l'aide d'un navigateur web.
Justifier le résultat obtenu à l'écran.

Lecture de données dans une table

Nous allons donner plus de contraintes pour filtrer d'avantage les informations souhaités

Modifier simplement la lignes de la requête sqli :

```
<?php
/*Définition des constantes*/
define('DB_ADDRESS','localhost');
define('DB_NAME','DB_Calendrier');
define('DB_USER','root');
define('DB_PASS','votre_mot_de_passe');

/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);

/*lecture du contenu de la table*/
$resultat = $mydb->query("SELECT id, prenom, nom, classe FROM Eleve WHERE classe='tsil'");

while($obj_lu=$resultat->fetch_object())
    echo 'Ligne : '.$obj_lu->id.' lue : '.$obj_lu->prenom.' '.$obj_lu->nom.' '.$obj_lu->classe.'  
';

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

puis visualiser le résultat à l'aide d'un navigateur web.

Justifier le résultat obtenu à l'écran.

Lecture de données dans une table

Nous pouvons également filtrer par début de mot

Modifier simplement la lignes de la requête sql :

```
<?php
/*Définition des constantes*/
define('DB_ADDRESS','localhost');
define('DB_NAME','DB_Calendrier');
define('DB_USER','root');
define('DB_PASS','votre_mot_de_passe');

/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);

/*lecture du contenu de la table*/
$resultat = $mydb->query("SELECT id, prenom, nom, classe FROM Eleve WHERE classe LIKE 'ts%'");

while($obj_lu=$resultat->fetch_object())
    echo 'Ligne : '.$obj_lu->id.' lue : '.$obj_lu->prenom.' '.$obj_lu->nom.' '.$obj_lu->classe.'  
';

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

puis visualiser le résultat à l'aide d'un navigateur web.
Justifier le résultat obtenu à l'écran.

Lecture de données dans une table

Lecture de données dans une table

Nous pouvons également filtrer par fin de mot

Modifier simplement la lignes de la requête sql :

```
<?php
/*Définition des constantes*/
define('DB_ADDRESS','localhost');
define('DB_NAME','DB_Calendrier');
define('DB_USER','root');
define('DB_PASS','votre_mot_de_passe');

/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);

/*lecture du contenu de la table*/
$resultat = $mydb->query("SELECT id, prenom, nom, classe FROM Eleve WHERE classe LIKE '%ts'");

while($obj_lu=$resultat->fetch_object())
    echo 'Ligne : '.$obj_lu->id.' lue : '.$obj_lu->prenom.' '.$obj_lu->nom.' '.$obj_lu->classe.'  
';

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

puis visualiser le résultat à l'aide d'un navigateur web.
Justifier le résultat obtenu à l'écran.

Lecture de données dans une table

Nous pouvons également filtrer par ce qui est contenu dans un mot

Modifier simplement la lignes de la requête sql :

```
<?php
/*Définition des constantes*/
define('DB_ADDRESS','localhost');
define('DB_NAME','DB_Calendrier');
define('DB_USER','root');
define('DB_PASS','votre_mot_de_passe');

/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);

/*lecture du contenu de la table*/
$resultat = $mydb->query("SELECT id, prenom, nom, classe FROM Eleve WHERE classe LIKE '%ts%'");

while($obj_lu=$resultat->fetch_object())
    echo 'Ligne : '.$obj_lu->id.' lue : '.$obj_lu->prenom.' '.$obj_lu->nom.' '.$obj_lu->classe.'  
';

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

puis visualiser le résultat à l'aide d'un navigateur web.
Justifier le résultat obtenu à l'écran.

Lecture de données dans une table

Nous allons maintenant lire 2 tables ensemble. Ecrire les lignes ci-dessous pour lire les tables Eleve et Sortie

Modifier simplement la lignes de la requête sql :

```
<?php
/*Définition des constantes*/
define('DB_ADDRESS','localhost');
define('DB_NAME','DB_Calendrier');
define('DB_USER','root');
define('DB_PASS','votre_mot_de_passe');

/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);

$resultat_eleve = $mydb->query("SELECT id, prenom, nom, classe FROM Eleve WHERE classe LIKE 'tsil'");

$resultat_sortie = $mydb->query("SELECT id, titre, date, classe FROM Sortie WHERE classe LIKE 'tsil'");

echo 'Eleves concernés : <br/>';
while($obj1_lu=$resultat_eleve->fetch_object())
    echo 'Ligne : '.$obj1_lu->id.' : '.$obj1_lu->prenom.' '.$obj1_lu->nom.' '.$obj1_lu->classe.'<br/>';

echo 'Sortie concernée : <br/>';
while($obj2_lu=$resultat_sortie->fetch_object())
    echo 'Ligne : '.$obj2_lu->id.' : '.$obj2_lu->titre.' '.$obj2_lu->date.' '.$obj2_lu->classe.'<br/>';

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

puis visualiser le résultat à l'aide d'un navigateur web

Lecture de données dans une table

Nous allons maintenant lire 2 tables ensemble avec 1 seule requête sql. Ecrire les lignes ci-dessous pour lire les tables Eleve et Sortie

Modifier simplement la lignes de la requête sql :

```
<?php
/*Définition des constantes*/
define('DB_ADDRESS','localhost');
define('DB_NAME','DB_Calendrier');
define('DB_USER','root');
define('DB_PASS','votre_mot_de_passe');

/*ouverture de la connexion à la base de données*/
$mydb = new mysqli(DB_ADDRESS, DB_USER, DB_PASS, DB_NAME);

/*lecture du contenu de la table*/
$resultat_ensemble = $mydb->query("SELECT prenom, nom, Eleve.classe AS classe, titre, date, FROM Sortie LEFT JOIN
Eleve ON Eleve.classe=Sortie.classe");

while($obj3_lu=$resultat_ensemble->fetch_object())
    echo 'Ligne lue : ' . $obj3_lu->prenom . ' ' . $obj3_lu->nom . ' ' . $obj3_lu->classe . ' :: ' . $obj3_lu->titre . ' ' . $obj3_lu->date . '<br/>';

/*fermeture de la connexion à la base de données*/
$mydb->close();
?>
```

puis visualiser le résultat à l'aide d'un navigateur web
Pourquoi obtient-on ces valeurs à l'écran ?

Petit projet de calendrier

Récupérer sur la plateforme le fichier : **calendrier_btssn.php**

Placer ce fichier dans votre répertoire de votre site virtuel

Dans un navigateur web, tester le fichier. Si les dates entrées sont l'une du calendrier, alors vous devriez voir un message apparaître sous le curseur de la souris.

Calendrier prévisionnel des événements à venir

Passez la souris sur les chiffres pour obtenir plus de détails

Janvier L M M J V S D 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31	Février L M M J V S D 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28	Mars L M M J V S D 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31	Avril L M M J V S D 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
Mai L M M J V S D 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31	Juin L M M J V S D 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30	Juillet L M M J V S D 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31	Août L M M J V S D 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Septembre L M M J V S D 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30	Octobre L M M J V S D 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31	Novembre L M M J V S D 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30	Décembre L M M J V S D 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31

Petit projet de calendrier – Ajout d'un formulaire

Ajouter maintenant un formulaire, dans l'emplacement indiqué dans le fichier **calendrier_btssn.php** et enregistrer les données du formulaire dans la base de données puis visualiser le résultat dans phpmyadmin et dans le projet

Résumé

Vous venez d'apprendre comment traiter des données venant d'un formulaire et de sécuriser vos pages php des données reçues.

Vous avez vu comment enregistrer et lire des données dans un fichier de type texte.

Enfin vous avez appris à dialoguer avec la base de données en utilisant le langage php et les instructions liées à SQLi