Brief PHP BASICS

# Contexte du projet

Vous êtes développeur dans une agence web.

L’agence a besoin de renouveler ses outils de travail, avec notamment un nouveau système de gestion de contenu fait main pour s’adapter aux besoins spécifiques.

Le but va être de créer une preuve de concept, mettant en place un système de gestion de contenu en PHP avec une base de données SQL.

Votre travail, étape par étape sera le suivant :

# Comprendre et présenter ce qu’est un CRUD

Avant de se lancer dans la programmation, il vous faudra faire une analyse sur ce qu’est un CRUD.

* A quoi correspond cet acronyme ?

CRUD : Create Read Update delete

* A quoi sert un crud ?
* Pourquoi cette notion a été inventée ?
* A quoi un CRUD peut s’appliquer ?
* Le CRUD est-il limité à certains langages de programmation ?
* Quel autre principe de programmation connaissez-vous ?

Une petite présentation sera à faire pour échanger sur ce principe avec toute la promotion, afin d’avoir une vision bien complète de ce qu’est un CRUD.

# La base de données

Quand on parle de CRUD on va toujours avoir une base de données associée, car sans ça impossible de mettre en place celui-ci.

## Aborder la base de données

Avant de faire quoi que ce soit je vous invite à aller vous renseignez sur ce qu’est une base de données, et sous quelle forme on les retrouve.  
  
Servez-vous de votre expérience personnelle, ainsi que de vos recherches pour répondre à ces questions :

* Qu’est-ce qu’une base de données ?
* Expliquez le fonctionnement d’une base de données avec des objets du quotidien
* Quels types de base de données peut-on retrouver ?
* Quels sont les spécificités des différents types de base de données
* Quelles sont les plus répandues
* Si vous avez déjà travaillé avec des bases de données, dites-nous lesquelles ainsi que leur point fort et les choses difficiles à aborder.

## Schématiser une base de données

Avant de faire quoi que ce soit, on commence toujours par schématiser une base de données.  
Le principe c’est qu’on mette en forme les données et les relations entre celles-ci avant de faire quoi que ce soit de technique, ça nous permet de vraiment nous concentrer sur la donnée et éviter de faire des erreurs assez classiques.  
  
Ce genre de pratique s’appelle le MCD : Modèle Conceptuel de donnée.  
  
Le but de cet étape va être de se renseigner sur comment faire un modèle conceptuel de données, et de nous tester sur trois petits exercices afin de bien valider notre compréhension.  
  
Portez bien attention aux points suivants :

* Cardinalités
* Relations

Après avoir fait vos recherches essayez de réaliser les exercices suivants :

Vous pouvez faire les schémas sur papier ou utiliser des outils du type www.draw.io

### Exercice 1 :

Créer un MCD permettant de représenter un système de blog en ayant les éléments suivants :

* Auteur
  + Nom d’utilisateur
  + Date d’inscription
  + Email
* Article
  + Titre
  + Description
  + Contenu
* Catégorie
  + Nom
  + Description
* Images
  + Nom
  + Alt
  + URL

Un article ne peut avoir qu’un et un seul auteur, un article peut être présent dans plusieurs catégories, et un article peut avoir plusieurs images.

### Exercice 2 :

Créer un MCD permettant de représenter une discothèque (CD), en ayant les éléments suivants :

* Artistes
  + Nom d’artiste
  + Label
* Albums
  + Nom
  + Prix
  + Date de sortie
* Musiques
  + Nom
  + Durée
  + Date de sortie
* Genre
  + Nom du genre
  + Description

Attention ici, une musique peut avoir plusieurs genres, un album peut avoir plusieurs genres, et un artiste peut avoir plusieurs genres.  
Il peut bien sûr y avoir plusieurs artistes, musiques et albums par genre.

### Exercice 3 :

Créer un MCD permettant de représenter un Aquarium, en ayant les éléments suivants :

* Animaux
  + Nom
  + Caractère
  + Taille
  + Poids
* Familles (au sens famille d’animaux, type manchots, orcs, dauphins…)
  + Nom
  + Type
* Bassins
  + Nom
  + Code
  + Date de nettoyage
* Soigneurs
  + Nom
  + Spécialité
* Repas
  + Nom
  + Ingrédients
  + Poids

Ici, les animaux appartiennent à une seule et même famille, mais une famille peut contenir plusieurs animaux.

Les bassins contiennent une ou plusieurs familles d’animaux.

Les soigneurs soignent une ou plusieurs familles d’animaux, et les animaux ont un ou plusieurs soigneurs.

Les animaux ont plusieurs repas, et un repas peut être servi à plusieurs animaux en même temps, mais on doit garder une trace de l’heure où a été donné le repas.

### Exercice 4

Créer un MCD correspondant à une base de données de bookmark, on doit pouvoir enregistrer des catégories dans lesquels vont faire partie des liens.  
  
Le but de cette base étant de faire un mémento de différentes ressources que vous avez pu trouver sur le web.  
  
A vous de voir ce que vous allez avoir besoin ici, en termes de données dans les tables et leurs relations.

## Faire prendre vie à votre base de données

Maintenant que l’on sait faire des MCD pour représenter notre base, il va falloir venir faire prendre vie à notre base de données !

A l’aide du site SQL.sh et des informations que vous trouvez sur internet, essayez de créer la base de données de l’exercice 4 depuis PHPmyadmin.  
(Attention ici interdit d’utiliser autre chose que le SQL pour créer les tables !)

Si vous êtes en avance, commencez à préparer une petite présentation des points techniques de la création de base de données

## Insérer des données dans votre base

Maintenant que nous avons notre base crée, il va falloir insérer quelques données à l’intérieur.

Commencez à ajouter des liens dans votre bookmark via le SQL, attention, SQL seulement !   
Pas le droit d’utiliser l’interface de PHPmyAdmin.

Mais comme chaque donnée n’est pas définitive, vous devez aussi pouvoir les modifier et les supprimer via des commandes SQL.

## Se servir du SQL pour récupérer des informations

Maintenant que vous avez ces données, il va falloir que par le biais du SQL, vous arriviez à récupérer les informations correspondantes aux action suivantes :

* Récupérer tous les bookmarks
* Récupérer seulement 10 bookmarks
* Récupérer tous les bookmarks d’une catégorie
* Récupérer les bookmarks qui contiennent un mot précis dans le nom
* Récupérer les bookmarks venant d’un certain site

Maintenant que l’on a fait ça, on a désormais toutes les bases pour pouvoir enregistrer et gérer nos données dans le but de les exploiter via notre petite application PHP.

# Les bases en PHP

Dans un premier temps il va falloir créer un programme vous permettant de vous connecter à la base de données grâce à PDO.  
  
Vous devez trouver comment marche PDO et le mettre en place pour qu’il puisse venir faire le lien entre PHP et la base de données.

Faites bien attention à Utiliser les blocs try et catch pour bien gérer les erreurs de connexion

## Afficher les éléments présents dans la base

Maintenant que la connexion est faite, il va falloir récupérer les bookmarks de la base tout en les affichant de manière basique sur la page à l’aide du PHP.

## Ajouter de nouveaux éléments à l’aide d’un formulaire

Afin de pouvoir ajouter de nouveaux bookmarks, il faudra ajouter une page avec un formulaire permettant d’ajouter un bookmark dans la base de données.

## Modifier les éléments avec un formulaire pré-rempli

Ajoutez une nouvelle page permettant de modifier un bookmark, attention par contre le formulaire doit être pré-rempli avec les informations du bookmark en question.

## Supprimer les éléments avec un formulaire

Les éléments doivent pouvoir être supprimé à l’aide d’un clic sur un bouton.  
  
Vous devez utiliser les formulaires pour faire ça, il faudra être malin dans son intégration.

# Aller plus loin

Maintenant que l’on a la base de notre interaction, on va aller un peu plus loin pour rajouter de de qualité à notre projet !  
Et ce en passant par différentes phrases ☺

## Gérer un système de template pour l’affichage HTML

Dans cette catégorie, il est important de séparer le HTML du PHP, vous allez donc devoir trouver un moyen de séparer les deux.  
  
Il y a des mécaniques simples pour faire ça efficacement, regardez du côté de l’import de fichier en PHP  
  
De cette manière-là, on va pouvoir séparer les deux pour bien séparé ce qui est l’affichage, de la récupération.

## Rassembler ses requêtes dans un fichier

Maintenant que l’on a séparé notre template de notre PHP, on va désormais faire une nouvelle chose : Séparer nos requêtes dans un nouveau fichier dédié.

De la même manière que le précédent, il faudra jouer avec l’import pour pouvoir l’utiliser dans le cadre de votre projet

Vous devrez répondre aussi à cette question : Quel est l’avantage de sépare tout en différentes parties ?

## Le principe de Model-View-Controller

Avant d’avancer plus, faites un peu de veille sur le Model-View-Controller en répondant aux questions suivantes :

* A quoi ça correspond ?
* Quels sont les avantages ?
* Quels sont les désavantages ?
* Dans quel cadre doit-on l’utiliser ?
* Comment vous vous sentez par rapport à cette chose-là ?

Vous êtes bien sûr libre de rajouter de nouvelles questions à votre guise pour complémenter l’explication.  
S’en suivra une petite présentation et une discussion tous ensemble pour débattre sur le sujet ☺

## Passer par un fichier unique

Maintenant, c’est bien beau, mais à chaque fois que l’on va accéder à une nouvelle fonctionnalité de notre site on doit passer par des urls qui sont différentes (et qui plus est avec .php à la fin !)  
  
Et ce n’est pas forcément quelque chose que l’on veut !  
Ici nous allons essayer de mettre en place le principe de routeur !   
  
Je vous laisse faire un petit peu de veille pour savoir quel est le rôle d’un routeur.  
  
Une fois fini, le but va être d’en mettre un en place de manière très simple, pour ne plus avoir des routes comme celles-ci :  
  
<http://localhost/yourfile/index.php>

<http://localhost/yourfile/create.php>  
  
Mais comme celles-ci :

http://localhost/yourfile/  
<http://localhost/yourfile/create>

Il faudra donc faire quelques recherches sur comment vous pouvez faire ça à l’aide de APACHE et son rewrite engine !  
  
Trouvez ensuite un moyen d’exécuter les bonnes pages suite à cela.

## Mettre en place un système d’utilisateur avec authentification login / mot de passe

## Gérer les styles de validation du formulaire en css