

TD 3 - PHP

MVC :Modèle-Vue-Contrôleur

POO : Programmation Orientée Objet

DAO : Data Access Object

Objectif : Utilisation de la POO pour la partie Modèle et le transfert des données du modèle au contrôleur et du contrôleur à la vue.

Exercice 1 :

1. Créer un répertoire entities à la racine de votre site et une constante PATH_ENTITY. Dans ce répertoire entities, créer la classe Utilisateur.

Utilisateur
- _login - _mot - _nbRepet
+ __construct(string, string, int) + getLogin() : string + getMot() : string + getNbRepet() : int

2. Copier les classes Connexion et DAO dans le répertoire model de votre site. Implémenter la classe UtilisateurDAO qui hérite de la classe DAO.

3.

Connexion
- _bdd : PDO - __instance : Connexion
- __construct() - __clone() - __wakeup() + getInstance() : Connexion + getBdd() : PDO

DAO
- _erreur : string
- _requete (string, array) : PDOStatement + getErreur() : string + queryRow (string, array) : array + queryAll (string, array) : array

UtilisateurDAO
+ getUSer(string) : Utilisateur

Modifier le contrôleur c_hello.php et la vue v_hello.php pour prendre en compte le nouveau modèle. On supprimera l'ancien modèle m_hello.php.

Exercice 2 :

Introduction :

Ce travail sera effectué en binôme. Il faudra que le résultat soit visible sur le site de l'IUT. Créer un nouveau projet privé dans gitlab avec comme membres le binôme et l'enseignant. Ajouter un readme à ce projet rappelant le nom du binôme et précisant l'URL du site sur le serveur de l'IUT. A chaque ajout d'une nouvelle fonctionnalité (chaque fin de question), effectuer un commit pour montrer votre progression. D'autres commits (correction d'un bug, amélioration de code ...) peuvent bien sûr être effectués.

Problématique :

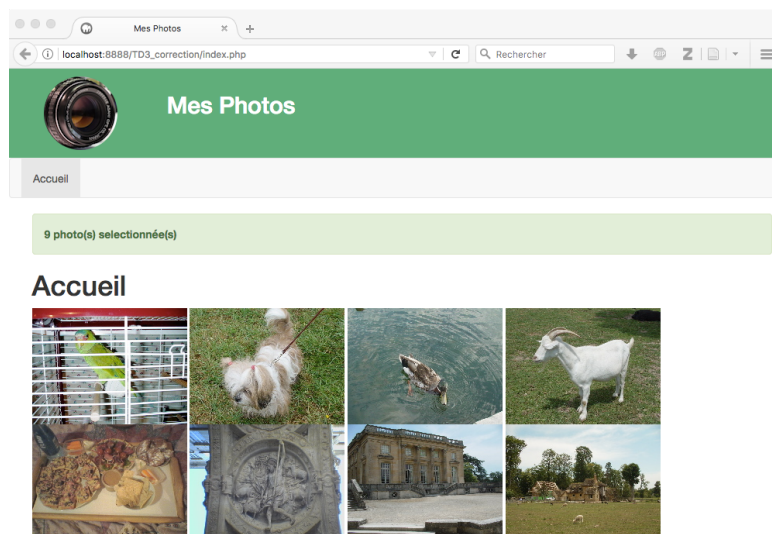
Version modifiée d'un TP d'Olivier Glück, Université Lyon 1¹. On souhaite utiliser une base de données de photos classées par catégorie et permettre aux internautes d'accéder au contenu de la base. La base de données est modélisée de la façon suivante :

Categorie(catId, nomCat)

Photo (photoId, nomFich, description, #catId)

Les clés primaires sont soulignées et les clés étrangères sont précédées d'un dièse.

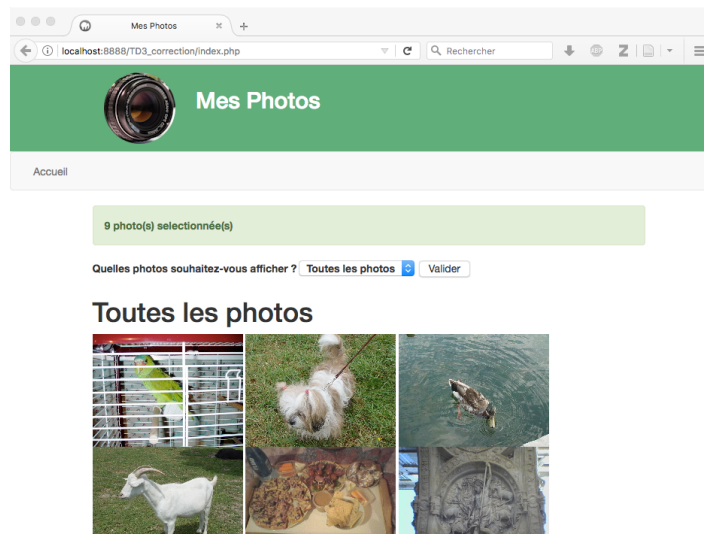
1. Pour commencer, télécharger le zip mvc_maison. Décompresser et renommer en TD3. Récupérer et sauvegarder à l'endroit adéquat les premières images à intégrer au site (images.zip).
2. Créer les tables dans la base de données. Les identifiants sont des entiers. Les noms et la description sont des varchar de taille 250 max avec utf8_general_ci comme valeur d'interclassement. Importer les données dans les tables (tuplesPhoto.sql). Exporter les tables et leur contenu dans un fichier bdd.sql dans un répertoire bdd à la racine de votre site pour sauvegarde.
3. Dans un premier temps, on souhaite permettre à l'internaute de visualiser sur une page l'ensemble des photos référencées dans la base de données. Pour ça, on construit la page d'accueil qui donnera le résultat :



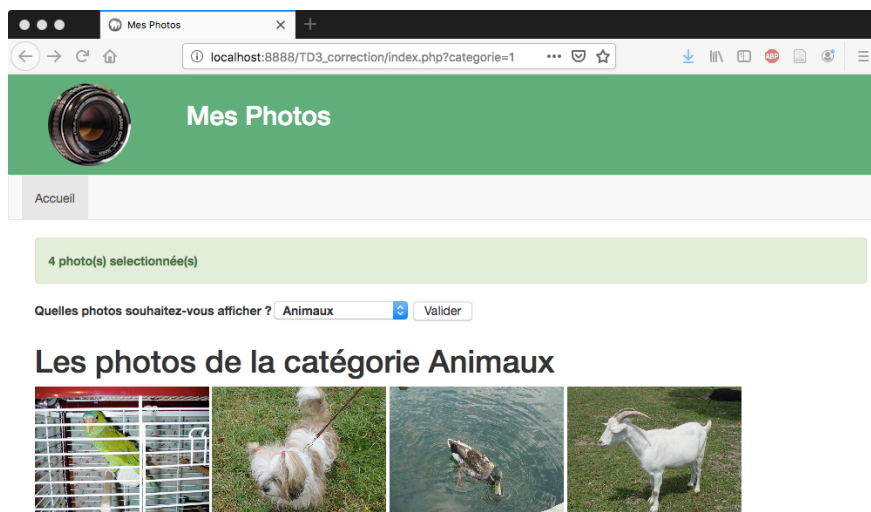
¹ https://perso.univ-lyon1.fr/olivier.gluck/Cours/Supports/M1IF_PW/PW-TP3_PHP.pdf

4. Dans un second temps, on veut pouvoir visualiser les photos pour une catégorie donnée.

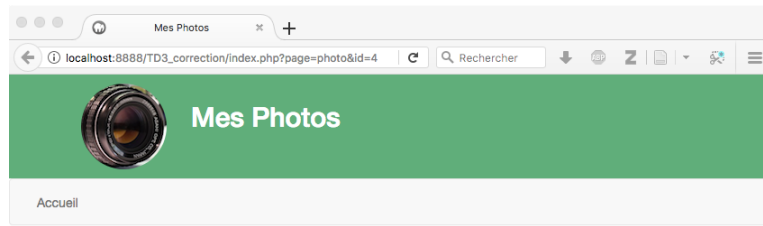
Au lancement de la page :



Après sélection d'une catégorie et validation :



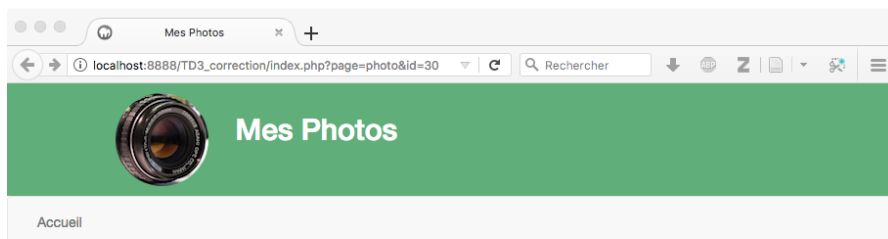
5. Pour finir, on souhaite pouvoir afficher le détail d'une photo en cliquant sur une image :
 - a) Ce sera une page dédiée à cette image. On verra l'image bien sûr, mais aussi le nom du fichier correspondant sa description et sa catégorie.
 - b) Des liens permettront de revenir sur la page qui affiche toutes les photos (accueil dans le menu) ou sur celle qui affiche toutes les photos de la même catégorie (en cliquant sur la catégorie).



Les détails sur cette photo



Description	Une chèvre dans un pré
Nom du fichier	DSC01446.jpg
Catégorie	Animaux



Identifiant de photo incorrect dans l'URL



Remarques sur l'utilisation de bootstrap :

1. La page d'une photo est obtenue en mettant l'image d'une part et la table d'autre part dans deux balises div : `<div class = "col-md-6 col-sm-6 col-xs-12">`
2. La présentation de la table est obtenue avec deux classes ainsi : `<table class="table table-bordered">`