

🎯 Mini-Projet Laravel : Application de Réservation de Voitures en Ligne

📖 Contexte :

Vous devez développer une application web simple de réservation de voitures en ligne. Cette plateforme permet aux utilisateurs d'accéder à la liste des voitures disponibles et de soumettre des demandes de réservation. Elle comprend également une interface d'administration pour gérer les voitures et consulter les réservations.

✓ Objectifs du projet :

- Créer une application Laravel avec les concepts de base du MVC.
- Gérer les opérations CRUD pour les voitures.
- Permettre la réservation de voitures en ligne avec vérification des dates.
- Afficher les réservations enregistrées dans une interface d'administration simple.

Fonctionnalités principales :

🚗 Gestion des voitures :

- Ajouter, modifier, supprimer et afficher les voitures.
- Chaque voiture comporte : marque, modèle, immatriculation, prix par jour, image, disponibilité.

📅 Réservation de voitures :

- Formulaire de réservation :
 - Sélection d'une voiture.
 - Choix des dates de début et de fin.
 - Calcul automatique du prix total.
 - Enregistrement de la réservation si la voiture est disponible sur la période.

📅 Gestion des réservations (interface admin simple) :

- Afficher toutes les réservations.
- Filtrer les réservations par voiture ou par date.
- Visualiser le détail de chaque réservation.

🔗 Relations de données :

- Une voiture peut avoir plusieurs réservations.
- Une réservation est liée à une seule voiture.

📁 Structure de base des tables (suggestion) :

- cars: id, marque, modèle, immatriculation, prix_par_jour, image, disponible
- reservations: id, nom_client, email_client, voiture_id, date_debut, date_fin, prix_total

Contraintes techniques :

- Laravel 10+
- Base de données MySQL
- Utilisation de Blade pour les vues
- Bootstrap ou Tailwind CSS pour le style

🗄️ Structure de la base de données (MySQL)

Table cars

```
CREATE TABLE cars (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    marque VARCHAR(100),  
    modele VARCHAR(100),  
    immatriculation VARCHAR(50) UNIQUE,  
    prix_par_jour DECIMAL(10,2),  
    image VARCHAR(255),  
    disponible BOOLEAN DEFAULT TRUE  
);
```

Table reservations

```
CREATE TABLE reservations (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nom_client VARCHAR(100),  
    email_client VARCHAR(100),  
    voiture_id INT,  
    date_debut DATE,  
    date_fin DATE,  
    prix_total DECIMAL(10,2),  
    FOREIGN KEY (voiture_id) REFERENCES cars(id) ON DELETE CASCADE  
);
```

Données d'exemple

🚗 Insertion de voitures (cars)

```
INSERT INTO cars (marque, modele, immatriculation, prix_par_jour, image, disponible) VALUES  
( 'Peugeot', '208', '123-TN-456', 80.00, 'peugeot_208.jpg', TRUE),  
( 'Renault', 'Clio', '789-TN-321', 75.00, 'renault_clio.jpg', TRUE),  
( 'Hyundai', 'i20', '456-TN-789', 90.00, 'hyundai_i20.jpg', TRUE),
```

```
('Volkswagen', 'Golf', '147-TN-258', 120.00, 'vw_golf.jpg', TRUE);
```

Insertion de réservations (**reservations**)

```
INSERT INTO reservations (nom_client, email_client, voiture_id,  
date_debut, date_fin, prix_total) VALUES  
( 'Ahmed Ben Salah', 'ahmed@example.com', 1, '2025-05-01', '2025-05-05',  
320.00),  
( 'Fatma Messaoudi', 'fatma@example.com', 2, '2025-05-02', '2025-05-04',  
150.00);
```