

# TD n°7/8 - Vente de voitures

---

## Projet

Le projet consiste en la conception et l'implémentation en PHP/MySQL d'un site web interne pour gérer la vente des voitures dans un garage. Plus précisément, il est demandé d'implémenter une partie commune (correspondant à la partie A), plus deux extensions (à choisir parmi celles données dans la partie B, en en prenant une dans chaque groupe). En plus du site, il sera demandé un rapport dans lequel on indiquera le choix des extensions, le diagramme entités-associations final avec une explication des raisons pour lesquelles il diffère du schéma de la partie A, les relations, les limitations de l'implantation et les difficultés rencontrées. Le rapport ne devra pas dépasser 5 pages, avec au maximum 1,5 pages par extension. Il sera également demandé d'insérer dans la base un ensemble de données pertinent, permettant de tester les fonctionnalités du logiciel. Une soutenance de projet, avec démonstration, aura lieu lors de la dernière séance de TP.

Remarques :

- De plus amples détails sur l'implémentation seront donnés en TP.
- L'esthétique du site ne sera pas prise en compte dans l'évaluation du projet.

### ◇ Partie A

Soit l'énoncé suivant :

Un garage vous demande de concevoir une base de données permettant de mémoriser les modèles de voitures qu'il vend, les différentes motorisations de ces modèles ainsi que les options susceptibles d'être vendues sur chaque véhicule. Il vous demande également de concevoir une petite interface web permettant de réaliser ces opérations.

Une voiture est identifiée par sa marque (Renault, Citroën...) et sa série (Megane, Clio...). Un descriptif est associé au modèle. Pour chaque modèle, il peut exister plusieurs motorisations possibles (1.6l 16V, 1.9 Diesel...) faisant varier le prix de la voiture. De nombreuses options sont disponibles (peinture métallisée, jantes en alu...) et les véhicules vendus peuvent prendre un ensemble d'options, chacune a un prix déterminé d'avance. Finalement,

à chaque vente d'un véhicule, l'immatriculation du véhicule est mémorisée ainsi que la date d'achat du véhicule et le nom du client.

1. Trouver le schéma entités-associations correspondant (n'oubliez pas de souligner les identifiants)
2. En déduire les relations
3. Écrire en SQL des requêtes pertinentes utilisant ces relations

## ◇ **Partie B**

Voici un ensemble de possibilités d'extensions de l'application précédente, découpé en deux groupes.

### **Groupe 1**

1. On veut pouvoir regrouper des options par catégories, par exemple peinture, jantes, sièges, etc ... Deux options faisant partie d'une même catégorie ne peuvent être prise simultanément pour un même véhicule (par exemple on ne peut pas choisir en même temps peinture métallisée et peinture nacrée). L'interface web doit s'assurer d'une manière ou d'une autre que deux options d'une même catégorie ne sont pas choisies pour un même véhicule. Certaines options peuvent ne pas avoir de catégorie.
2. On ajoute une notion de gamme de voiture correspondant à un ensemble d'options prédéfinies. Une gamme peut être utilisée par un ensemble bien défini de modèles (par exemple la gamme "RXE" regroupant des jantes alu, une peinture métallisée et un pack hiver peut s'appliquer sur les Renault Megane 1,6l 16v, les Renault Clio 1,4l 16v, mais pas sur sur des Renault Clio 1,2l ni sur des Peugeot).
3. On veut pouvoir gérer des informations sur les vendeurs : nom, prénom, véhicules vendus, salaire, date d'embauche.
4. On veut que le prix d'une option puisse varier en fonction du modèle. L'option conserve un prix par défaut. Par exemple, le prix de l'option "jantes alu" est par défaut de 1500 euros, sauf sur les Renault Clio 1,2l où son prix est de 1300 euros et sur les Peugeot 607 3,0l v6 où son prix est de 2000 euros.

### **Groupe 2**

1. Gestion de stock. On veut pour chaque modèle et chaque option une information sur sa disponibilité : en stock ou non, si non, quel est son délai de livraison.

2. Statistiques sur les véhicules. On veut afficher un ensemble de statistiques intéressantes sur les véhicules et les options, par exemple modèle le moins cher, option la plus prisée, prix moyen des véhicules vendus, etc ...
3. Statistiques sur les vendeurs : celui qui a vendu le plus de voiture, celui qui est le plus récemment embauché, le salaire moyen d'un vendeur, etc ...
4. Accès au site par identifiant. On suppose que l'accès au site nécessite de passer par une page d'identification où l'on saisit un login et un mot de passe. Une fois l'utilisateur identifié, on gardera trace de qui il est grâce à un argument `identifiant` qui sera passé. à chaque page php (sous la forme `identifiant=xxxxxx` dans l'url, où `xxxxxx` est le login, c.f. `GET`). On distinguera deux catégories d'utilisateurs, selon qu'ils auront ou non le droit de saisir de nouvelles données (s'ils n'ont pas le droit, on ne fera pas apparaître le lien vers la page des saisies). Note : il s'agit bien évidemment d'une gestion très simplifiée d'un accès par login, on ne cherche pas ici à construire un site véritablement sécurisé.
5. On souhaite pouvoir effectuer une recherche de véhicule ou de modèle de véhicule. On proposera plusieurs critères (prix minimum/maximum, marque, motorisation ...) qui devront pouvoir être combinés.

Choisir une extension dans chaque groupe, puis modifier le diagramme entités-associations de la partie A, pour en tenir compte. Modifier ensuite les relations puis écrire quelques requêtes SQL pouvant être utilisées dans le cadre de ces extensions.