

## TP #1 (30%) (Travail individuel)

Distribué le lundi 20 juin 2022  
Date de remise : le lundi 4 juillet 2022

### Affichage des données du site *Image du jour*

#### Objectif pédagogique

---

Dans ce travail, vous démontrerez vos aptitudes à la programmation PHP orientée objet et à l'intégration d'une base de données relationnelle avec la librairie PDO. Les concepts et connaissances pertinents sont ceux de requête/réponse HTTP, l'intégration d'un SGBDR, et l'utilisation de la syntaxe orientée objet de PHP.

#### Ce qu'il faut accomplir

---

Dans ce travail, vous utiliser le code développé en classe pour compléter l'intégration de la BD dans le site Web statique donné en même temps que cet énoncé.

Vous devez utiliser le paradigme de *programmation orientée objet* pour implémenter ces fonctionnalités mais pas le motif *architectural MVC* (dans un prochain TP, vous aurez à utiliser le cadriciel MVC développé en classe, mais pas dans le présent TP !)

Commencez par l'implémentation de la base de données requise dans MySQL (structure et données de test, les fichiers de code SQL sont fournis avec le code de départ).

Seules les fonctionnalités suivantes doivent être produites :

#### 1) Diagramme de classes (10 points)

---

Voir le fichier *idj.png*

- Remarquez que la classe `AccesBd` est complète : vous n'avez rien à faire dans le fichier correspondant ;
- Pour la classe `Commentaire`, la méthode `toutAvecVote()` nécessite l'écriture d'une requête SQL plus difficile, alors je vais la considérer optionnelle (vous vous mériterez 2 points bonis qui pourront être transférés à la note finale du cours !) ;
- La classe `Utilitaire` est une classe qui ne contient que des méthodes statiques, comme le nom l'indique cette classe sert de coffre à outils pour votre programme, dans lequel vous placez des méthodes qui n'appartiennent pas naturellement à un des « acteurs » de l'application.

#### 2) Intégration dans la page Web (10 points)

---

Voir la démo en classe et le fichier gif animé *idj-demo.gif*

- Intégrez vos fichiers de classes à l'aide de l'autochargement des classes ;
- Faites afficher les éléments suivants dans la page Web : l'image de fond (CSS) ; la description de l'image (s'il y a lieu seulement !) ; la date formatée pour laquelle l'image du jour est affichée ; les commentaires correspondants à cette image ; et si vous avez réussi à produire la fonction à points bonis, vous pouvez aussi afficher les votes sur chaque commentaire, sinon, vous devez faire afficher des nombres aléatoires à la place du compte des votes de la base de données ;

- c. Programmez l'interactivité suivante : lorsque l'utilisatrice clique sur un des boutons de contrôle visibles au-dessus de la description de l'image du jour, une nouvelle requête HTTP est envoyée au serveur avec, en paramètre d'URL, la date du jour selon le bouton cliqué (dans le format spécifié – type DATE dans MySQL) ; voici, si ce n'est pas évident, la description de l'interactivité des quatre boutons (en partant de la gauche) :
- Premier jour disponible (vous l'obtenez avec la méthode correspondante de la classe `ImageDuJour`) ;
  - Jour précédent (par rapport au jour de l'image du jour actuellement affichée) ;
  - Jour suivant (idem) ;
  - Dernier jour disponible (autrement dit « aujourd'hui » ; en effet, même si la base de données contient des images pour les jours futurs, le site Web ne peut afficher que les images correspondantes aux jours qui sont passés ou au jour courant).

### 3) Améliorez l'interface utilisateur du site (5 points)

---

- Les éléments d'interface (boutons de contrôle) permettant de naviguer dans l'historique des images doivent être rendus actifs/inactifs selon la disponibilité de l'image précédente ou suivante : autrement dit, pas de requête inutile au serveur ! Aussi, utilisez adéquatement la classe « `inactif` » (présente dans le fichier CSS) pour modifier le formatage des boutons au besoin (pour indiquer visuellement à l'utilisatrice que ces boutons sont désactivés) ;
- Modifiez l'opacité d'affichage d'un commentaire selon le taux de votes positifs reçus (nombre de votes positifs divisés par le nombre de tous les votes : attention à la division par 0 ! Logiquement si le nombre de votes négatifs est 0, le taux de votes positifs devrait être 1 😊).

### Indications techniques (5 points)

---

- Réutilisez au besoin tout code PHP développé en classe ;
- Ne modifiez pas la structure des répertoires du code de départ fourni, en particulier ne créez AUCUN autre dossier, et ne renommez pas les dossiers !
- Produisez vos fonctionnalités dans des classes PHP comme indiqué dans le diagramme de classe fourni : les fichiers de classes doivent être placés dans le dossier `classes` ;
- Appliquez une mise en forme adéquate à votre code source (suivez les recommandations/normes des bonnes pratiques PHP) ;
- Utilisez des identifiants qui contribuent à la clarté du code ;
- Remarquez que j'ai donné des indications des *types* des paramètres, des propriétés et des valeurs de retour des méthodes dans la classe `AccesBd` : si vous voulez pratiquer cet aspect, faite de même pour vos classes (pas obligatoire) ;
- Ajoutez des commentaires adéquats à votre code source (en particulier des blocs de commentaire `PHPDOC` pour toutes les méthodes de vos classes : inspirez-vous des commentaires que j'ai laissés dans la classe `AccesBd`).

### Remise

---

- Faites la remise de vos fichiers de solution (une seule archive « Zip ») par LÉA avant minuit à la date d'échéance.