TP8 Création d’un projet Bootstrap

1. Objectifs

- Télécharger et personnaliser une **Template Bootstrap 5** pour créer un site web.
- Se familiariser avec l'utilisation de Bootstrap 5 pour la création de sites web, ainsi que la personnalisation de templates existantes pour créer des designs uniques.

2. Énoncé:

- **Partie 01 : Page index**
  - Télécharger la template Bootstrap 5 en suivant le lien **https://bootstrapmade.com/demo/Medilab/**.
  - Extraire le fichier ZIP téléchargé et copier/coller les fichiers dans un nouveau dossier sur votre ordinateur.
  - Ouvrir le fichier index.html dans un éditeur de texte (par exemple, Visual Studio Code).
  - Examiner le code HTML et les fichiers CSS et JS associés pour comprendre la structure et les styles appliqués à la template.
  - Personnaliser la template en ajoutant votre propre contenu, en modifiant les couleurs et les styles, et en utilisant vos propres images.
  - Enregistrer les modifications et afficher la page HTML dans un navigateur pour voir le résultat final.
  - Si vous le souhaitez, vous pouvez également ajouter des fonctionnalités supplémentaires en utilisant des plugins ou en modifiant le code JavaScript.
  - Héberger le site web sur un serveur local pour le rendre accessible.
- **Partie 02 : Base de données**
  - Voici un schéma de base de données MySQL pour un clinique. La base de données comprendra au moins cinq tables.
  - **Table `"Patients"`**
    - La table "Patients" stocke des informations sur les patients

```
Id (clé primaire)
Nom
```

apcpedagogie.com

```
Prénom
Date de naissance
Adresse
Numéro de téléphone
Adresse e-mail
```

```
CREATE TABLE Patients (
  Id INT NOT NULL AUTO_INCREMENT,
  Nom VARCHAR(50) NOT NULL,
  Prénom VARCHAR(50) NOT NULL,
  Date_de_naissance DATE NOT NULL,
  Adresse VARCHAR(100),
  Numéro_de_téléphone VARCHAR(20),
  Adresse_email VARCHAR(100),
  PRIMARY KEY (Id)
);
```

- **Table** `"Consultations"`
  - La table "**Consultations**" enregistre les informations relatives aux consultations, y compris le médecin qui a effectué la consultation.

```
Id (clé primaire)
Date de consultation
Heure de consultation
Id_patient (clé étrangère liée à la table "Patients")
Id_médecin (clé étrangère liée à la table "Médecins")
Motif de consultation
Diagnostic
Traitement
```

```
CREATE TABLE Consultations (
  Id INT NOT NULL AUTO_INCREMENT,
  Date_de_consultation DATE NOT NULL,
  Heure_de_consultation TIME NOT NULL,
  Id_patient INT NOT NULL,
  Id_médecin INT NOT NULL,
  Motif_de_consultation VARCHAR(255),
  Diagnostic VARCHAR(255),
  Traitement VARCHAR(255),
  PRIMARY KEY (Id),
```

apcpedagogie.com

```
  FOREIGN KEY (Id_patient) REFERENCES Patients(Id),
  FOREIGN KEY (Id_médecin) REFERENCES Médecins(Id)
);
```

- **Table** `"Médecins"`
  - La table "Médecins" contient des informations sur les médecins travaillant à la clinique, y compris leur tarif de consultation.

```
Id (clé primaire)
Nom
Prénom
Spécialité
Tarif de consultation
```

```
CREATE TABLE Médecins (
  Id INT NOT NULL AUTO_INCREMENT,
  Nom VARCHAR(50) NOT NULL,
  Prénom VARCHAR(50) NOT NULL,
  Spécialité VARCHAR(100),
  Tarif_de_consultation DECIMAL(8, 2),
  PRIMARY KEY (Id)
);
```

- **Table** `"Rendez-vous"`
  - La table "Rendez-vous" stocke des informations sur les rendez-vous entre les patients et les médecins.

```
Id (clé primaire)
Date de rendez-vous
Heure de rendez-vous
Id_patient (clé étrangère liée à la table "Patients")
Id_médecin (clé étrangère liée à la table "Médecins")
```

```
CREATE TABLE Rendez_vous (
  Id INT NOT NULL AUTO_INCREMENT,
  Date_de_rendez_vous DATE NOT NULL,
  Heure_de_rendez_vous TIME NOT NULL,
  Id_patient INT NOT NULL,
  Id_médecin INT NOT NULL,
  PRIMARY KEY (Id),
```

apcpedagogie.com

```
  FOREIGN KEY (Id_patient) REFERENCES Patients(Id),
  FOREIGN KEY (Id_médecin) REFERENCES Médecins(Id)
);
```

- **Table** `"Honoraire_medecin"`
  - la table "Honoraire_medecin" enregistre les paiements effectués aux médecins sous forme d'honoraires, avec une entrée pour chaque mois et chaque année.

```
Id (clé primaire)
Id_médecin (clé étrangère liée à la table "Médecins")
Année
Mois
Montant versé
```

```
CREATE TABLE Honoraire_médecin (
  Id INT NOT NULL AUTO_INCREMENT,
  Id_médecin INT NOT NULL,
  Année INT NOT NULL,
  Mois INT NOT NULL,
  Montant_versé DECIMAL(8, 2),
  PRIMARY KEY (Id),
  FOREIGN KEY (Id_médecin) REFERENCES Médecins(Id)
);
```

- **Partie 03 : Pages de connexion**
  - Créer les pages PHP qui permettent de se connecter à une base de données (crées précédemment) MySQL avec PDO de manière sécurisée :
    - Tout d'abord, assurez-vous que le pilote PDO pour MySQL est installé sur votre serveur web.
    - Créez un fichier de configuration pour stocker les informations de connexion à la base de données, en utilisant les constantes PHP. Ce fichier doit être stocké en dehors du répertoire racine de votre site web, pour des raisons de sécurité.
    - Dans votre fichier PHP de connexion à la base de données, incluez le fichier de configuration à l'aide de la fonction PHP "require_once".
    - Utilisez la classe PDO de PHP pour établir une connexion à la base de données MySQL en utilisant les informations de connexion stockées

apcpedagogie.com

dans votre fichier de configuration.

- Utilisez la méthode PDO::setAttribute() pour définir le mode d'erreur PDO sur PDO::ERRMODE_EXCEPTION, afin de gérer les erreurs de manière appropriée.
- Pour protéger votre application contre les attaques par injection SQL, utilisez des requêtes préparées avec des paramètres de liaison.

- **Partie 04 : Système de connexion**

  - **Concevoir et développer un système de connexion sécurisé pour votre site web en utilisant des pages de connexion en PHP, incluant une gestion d'oublie de mot de passe et la récupération par session pour assurer la confidentialité et l'accessibilité des comptes utilisateurs.**
    - Pour concevoir et développer un système de connexion sécurisé pour un site web en utilisant des pages de connexion en PHP, incluant une gestion d'oubli de mot de passe et la récupération par session, voici les étapes à suivre :

      - **Planification et conception du système de connexion :**

        - Identifier les besoins de votre site web en matière de sécurité de connexion et de gestion des comptes utilisateurs.
        - Élaborer un plan détaillé pour le système de connexion, en tenant compte des différentes fonctionnalités requises (inscription, connexion, déconnexion, récupération de mot de passe, etc.) et des exigences de sécurité (chiffrement des données, protection contre les attaques par force brute, etc.).

      - **Création des pages de connexion :**

        - Concevoir les pages de connexion en PHP, en utilisant des éléments de sécurité tels que les tokens CSRF et les mécanismes de validation des entrées utilisateur.
        - Mettre en place des mesures de protection contre les attaques par force brute, telles que le verrouillage de compte après un certain nombre de tentatives de

apcpedagogie.com

connexion infructueuses.

- **Gestion des comptes utilisateurs :**
  - Développer une base de données pour stocker les informations des utilisateurs, y compris les noms d'utilisateur et les mots de passe.
  - Utiliser une méthode de hachage de mot de passe sécurisée pour stocker les mots de passe des utilisateurs.
  - Mettre en place un processus de validation d'adresse e-mail pour les nouveaux utilisateurs, afin de s'assurer de l'authenticité de leur compte.
- **Gestion de l'oubli de mot de passe :**
  - Développer une fonctionnalité de récupération de mot de passe, en utilisant un processus de vérification en deux étapes pour garantir l'authenticité de l'utilisateur.
  - Envoyer un e-mail de réinitialisation de mot de passe avec un lien unique et temporaire pour réinitialiser le mot de passe.
- **Gestion des sessions utilisateur :**
  - Utiliser des sessions pour suivre l'état de connexion de l'utilisateur.
  - Développer une fonctionnalité de déconnexion claire pour permettre aux utilisateurs de se déconnecter en toute sécurité.
  - En suivant ces étapes, vous pouvez concevoir et développer un système de connexion sécurisé pour votre site web en utilisant des pages de connexion en PHP, incluant une gestion d'oubli de mot de passe et la récupération par session pour assurer la confidentialité et l'accessibilité des comptes utilisateurs.

- **Partie 05 : Pages du site**

apcpedagogie.com

- Créer un menu standard pour toutes les pages, avec des liens vers les différentes pages principales et un bouton de déconnexion.
- Créer une page principale pour chaque table, avec des fonctions CRUD (create, read, update, delete) pour les données de cette table.
- Ajouter une zone de recherche et une pagination Bootstrap 5 à chaque page principale.
- Organiser les pages PHP CRUD pour chaque table dans des dossiers distincts.
    - Voici les étapes à suivre pour concevoir et développer une telle page(page cours comme exemple :

        - **Planification et conception de la page :**

            - Définir les besoins de la page en matière d'affichage et de fonctionnalités.
            - Élaborer un plan détaillé pour la page, en tenant compte des différentes fonctionnalités requises (affichage de données, recherche, insertion, modification, suppression, etc.) et de l'interface utilisateur.

        - **Création de la page principale :**

            - Concevoir la page en HTML 5, bootstrap 5 et en CSS, en utilisant des éléments de Bootstrap pour la mise en page et le style.
            - Écrire le code PHP pour récupérer les données de la table 'cours' à partir de la base de données MySQL et les afficher dans un tableau HTML.
            - Ajouter une zone de recherche en haut de la page pour filtrer les données affichées.
            - Ajouter un bouton pour insérer un nouveau cours.

        - **Création des modaux pour les fonctions CRUD :**

            - Concevoir des modaux pour les fonctions CRUD (create, read, update, delete) en utilisant des éléments de Bootstrap pour la mise en page et le style.
            - Écrire le code PHP pour chaque fonctionnalité CRUD en utilisant les requêtes MySQL appropriées pour interagir

apcpedagogie.com

avec la base de données.

- **Ajout des boutons détail, modifier et delete :**
    - Ajouter une colonne à la table pour les boutons détail, modifier et delete pour chaque ligne.
    - Écrire le code JavaScript ou JQuery pour déclencher les modaux appropriés lorsqu'un bouton est cliqué.
    - En suivant ces étapes, vous pouvez concevoir et développer une page principale pour chaque table MySQL, permettant d'afficher toutes les données de chaque table avec une zone de recherche en haut et un bouton pour insérer un nouveau enregistrement, ainsi qu'une colonne qui contient des boutons détail, modifier et delete. Le développement doit inclure la création de pages PHP et de modaux Bootstrap avec des fonctions CRUD (create, read, update, delete) pour les données de chaque table.
- Créer une page index qui propose des boutons cliquables pour toutes les pages principales.
- Tester l'application pour vérifier le bon fonctionnement de toutes les fonctionnalités CRUD (c'est seulement la page crée précédemment).
- Optimiser le code et la conception de l'interface utilisateur pour améliorer les performances et la convivialité de l'application.

Sommaire du cours Bootstrap

Source : https://apcpedagogie.com/tp8-creation-dun-projet-bootstrap · téléchargé le 07/11/2026 · © APC Pédagogie · apcpedagogie.com