

# Atelier 5: Création d'un formulaire de contact pour un site portfolio avec base de données MySQL

---

## Objectifs

1. Créer un formulaire de contact fonctionnel avec Bootstrap.
  2. Enregistrer les données du formulaire dans une base de données MySQL.
  3. Utiliser Bootstrap pour styliser le formulaire et rendre l'interface responsive.
- 

## Prérequis

- Installation de XAMPP (ou un environnement contenant Apache, MySQL et PHP).
  - Connaissance de base en HTML, PHP et MySQL.
  - Bootstrap disponible via un lien CDN ou fichier local.
- 

## Étapes

### 1. Préparer la base de données MySQL

1. Connectez-vous à MySQL via phpMyAdmin ou le terminal.
2. Créez une base de données appelée `portfolio` :

```
CREATE DATABASE portfolio;
```

3. Accédez à la base de données :

```
USE portfolio;
```

4. Créez une table `contact` :

```
CREATE TABLE contact (  
  idContact INT AUTO_INCREMENT PRIMARY KEY,  
  nomContact VARCHAR(50) NOT NULL,  
  prenomContact VARCHAR(50) NOT NULL,  
  emailContact VARCHAR(100) NOT NULL,  
  telephoneContact VARCHAR(15),  
  sujetContact VARCHAR(100) NOT NULL,  
  demandeContact TEXT NOT NULL
```

```
);
```

## 2. Créer le formulaire HTML avec Bootstrap

1. Créez un fichier nommé `contact.html`.
2. Insérez le code suivant :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Formulaire de Contact</title>
  <!-- Lien CDN Bootstrap -->
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>
  <div class="container mt-5">
    <h2 class="text-center">Contactez-nous</h2>
    <form action="contact.php" method="POST" class="shadow p-4 rounded bg-light">
      <div class="mb-3">
        <label for="nomContact" class="form-label">Nom :</label>
        <input type="text" class="form-control" id="nomContact" name="nomContact" required>
      </div>
      <div class="mb-3">
        <label for="prenomContact" class="form-label">Prénom :</label>
        <input type="text" class="form-control" id="prenomContact" name="prenomContact"
required>
      </div>
      <div class="mb-3">
        <label for="emailContact" class="form-label">Email :</label>
```

```

        <input type="email" class="form-control" id="emailContact" name="emailContact"
required>
    </div>
    <div class="mb-3">
        <label for="telephoneContact" class="form-label">Téléphone :</label>
        <input type="text" class="form-control" id="telephoneContact" name="telephoneContact">
    </div>
    <div class="mb-3">
        <label for="sujetContact" class="form-label">Sujet :</label>
        <input type="text" class="form-control" id="sujetContact" name="sujetContact" required>
    </div>
    <div class="mb-3">
        <label for="demandeContact" class="form-label">Message :</label>
        <textarea class="form-control" id="demandeContact" name="demandeContact" rows="4"
required></textarea>
    </div>
    <button type="submit" class="btn btn-primary w-100">Envoyer</button>
</form>
</div>
<!-- Script Bootstrap -->
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
</body>
</html>

```

### 3. Créer le script PHP pour traiter les données

1. Créez un fichier nommé `contact.php`.
2. Ajoutez le code suivant :

```

<?php
// Connexion à la base de données
$host = 'localhost';
$dbname = 'portfolio';

```

```

$username = 'root';
$password = '';

try {
    $pdo = new PDO("mysql:host=$host;dbname=$dbname;charset=utf8", $username, $password);
    $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch (PDOException $e) {
    die("Erreur de connexion : " . $e->getMessage());
}

// Vérifier si le formulaire est soumis
if ($_SERVER['REQUEST_METHOD'] == 'POST') {
    $nom = $_POST['nomContact'];
    $prenom = $_POST['prenomContact'];
    $email = $_POST['emailContact'];
    $telephone = $_POST['telephoneContact'];
    $sujet = $_POST['sujetContact'];
    $demande = $_POST['demandeContact'];

    // Insertion dans la base de données
    $sql = "INSERT INTO contact (nomContact, prenomContact, emailContact, telephoneContact,
sujetContact, demandeContact)
        VALUES (:nom, :prenom, :email, :telephone, :sujet, :demande)";
    $stmt = $pdo->prepare($sql);

    $stmt->execute([
        ':nom' => $nom,
        ':prenom' => $prenom,
        ':email' => $email,
        ':telephone' => $telephone,
        ':sujet' => $sujet,
    ]);
}

```

```
'demande' => $demande,  
]);  
  
echo "Votre message a été envoyé avec succès."  
}  
?>
```

#### 4. Test de l'application

1. Lancez le serveur local avec XAMPP :

```
sudo /opt/lampp/lampp start
```

- Placez les fichiers `contact.html` et `contact.php` dans le dossier **htdocs** de XAMPP.
- Ouvrez un navigateur et accédez à :

```
http://localhost/contact.html
```

- Remplissez le formulaire et soumettez-le.
- Vérifiez les données dans la table `contact` via phpMyAdmin.

#### Explication de l'utilisation de Bootstrap

1. **Classes utilitaires** : Bootstrap fournit des classes prédéfinies comme `form-control`, `btn-primary`, `container`, et `shadow`, qui simplifient la mise en page et le style.
2. **Responsive Design** : Le système de grille et les classes comme `col`, `row` permettent d'adapter l'affichage aux écrans de différentes tailles.
3. **Accessibilité et cohérence** : Les composants prêts à l'emploi (boutons, formulaires, alertes) respectent les standards d'accessibilité et assurent un design uniforme.

#### Validation de l'atelier

- Le formulaire doit enregistrer correctement les données dans la base de données.
- L'interface doit être claire, propre et responsive grâce à Bootstrap.