



Langage PHP

Dr. Dellys Hahchemi

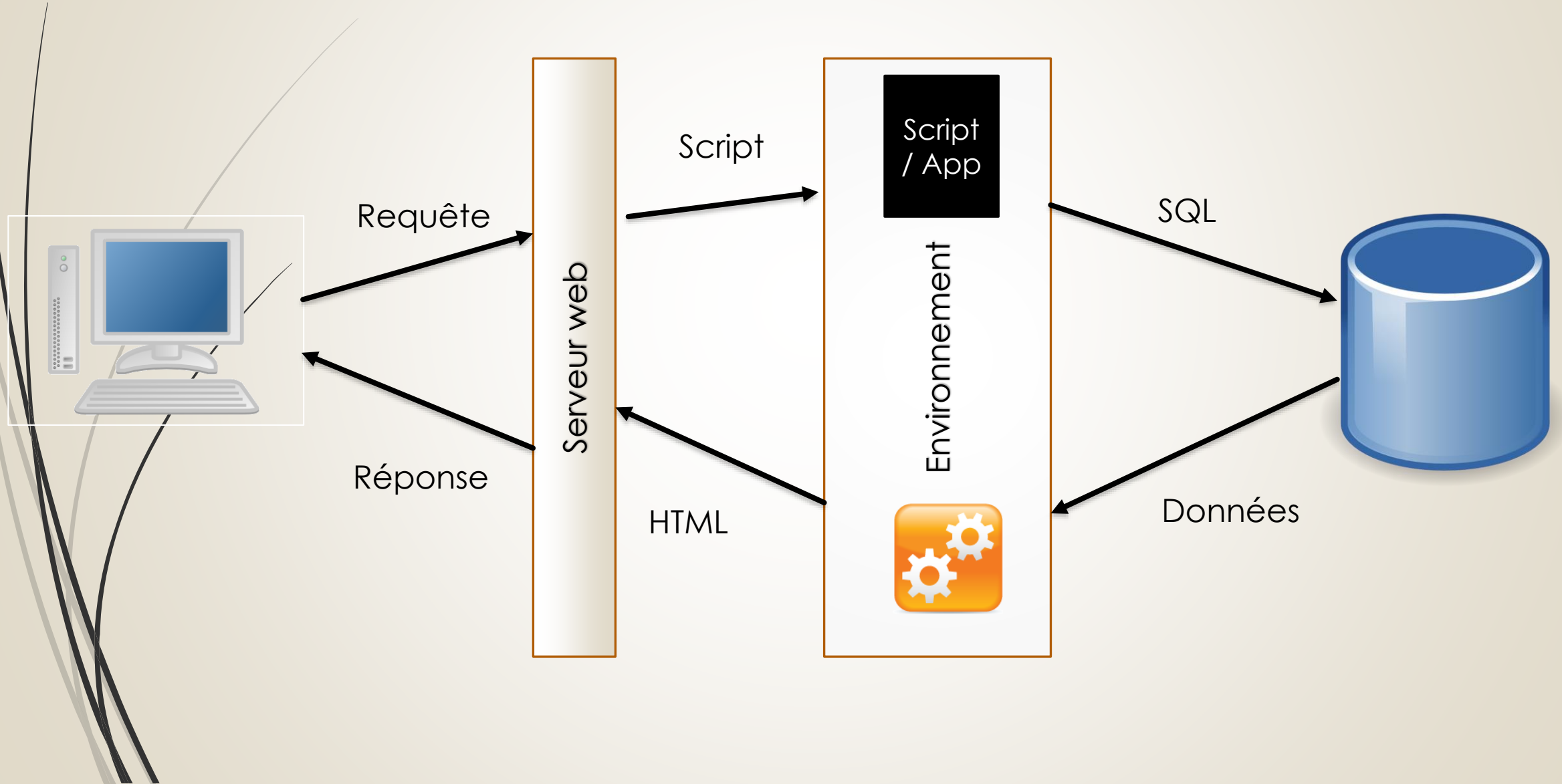
Cours Technologie et Développement Web (TDW) 2CS

2019-2020

Web Dynamique

- Les sites web faits exclusivement en HTML, CSS, JavaScript sont dits *statiques*
- Les sites statiques ne s'adaptent pas à beaucoup de situations : refléter le contenu d'une BDD, adapter le contenu à un profil...
- Le **web dynamique**: processus de *génération* de documents HTML par des scripts (côté serveur) ou des programmes au lieu d'être créés manuellement.
- ASP.NET et PHP sont des langages permettant le développement du web dynamique

Web Dynamique





Langage PHP



- Langage WEB de script (interprété) côté serveur créé en 1994 par Rasmus lerdof.
- Bibliothèque C pour les pages web dynamiques transformée en langage à part entière.
- Code PHP direct avec l'extention « .php » ou intégré à une page HTML avec la balise <PHP>.
- Code côté serveur, pas accessible par le client.
- Les codes générés sont en langage HTML, CSS et JavaScript.
- Obligation d'une application serveur (Apache, IIS ...).
- **SYNTAXE PROCHE du LANGAGE C**

Syntaxe PHP

- Écrire directement sur un fichier d'extension « .PHP »
- Utiliser une balise PHP dans un fichier HTML sur un serveur.
- Syntaxe 1:

```
<?php  
  
echo "code PHP";  
  
?>
```

- Syntaxe 2:

```
<script language="php"> echo 'l\utilisation de PHP par la balise script'; </script>
```

- L'anti-slash « \ » est le caractère d'échappement en PHP.

Variables PHP

- ▶ Les variables sont identifiées par le caractère « \$ » au début du nom de la variable.
- ▶ Les Variable PHP sont sensibles à la casse.
- ▶ Exemple

```
<?php
$Nom='Ali';
$Prenom="Mohamed";
echo 'Nom: ', $Nom, ' Prénom: ', $Prenom, '.';
?>
```



Variables PHP

- ▶ Les variables déclarés dans une fonction sont considérées comme variables locales (la variable locale est prioritaire sur une variable global de même nom).
- ▶ Si une variable locale possède le même nom qu'une variable globale, cette dernière peut être accédée par le tableau prédéfini `$GLOBAL['nom_variable']`.
- ▶ Pour rendre une variable locale visible de l'extérieure de la fonction, utilisez le mot clés « global ».
- ▶ Les variables externes au fichier PHP sont accessibles via les superglobals.

Variables Externes (Superglobals)

Superglobals	Description	Exemple
<code>\$GLOBALS</code>	Toutes les variables globales de l'application	<code>\$GLOBALS['var']</code>
<code>\$_POST</code>	Les variables envoyées par la méthodes POST via un formulaire	<code>\$_POST['password']</code>
<code>\$_GET</code>	Les variables envoyées par la méthodes GET via un formulaire	<code>\$_GET['firstname']</code>
<code>\$_FILES</code>	Tableau associatif des valeurs associées aux éléments téléchargés par le script courant	<code>\$_FILES['nom_fichier']['size']</code>
<code>\$_SESSION</code>	Tableau associatif des valeurs associées à la session courante	<code>\$_SESSION['login']</code>
<code>\$_COOKIE</code>	Tableau associatif des valeurs associées à un cookie	<code>\$_COOKIE['site_prefere']</code>
<code>\$_SERVER</code>	Les informations liées au serveur utilisé.	<code>\$_SERVER['SERVER_NAME']</code>

Fonctions

- Les fonctions en PHP sont déclarées à l'aide du mot clés « function ».
- Exemple (passage par valeur):

```
function Addition($a, $b){  
    $c=$a+$b;  
    return $c;  
}
```

- Exemple (passage par référence):

```
function Addition($a, $b, &$amp;$c){  
    $c=$a+$b;  
}
```



Tableaux

- Les tableaux sous PHP sont dynamiques et n'ont pas de taille prédéfinie et peuvent être non uniformes.
- Exemple:

```
Tab_Jus['pomme']=1;  
Tab_Jus['pomme']['orange']=3;  
Tab_Jus['citron']['anans']='zéro';  
Tab_Jus['citron']=4;
```



Base de données et PHP



- ▶ Le côté dynamique de PHP requiert des modifications fréquentes des informations, d'où la nécessité d'une source de données dynamique tel qu'une base de données.
- ▶ MySQL et SQLServer sont des langages souvent prédéfinis dans les serveurs PHP.
- ▶ Activer les variables d'environnement dans httpd.conf pour utiliser un SGBD dans un serveur HTTP.
- ▶ MySQL est activé par défaut par les applications WAMP, XAMP....

MySQL et PHP

- PHP se connecte à la base de données MySQL via la fonction « `mysqli_connect` ».
- Exemple:

```
$link = mysqli_connect("Serveur", "Utilisateur", "Password")  
or die("<p> Connexion impossible: ".mysql_error()."</p>");
```

- `$Link`: l'identifiant de la connexion.
- « Serveur »: L'URL ou l'adresse IP du serveur de base de données.
- « Utilisateur »: Le nom de l'utilisateur de la BDD.
- « password »: mot de passe.

MySQL et PHP

- La déconnexion se fait via la fonction « `mysql_close ($link)` ».
- `$link` est l'identifiant de la base de données qu'on désire déconnecter.
- PHP peut ouvrir plusieurs bases de données avec des identifiants différents.
- Pour lancer une requête MySQL avec PHP, il faut utiliser la fonction « `mysql_query()` »
- Exemple:

```
$query="select prenom,nom from Person";  
$Result=mysql_query($query);
```

Fonctions MySQL en PHP

Fonction	Description
mysqli_connect(\$server, \$user, \$pswd)	Connexion à une base de données dont le serveur \$server avec nom d'utilisateur et mot de passe.
mysqli_close(\$idconnexion)	Ferme la connexion à la base de donnée identifiée
mysqli_query(\$request, \$id)	Effectue une requête SQL sur la base de données identifiée
mysqli_result(\$result, \$row)	Récupère le résultat de la ligne \$row du résultat d'une requête
mysqli_affected_rows	Nombre de lignes affecté lors de la dernière requête
mysqli_insert_id()	Récupère le dernier identifiant généré en AUTO_INCREMENT de la dernière requête.
mysqli_real_escape_string ()	Échappe les caractères spéciaux de MySQL.
mysqli_select_db(\$name)	Sélectionne la base de données de nom « \$name »

PDO (PHP Data Object)

- PDO est une bibliothèque PHP qui permet d'utiliser les fonctionnalités des bases de données en utilisant l'Orienté Objet.
- PDO permet de se libérer des fonctions dédiées des SGBD en PHP. Un code peut utiliser plusieurs BDD.
- La connexion à la BDD se fait en instanciant l'objet PDO, exemple:

```
$dsn="mysql:dbname=\"$base.\";host=\"$server;";
try{
    $connexion=new PDO($dsn,$user,$password);
}
catch (PDOException $e){
    printf("Échec de la connexion : %s\n", $e->getMessage());
    exit();
}
```

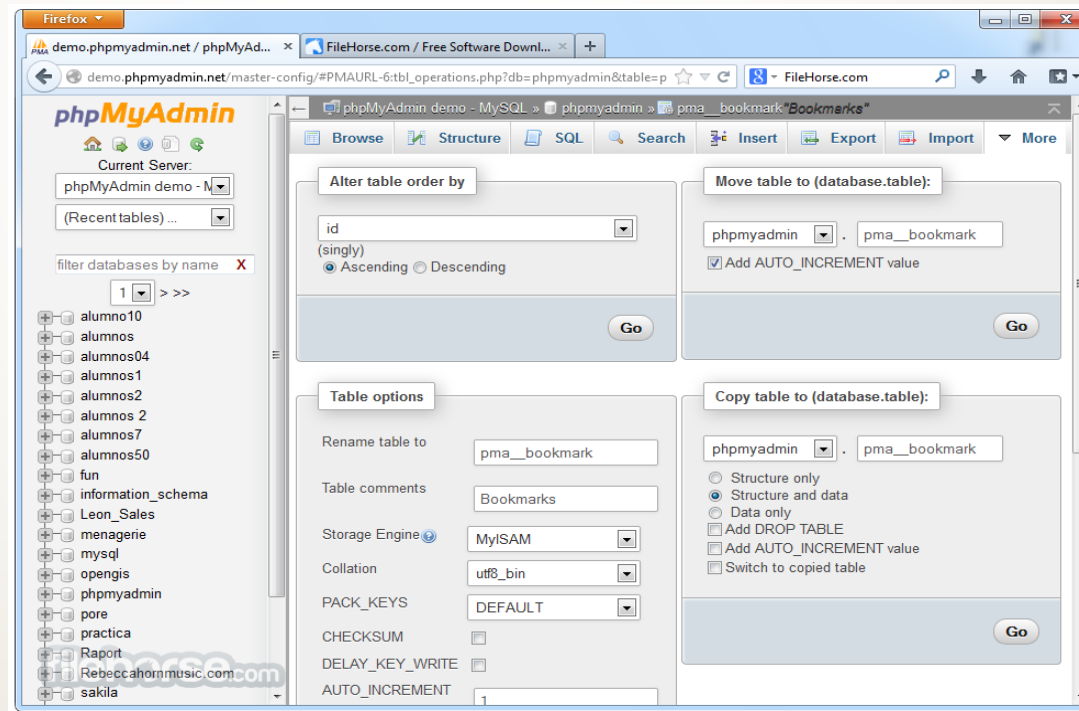

PDO (PHP Data Object)

- Les requêtes s'exécutent via la méthode query().
- Exemple:

```
$sql="select * from person";  
  
foreach ($connexion->query($sql) as $row)  
    echo  
        $row['PRENOM']." ".  
        $row['NOM']."<br/>\n";
```

PHPmyAdmin

- PHP via une application serveur (Wamp, Xamp ...) offre une interface simple de gestion de base données qui est « phpmyadmin »





Exercice

- ▶ Créer une base de données « TDW » à l'aide d'un script sql en utilisant PHPMyadmin.
- ▶ Créer dans la BDD TDW, la table « Formations » et les remplir avec les valeurs appropriées.
- ▶ Dans le site web de l'école, supprimer l'affichage fixe en HTML des formations et leurs type et remplacer le par un code PHP qui récupère les valeurs dans la base TDW.
- ▶ Reprendre la fonctionnalité d'ajout d'une formations réalisée avec JQuery et faire en sorte que l'ajout reste permanent.



Sessions



- ▶ Une session est une période de temps où un serveur est en communication authentifiée avec un utilisateur;
- ▶ Une session ne concerne qu'un seul utilisateur à la fois;
- ▶ Les informations d'authentification (mot de passe, numéro de carte ...) ne doivent pas être conservées au sein d'une session;
- ▶ Une session contient habituellement des informations liées au compte d'un utilisateur.

Sessions avec PHP

- L'ouverture d'une session sous PHP se fait par la fonction « session_start() »
- Exemple

```
<?php
// création d'une session
session_start();
?>
```

- La gestion des variables de session se fait via la variable superglobal « \$_SESSION[] ».
- Exemple

```
<?php
session_start();
$_SESSION["user"] = "Ahmed";
$_SESSION["Password"] = md5("mot_de_passe");
echo "l'utilisateur".$_SESSION["user"]." est connecté";
?>
```

Sessions avec PHP

- La suppression des variables de session se fait par la fonction « session_unset() »
- Exemple:

```
<?php  
session_unset();  
?>
```

- La fermeture de la session se fait par la fonction « session_destroy() ».
- Exemple

```
<?php  
session_destroy();  
?>
```


Cookies avec PHP

- ▶ Un cookie est un fichier créé au sein du navigateur afin de sauvegarder des données pour une période prédéfinie.
- ▶ La création d'un cookie avec PHP se fait à l'aide de la fonction « `setcookie()` »

```
<?php
$name = "nom";
$valeur = "Ahmed";
setcookie($name, $valeur, time() + (24 * 60 * 60 * 30), "/"); // 30 jours
?>
```

- ▶ `$name` : nom de variable;
- ▶ `$valeur`: valeur de la variable;
- ▶ `Time() + (24*60*60*30)` : date d'expiration du cookie;
- ▶ `"/"`: cookie valide pour tout le site.

Cookies avec PHP

- L'accès aux valeurs des cookies se fait via la variable superglobal « \$_COOKIE »

```
<?php
    echo "la valeur du cookie est : " . $_COOKIE[$name];
?>
```

- La suppression d'un cookie se fait en mettant une date passée comme date d'expiration dans la fonction « setcookie() »

```
<?php
setcookie($name, $valeur, time() - 3600, "/");
?>
```

L'orienté Objet avec PHP

- ▶ PHP implémente la notion d'orienté objet à partir de la version 5.
- ▶ La déclaration des classes, attributs et méthodes respecte la syntaxe suivante:

```
<?php

class Person{

    private $nom;
    private $prenom;

    public function setNom($nom, $prenom)
    {
        $this->nom=$nom;
        $this->prenom=$prenom;
    }
}

?>
```

Propriétés

- La syntaxe de l'Orienté Objet en PHP ressemble à la syntaxe C++;
- La création d'un Objet se fait à l'aide de la fonction new() et l'accès aux propriétés se fait à l'aide d'une flèche;

```
$p= new Personne();  
$p->setNom("A", "B");  
echo "Nom: ", $p->nom, " Prénom: ", $p->prenom, "/n";
```

- Outre le nom de la classe, on peut utiliser la méthode __construct (deux _) comme constructeur;

```
function __construct() { $this->nom="A";}
```

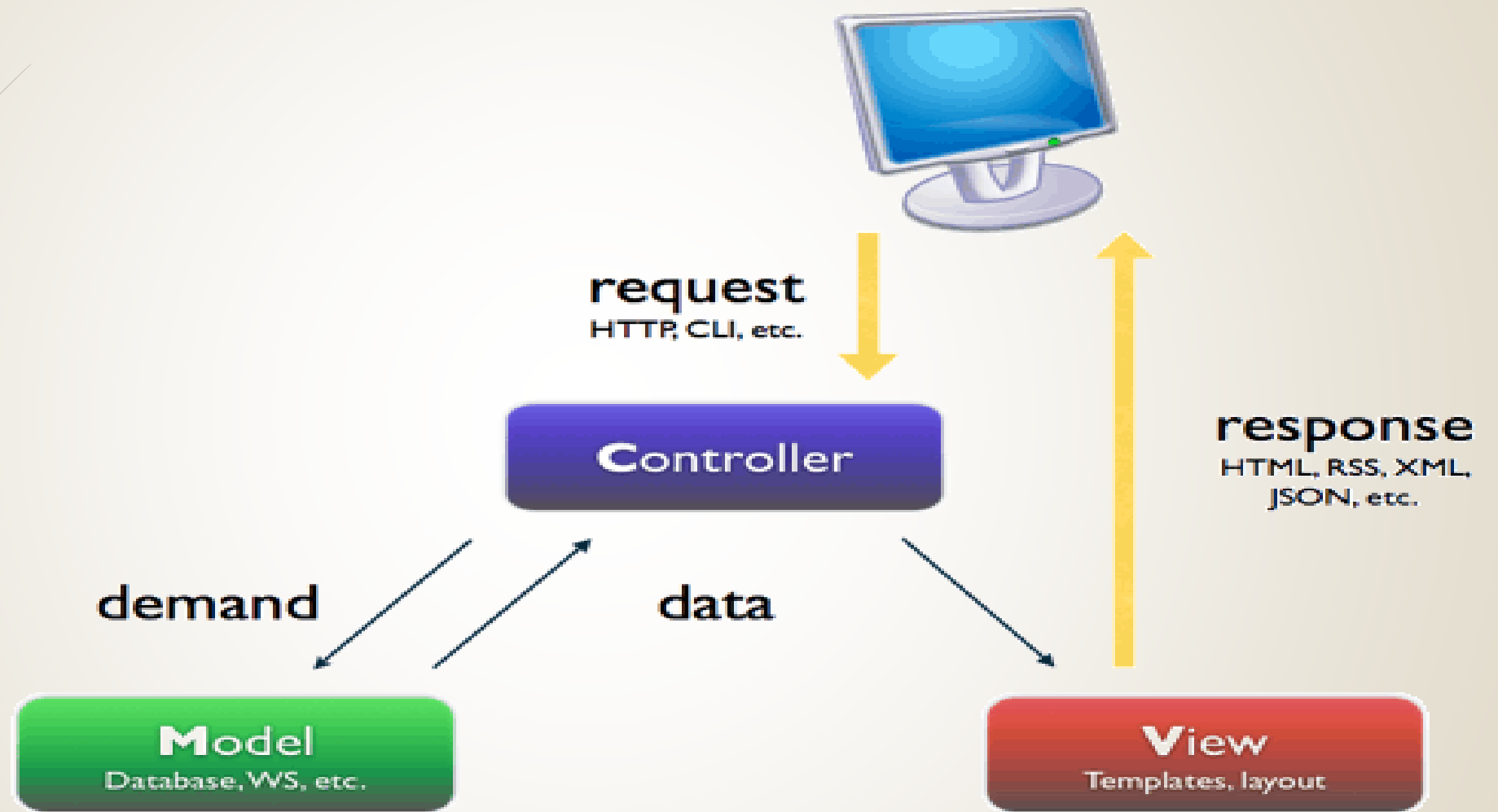
- De même, la fonction __destruct() permet de détruire un objet;

Héritage et Polymorphisme

- ▶ L'héritage et le polymorphisme existe en PHP, par contre jusqu'à la version 7 la surcharge n'est pas possible, seule la redéfinition peut être implémentée;
- ▶ Il est possible de faire référence aux méthodes et attributs d'une classe grâce à l'opérateur « :: ».

```
class Child extends Person {  
    private $nom;  
    private $prenom;  
  
    public function setNom()  
    {  
        parent::setNom(self::$nom, Child::$prenom);  
    }  
}
```

Architecture MVC





Exercice

- Nous voudrions améliorer la fonctionnalité de l'ajout de nouvelles formations.
- Pour cela, l'ajout d'une nouvelle formation se trouvera dans une nouvelle page administrateur accessible uniquement à travers un nom d'utilisateur et un mot de passe. Cette page a pour objectifs la gestion des formations de l'école.
- Une liste des formations triées par nom est affichée dans la page gestion des formations. Chaque formation possède un volume horaire, un tarif et une taxe.
- Chaque formation peut être modifiée ou supprimée. L'ajout d'une nouvelle formation se fait à travers un formulaire se trouvant dans un cadre qui s'affiche lors du click sur le bouton « Ajouter ».
- Toute modification de la part d'un administrateur doit être répercutée sur le tableau des formations côté utilisateur au bout de 5 secondes maximum sans que l'utilisateur ne soit obligé d'actualiser la page.



Bibliographie

- PHP Reference, W3C Schools.
 - Documentation PHP, php.net/docs.php
- 