

ElectronJS - Examen : ElectronImmo

Vous allez créer une application Electron permettant de gérer des biens immobiliers, nommée "ElectronImmo"

Les utilisateurs pourront :

- consulter la liste des biens immobiliers
- ajouter des biens immobiliers
- réserver un bien immobilier

Des éléments d'interface graphique et de CSS sont demandés : vous êtes **libres** sur la manière dont vous les intégrez. Vous pouvez utiliser les librairies JS/CSS de votre choix ou bien aucune si vous voulez. Une note d'appréciation visuelle sera donnée : le visuel peut être très simple mais ce que vous faites doit être soigné.

Exercice 1 - Fenêtre principale (Total: 4 points)

1. Créez une fenêtre principale de format 1024x768 qui contiendra la liste des biens immobiliers. - **(1 point)**
2. Dans cette fenêtre, on doit avoir : - **(2 points)**
 - une **sidebar** contenant le titre de l'application "HBZoo" et un **lien** "Ajouter un bien immobilier"
 - Une liste des biens immobiliers présents sous la forme d'un tableau contenant tous les biens.
 - Ce tableau contiendra les colonnes suivantes :
 - Le nom du bien
 - "Location" ou "Vente" en fonction du type de bien
 - Ville
 - Superficie
 - Prix (afficher : xxx €/mois si c'est une location ou xxx € simplement si c'est une vente)
 - Sa description
 - Sa photo dont le format sera adapté au tableau (par exemple, limité en hauteur) (la photo doit venir d'un lien vers une photo sur Internet)
 - Un bouton cliquable avec son état de réservation ("réservé" ou "non réservé")
3. Cette fenêtre doit avoir une taille **maximale** de 1024x768 et une taille **minimale** de 1024x768 (on ne peut ni agrandir ni réduire la taille de la fenêtre). - **(1 point)**

Exercice 2 - Ajout d'un bien immobilier (Total: 6 points)

1. Vous devez faire en sorte que le bouton **Ajouter un bien immobilier** ouvre une nouvelle fenêtre de taille 800x600 qui contiendra le formulaire d'ajout d'un bien immobilier. - **(3 points)**
 - Sur le même principe que l'exercice 1.3., la taille doit être fixe.
 - Dans cette page, vous aurez :
 - Un titre de la page en H1
 - Des champs de formulaire permettant de saisir :
 - Le nom du bien
 - Un choix "Location" ou "Vente"

- Ville
- Superficie
- Prix
- Sa description
- Sa photo (la photo doit venir d'un lien vers une photo sur Internet)
- Un choix de son état de réservation ("réservé" ou "non réservé")
- Un bouton "Ajouter le bien immobilier"

Attention : les champs doivent avoir des types cohérents en fonction des données à saisir
<https://developer.mozilla.org/fr/docs/Web/HTML/Element/Input>

1. Lors du clic sur "Ajouter le bien immobilier", le bien immobilier doit être ajouté à un tableau qui sera stocké dans le LocalStorage. - **(2 points)**
2. La fenêtre "Ajout d'un bien immobilier" doit se refermer une fois le bien enregistré. - **(1 point)**

Exercice 3 - Affichage des données (Total: 4 points)

1. Vous devez afficher sur la page d'accueil les données issues du LocalStorage. - **(2 points)**
2. À chaque fois que l'on ajoute un bien immobilier, la page d'accueil doit toujours afficher la liste complète des biens immobiliers, incluant le nouveau bien. - **(2 points)**

Exercice 4 - Réserver un bien immobilier (4 points)

Les utilisateurs pourront, en consultant les biens sur la page d'accueil, cliquer sur le bouton "Réserver" si l'état du bien n'est pas déjà "réservé".

Une fois que le bouton "Reservation" a été cliqué, vous devez changer l'état du bien dont le bouton a été cliqué pour le changer sur "réservé".

Un bouton pour un bien dont l'état est "réservé" ne **peut pas** être cliquable.

- Un bouton devant chaque bien immobilier - (1 point)
- Le bouton est cliquable et génère une action - (1 point)
- Le bouton est non cliquable si réservé - (1 point)
- Le bouton change en non cliquable quand il est cliqué - (1 point)

Exercice 5 - Rendu du projet (Total: 2 points)

1. Appréciation du code : le code HTML/CSS/JS doit être propre et sans erreurs et bien indenté. L'aspect visuel doit être soigné (il peut être simpliste mais pas bâclé). Le nommage des variables doit être cohérent.
2. Vous devez rendre un fichier zippé contenant tous vos fichiers **SAUF** le dossier `node_modules`. Le fichier zippé devra être nommé comme suit : `hbzoo-prenom`. Le projet doit pouvoir être lancé grâce aux commandes `npm install` puis `npm start` (ou bien vous devez fournir la documentation des commandes à saisir si vous utilisez des librairies requérant d'autres commandes).