Soit les deux entités « Batiment » et « Niveau » suivante :

|  |
| --- |
| Batiment |
| * nom :string * nbr\_niveau : number * description :string * adresse :string |

|  |
| --- |
| Niveau |
| * nom :string * nbr\_chambre :number * etat\_construction  :boolean * id\_batiment :string |

Figure 1: Collection Bâtiment et Niveau

Vous êtes amené à créer une application web en utilisant les technologies : Nodejs/Express MongoDB et mongoose qui permet la gestion des constructions de bâtiments.

1. Créer la base de données "myexam2024" avec les deux entités Bâtiment et Niveau (voir figure n°1). (1pt)
2. Créer une fonction qui permet d’ajouter un bâtiment avec la propriété nbr\_niveau initialisé à 0 (voir figure n°2). (2 pts)





Figure 2: Ajouter un nouveau bâtiment

1. Créer une fonction (voir figure n°3) qui permet d’afficher la liste des bâtiments. (1 pt)



Figure 3: Liste des bâtiments

1. Créer une fonction (voir figure n°4) qui permet de récupérer un bâtiment via son Id. (1 pt)



Figure 4: Récupérer un bâtiment via son Id

1. Créer une fonction “Supprimer” (voir figure n°5) qui permet de supprimer un bâtiment via son Id. (1 pt) 

Figure 5 : Supprimer un bâtiment

1. Créer une fonction qui permet de créer un niveau (voir figure n°6) à un bâtiment via son « id », l’état\_construction du niveau est initialisé à « false ». (2 pts)



Figure 6: Ajouter un niveau

1. Élaborez une fonction nommée "construction" qui prend en paramètre l’"id" du niveau tels qu'illustrés dans la figure n°7. Cette fonction a pour but de mettre à jour l’état\_construction à **true** puis incrémenter à **1** le nombre de niveau du bâtiment auquel appartient ce niveau. (3 pts)

NB : Si l’état\_construction du niveau est à **true** il ne faut pas mettre à jour le nbr\_niveau du bâtiment.

 Voici un exemple illustrant ces actions :

      -**Bâtiment avant la construction** :   
      \_id : 2 nbr\_niveau :3 description : ‘Batiment de 3 niveau’ adresse : ‘Tunis’

- **Niveau avant la construction** :

\_id : 4 nom :’département Web’ etat\_construction : false id\_batiment : 2

-**Batiment après la construction** :

\_id : 2 nbr\_niveau :**4** description : ‘Batiment de 3 niveau’ adresse : ‘Tunis’

-**Niveau après la construction** :

\_id : 4 nom :’département Web’ etat\_construction : **true** id\_batiment : 2



Figure 7 : Lancer une construction

1. Créer une nouvelle page web (voir figure n°8) où on y trouvera une partie consacrée à l’affichage des informations des niveaux de bâtiments. (1 pts)

NB : créer un champ qui permet de saisir un id d’un niveau et un bouton « construction »



Figure 8: Création d’une nouvelle construction

1. Afficher la liste des niveaux **non bâti** (etat\_construction :**false** )(voir figure n°9) en utilisant Socket.io. (3 pts)

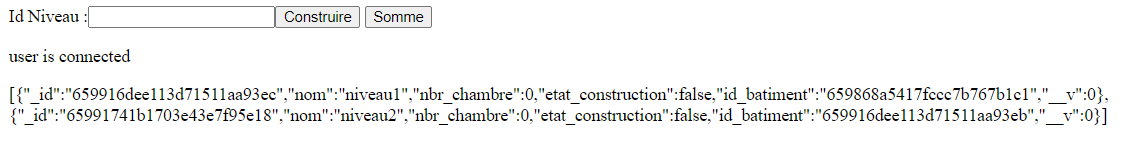


Figure 9 : Notification de la liste des niveaux

1. Lorsque rempli le champ par l’id d’un niveau non bâti et qu’on clique sur le bouton « construction » la fonction construction (question 7) va être exécuter puis on aura la nouvelle liste mis à jour des niveaux non bâti (voir figure n°10) en utilisant Socket.io. (3 pts)

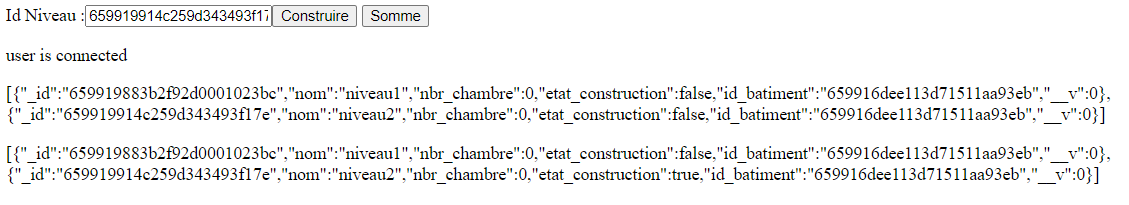


Figure 10 : Notification de la liste des niveaux

1. Créer une bouton « somme » qui permet de calculer la somme des bâtiments dont le nombre des niveaux est supérieur à **5** et dont l’adresse est à « **Tunis**». (2pts)

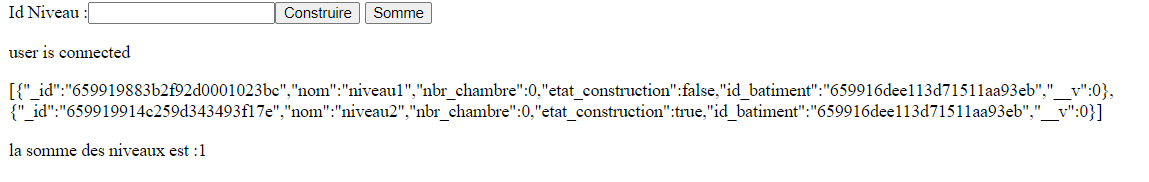


Figure 11: Calculer la somme des bâtiments

**Bon Travail ☺**