|  |
| --- |
| **Introduction SQL et NoSQL** |

Visualiser la vidéo et répondre aux questions ci-dessous.

Doc :<https://www.youtube.com/watch?v=BW3cdr5MjEE>

**Q1.**Donner une définition des bases de données NoSQL

La base de données NoSQL est une base sans schéma (données non structurées), il existe plusieurs types de base NoSQL et il n’y a pas de relation entre les données.

**Q2.** Quelles sont les avantages et les inconvénients des bases de données SQL

|  |  |
| --- | --- |
| SQL | |
| Avantages | Inconvénients |
| Données très structurées | Modèle de données trop rigide |
| Intégrité des données | Evolution souvent complexe à gérer |
| Langage de requête puissant | Plus il y a de données à traiter, plus c’est lent |

**Q3.** Quelles sont les avantages et les inconvénients des bases de données NoSQL

|  |  |
| --- | --- |
| NoSQL | |
| Avantages | Inconvénients |
| Modèle de données souple | Intégrité des données garantie par le programme |
| Vitesse d’exécution indépendante du nombre d’enregistrement | Requêtes parfois complexe |

**Q4.** Citer quelques services qui utilisent les bases de données NoSQL.

Parmi les services utilisant les bases de données NoSQL, il y a par exemple :

* mongoDB
* DynamoDB
* Cloud Firestore
* Etc…

|  |
| --- |
| **Introduction Firebase** |

Firebase est un ensemble de services d'hébergement pour n'importe quel type d'application (Android, iOS, Javascript, Node.js, Java, Unity, PHP, C++ ...). Il propose d'héberger en NoSQL et en temps réel des bases de données, du contenu, de l'authentification sociale (Google, Facebook, Twitter et Github), et des notifications.

**Q5.** Aller sur le lien ci-dessous et expliquer les différents services dans le tableausuivant :

Doc : <https://firebase.google.com/products-build>

|  |  |
| --- | --- |
| **Services** | **Explication** |
| Cloud Firestore | Cloud Firestore est une base de données de documents NoSQL qui vous permet de stocker, synchroniser et interroger facilement des données pour vos applications mobiles et Web, à l'échelle mondiale. |
| realtime-database | La base de données FirebaseRealtime est une base de données NoSQL hébergée dans le cloud qui vous permet de stocker et de synchroniser des données entre vos utilisateurs en temps réel. |
| remote-config | Avec FirebaseRemote Config, vous pouvez modifier le comportement et l'apparence de votre application à la volée à partir de la console Firebase, puis suivre les performances dans [Google Analytics pour Firebase](https://firebase.google.com/products/analytics) . |
| Authentification | FirebaseAuthentication vise à faciliter la création de systèmes d'authentification sécurisés, tout en améliorant l'expérience de connexion et d'intégration pour les utilisateurs finaux. |
| cloud-messaging | Firebase Cloud Messaging (FCM) fournit une connexion fiable et efficace sur batterie entre votre serveur et vos appareils, qui vous permet de transmettre et de recevoir gratuitement des messages et des notifications sur iOS, Android et le Web. |
| hosting | Avec FirebaseHosting, vous pouvez déployer une application Web d'une seule page, une page de destination d'application mobile ou une application Web progressive sans tous les tracas. |

**Q6.**Faire une comparaison entrerealtime-database et Cloud Firestore et remplir le tableau

Lien: <https://firebase.google.com/pricing>

|  |  |  |
| --- | --- | --- |
|  | Cloud Firestore | realtime-database |
| Stockage | 0.18$/GB | 5$/GB |
| Ecriture | 0.18$/100000 | **Non défini** |
| Lecture | 0.06$/100000 | **Non défini** |
| Suppression | 0.02$/100000 | **Non défini** |
| Connexion | **Non défini** | **Non défini** |

Pour plus de comparaison technique voir le lien suivant:

<https://firebase.google.com/docs/database/rtdb-vs-firestore>

**Q7.** Voir le lien ci-dessous et expliquer ce que veut dire firebase SDK

<https://firebase.google.com/docs/admin/setup>

Le firebaseSDK Admin est un ensemble de bibliothèques de serveurs qui vous permet d'interagir avec firebase à partir d'environnements privilégiés pour effectuer des actions.

**Q8.** Coller ci-dessous le SDK correspondant au applications WEB javascript

<https://firebase.google.com/docs/web/setup#from-the-cdn>

<body>  
  <!-- Insert these scripts at the bottom of the HTML, but beforeyou use anyFirebase services -->  
  
  <!-- Firebase App (the coreFirebase SDK) isalwaysrequired and must belisted first -->  
  <scriptsrc="https://www.gstatic.com/firebasejs/8.3.2/firebase-app.js"></script>  
  
  <!-- If youenabled Analytics in yourproject, add the Firebase SDK for Analytics -->  
  <scriptsrc="https://www.gstatic.com/firebasejs/8.3.2/firebase-analytics.js"></script>  
  
  <!-- AddFirebaseproductsthatyouwant to use -->  
  <scriptsrc="https://www.gstatic.com/firebasejs/8.3.2/firebase-auth.js"></script>  
  <scriptsrc="https://www.gstatic.com/firebasejs/8.3.2/firebase-firestore.js"></script>  
</body>

// For Firebase JavaScript SDK v7.20.0 and later, `measurementId` is an optionalfield  
varfirebaseConfig={  
  apiKey:**"API\_KEY",**  
  authDomain:**"PROJECT\_ID.firebaseapp.com",**  
  databaseURL:**"https://PROJECT\_ID.firebaseio.com",**  
  projectId:**"PROJECT\_ID",**  
  storageBucket:**"PROJECT\_ID.appspot.com",**  
  messagingSenderId:**"SENDER\_ID",**  
  appId:**"APP\_ID",**  
  measurementId:**"G-MEASUREMENT\_ID",**  
};

|  |
| --- |
| **Prise en main Firebase** |

**Etape 1 :** aller sur le lien suivant et connectez-vous avec votre compte google.

<https://firebase.google.com/?hl=fr>

**Etape 2:** suivre la vidéo ci-dessous de **0s à 1:47s**

**Etape 3:** Choisir une application WEB.

|  |
| --- |
|  |

**Etape4:** Donner un nom à votre application

|  |
| --- |
|  |

**Etape 5:** Enregistrer votre SDK qui s’affiche sous la forme suivante :

|  |
| --- |
| **<!-- Insert these scripts at the bottom of the HTML, but beforeyou use anyFirebase services -->**  **<!--Firebase App (the coreFirebase SDK) isalwaysrequired and must belisted first -->**  **<scriptsrc="https://www.gstatic.com/firebasejs/8.3.1/firebase-app.js"></script>**  **<!-- If youenabled Analytics in yourproject, add the Firebase SDK for Analytics -->**  **<scriptsrc="https://www.gstatic.com/firebasejs/8.3.1/firebase-analytics.js"></script>**  **<!--AddFirebaseproductsthatyouwant to use -->**  **<scriptsrc="https://www.gstatic.com/firebasejs/8.3.1/firebase-auth.js"></script>**  **<scriptsrc="https://www.gstatic.com/firebasejs/8.3.1/firebase-firestore.js"></script>**  **// For Firebase JavaScript SDK v7.20.0 and later, `measurementId` is an optionalfield**  **varfirebaseConfig = {**  **apiKey:"AIzaSyDOCAbC123dEf456GhI789jKl01-MnO",**  **authDomain:"myapp-project-123.firebaseapp.com",**  **databaseURL:"https://myapp-project-123.firebaseio.com",**  **projectId:"myapp-project-123",**  **storageBucket:"myapp-project-123.appspot.com",**  **messagingSenderId:"65211879809",**  **appId:"1:65211879909:web:3ae38ef1cdcb2e01fe5f0c",**  **measurementId:"G-8GSGZQ44ST"**  **};** |

Mon SDK :

<!-- The core Firebase JS SDK is always required and must be listed first -->  
<script src="https://www.gstatic.com/firebasejs/8.5.0/firebase-app.js"></script>  
  
<!-- TODO: Add SDKs for Firebase products that you want to use  
     https://firebase.google.com/docs/web/setup#available-libraries -->  
<script src="https://www.gstatic.com/firebasejs/8.5.0/firebase-analytics.js"></script>  
  
<script>  
  // Your web app's Firebase configuration  
  // For Firebase JS SDK v7.20.0 and later, measurementId is optional  
  var firebaseConfig = {  
    apiKey: "AIzaSyCB2rYIwGzSYhDmqe5LdJHxQI1l-Bi-QLk",  
    authDomain: "fir-hamaoduche.firebaseapp.com",  
    projectId: "fir-hamaoduche",  
    storageBucket: "fir-hamaoduche.appspot.com",  
    messagingSenderId: "596586232948",  
    appId: "1:596586232948:web:a2f7f2c43f1e1bc9e52521",  
    measurementId: "G-43LLBYPMSC"  
  };  
  // Initialize Firebase  
  firebase.initializeApp(firebaseConfig);  
  firebase.analytics();  
</script>

**Étape 6:** Dans cette séance on utilisera le cloud firestore pour stocker nos données.

<https://www.youtube.com/watch?v=KeTGxgcAx1g>

**--------------------STOP ICI ----------------------------------**

**Q9.** Firestore utilise un code pour paramétrer les règles de sécurité, Expliquer le code ci dessous en utilisant la documentation de firebase :

doc: <https://firebase.google.com/docs/rules/rules-language?authuser=0>

|  |
| --- |
| rules\_version = '2';  service cloud.firestore {    match /databases/{database}/documents {      match /{document=\*\*} {        allow read, write: if            request.time < timestamp.date(2021, 5, 1);      }    }  } |

**Q10.** Faire le tutoriel suivant de   0 à 6:48 et expliquer ce que veut dire document et collection .

<https://www.youtube.com/watch?v=2Vf1D-rUMwE&t=700s>

**Q11:** Que fait le code suivant ?

|  |
| --- |
| docRef.set({  hotDogStatus: textToSave     }).then(function(){       console.log("ok");   }).catch(function(error){       console.log(error)   });  }); |

Si votre rule (code d’autorisation est bon vous aurez ces données)

|  |
| --- |
| https://lh5.googleusercontent.com/xuTY_xoCRaNCxTMsacIxMkcHPXAntn1JA4RoaPw-PLg9oqca6k1WquMxM2HhTIiPKut-qrbS6HQ6piYPu25DoOb9uY939ZrcHylOV19MsPBPOgucNxaMTWj2p73Hez6givFCuPSs |

**Continuer la vidéo jusqu'à 9:33**

**Q12:** Remplacer ce code

|  |
| --- |
| if(doc && doc.exists){     const myData=doc.data();     outputHeaser.innerText= myData.hotDogStatus;  } |

par le code suivant :

|  |
| --- |
| if(doc && doc.exists){     const myData=doc.data();     console.log(myData);  } |

Afficher la réponse (inspecter-> console)

**Q13.**Quelle est la forme de cette réponse ?

**Q14.** Que représente cette donnée  myData.hotDogStatus

**Continuer la vidéo**

**Q15.**Quelle est la différence entre l’ancien programme et le programme suivant:

|  |
| --- |
| **docRef.onSnapshot(function(doc){**  **if(doc && doc.exists){**  **const myData=doc.data();**  **outputHeaser.innerText= myData.hotDogStatus;**  **}**  **})** |

**Q16. Voir la documentation de firebase et expliquer les différents méthode ci-dessous:**

**Documentation**

**https://firebase.google.com/docs/firestore/manage-data/add-data**

|  |  |
| --- | --- |
| **fonction** | **Description** |
| **doc().add()** |  |
| **doc().set()** |  |
| **doc().update()** |  |
| **doc().delete()** |  |
| **doc().onSnapshot()** |  |

**--------------------------------POURSUIVE ICI ----------------**

|  |
| --- |
| **Afficher les personnes** |
|  |

**Q16.** Aller sur votre base de données précédente et supprimer son contenu

ensuite, créer votre base de données des personnes manuellement sous la forme ci dessous.

|  |
| --- |
| **var personnes =[ //Votre collection**  **{//Document 1**  **id:"0",**  **prenom: "Jules",**  **Nom: "Ferry",**  **age: 50**  **},**    **{//Document 2**  **id:"1",**  **prenom: "gerard",**  **Nom: "briaud",**  **age: 30**  **},**    **{ // Document 3**  **id:"2",**  **prenom: "Alexandre",**  **Nom: "Dumas",**  **age: 28**  **}**      **];** |

|  |
| --- |
| **https://lh5.googleusercontent.com/3grh415RrBRPbHXUMRhgs_umAJ1XyWVqmDZ7Imwr_hdfrs3UQY29K5I3rMB6t45w0BcK5J5R6wCSzhtqY5ljmBuM-WzO-1BWH3KdtH0AYjUF-h7IeMYXqqQxVVn7Y8XgL45KLB_N** |

**Créer un dossier et mettre le code ci-dessous.**

|  |  |  |
| --- | --- | --- |
| **html** | **css** | **js** |
| **<html>**  **<head>**  **<script src="https://www.gstatic.com/firebasejs/5.0.3/firebase-app.js"></script>**  **<script src="https://www.gstatic.com/firebasejs/5.0.3/firebase-firestore.js"></script>**  **<link rel="stylesheet" href="styles.css">**  **</head>**  **<body>**    **<h1>Cloud Personnes</h1>**    **<div class="content">**    **<form id="ajouter">**  **<input type="text" name="nom" placeholder="nom">**  **<input type="text" name="prenom" placeholder="prenom">**  **<input type="text" name="age" placeholder="age">**  **<button>Ajouter</button>**  **</form>**    **<ul id="list"></ul>**    **</div>**    **<script>**  **//---------------------  Firebase TOKEN -------------------------------**  **var firebaseConfig = {**  **apiKey: "AIzaSyD5-yEaIH1VsSiQ9u-Si0HbFBdzj\_owdsk",**  **authDomain: "training-98e50.firebaseapp.com",**  **projectId: "training-98e50",**  **storageBucket: "training-98e50.appspot.com",**  **messagingSenderId: "827450532534",**  **appId: "1:827450532534:web:c9dce6d7ea222dcf5abc17",**  **measurementId: "G-G0ZSTH1FQ6"**  **};**  **//---------------------------------------------------------------------------------------**        **//----------------- FIREBASE INITIALISATION -------------------------------------------------**  **firebase.initializeApp(firebaseConfig);**  **const db = firebase.firestore();**  **db.settings({ timestampsInSnapshots: true });**  **//------------------------------------------------------------------------------------**  **</script>**  **<script src="app.js"></script>**  **</body>**  **</html>** | **body{**  **background: #615596;**  **font-family: ubuntu;**  **}**    **h1{**  **color: #fff;**  **font-size: 64px;**  **letter-spacing: 2px;**  **margin-top: 80px;**  **text-align: center;**  **}**    **.content{**  **background: #fff;**  **max-width: 960px;**  **margin: 30px auto;**  **padding:  20px 30px;**  **border-radius: 10px 120px 10px 10px;**  **box-shadow: 1px 3px 5px rgba(0,0,0,0.1)**  **}**    **ul{**  **list-style-type: none;**  **background: #dad0e6;**  **padding: 0;**  **}**    **li{**  **padding: 20px;**  **background: #dad0e6;**  **font-size: 20px;**  **color: #555;**  **position: relative;**  **border-bottom: 1px solid #e6e6e6;**  **height: 46px;**  **}**    **li:nth-child(even){**  **padding: 20px;**  **background: #dad0e6;**    **}**    **li div{**  **position: absolute;**  **top: 0;**  **right: 0px;**  **color: rgb(255, 252, 252);**  **background: rgb(214, 13, 13);**  **width: 40px;**  **text-align: center;**  **padding: 10px 0;**  **font-weight:  bold;**  **cursor:  pointer;**  **}**    **form input{**  **float: left;**  **width: 20%;**  **margin: 0;**  **border: 0;**  **border-bottom: 1px solid #eee;**  **margin: 0 1%;**  **padding: 10px;**  **display: block;**  **box-sizing: border-box;**  **font-size: 18px;**  **}**    **form input:focus{**  **outline: none;**  **border-bottom: 3px solid #88236f;**  **padding-bottom: 8px;**  **transition: all ease 0.2s;**  **}**    **form:after{**  **content: '';**  **clear: both;**  **display: block;**  **}**    **button{**  **border: 0;**  **background: rgb(214, 13, 13);**  **border-radius: 10px;**  **padding: 13px;**  **width: 14%;**  **box-shadow: -1px 0px 1px rgba(0,0,0,0.1);**  **font-weight: bold;**  **font-family: ubuntu;**  **letter-spacing: 1px;**  **color: rgb(255, 252, 252);**  **cursor:  pointer;**  **}** | **//--------------------------fonction GET DATA ----------------------------------**  **db.collection('personnes').get().then(**  **(mydata)=>{console.log(mydata.docs)**  **}**  **);**  **//--------------------------------------------------------------------------------** |

**Q17.** Afficher la réponse sur la console. Quelle est la forme de ces données ?

Utiliser le code ci-dessous pour afficher les données un par un.

|  |
| --- |
| //--------------------------fonction GET DATA ----------------------------------  db.collection('personnes').get().then(    (mydata)=>{     mydata.docs.forEach(doc=>{       console.log(doc)     })     });  //-------------------------------------------------------------------------------- |

**Q18.** Pour avoir les données réels il faut utiliser la méthode **data() comme suit**

|  |
| --- |
| console.log(doc.data()) |

**faire une capture d’écran de ces données.**

Pour afficher les données dans la page html il faut utiliser le code javascript ci-dessous

|  |
| --- |
| const ul = document.querySelector('#list'); // Seléctionner l'élément list sur le document HTML      //---------------------------Fonction Affficher les données -------------------------------------------------  function affiche(doc){    let li = document.createElement('li'); // Créer une balise li    let supprimer = document.createElement('div'); // créer une balise div      li.setAttribute('data-id', doc.id) // donner l'identifiant de la personne comme attribut a la liste    li.textContent=doc.data().Nom + ' ' + doc.data().prenom + ' ' +doc.data().age + ' , id=  ' + doc.id //mettre les données dans la liste    supprimer.textContent='X' // Ecrite dans la balise supprimer      li.appendChild(supprimer) // intégrer le 'x' dans la balise li      ul.appendChild(li);//intégrer la balise li dans la balise ul présente sur le html    }  //----------------------------------------------------------------------------------------------------    //--------------------------fonction GET DATA ----------------------------------  db.collection('personnes').get().then(    (mydata)=>{     mydata.docs.forEach(doc=>{        affiche(doc)     })       });  //-------------------------------------------------------------------------------- |

**Enregistrer votre travail ou faire un git commit afin de le sauvegarder.**

**Q19.**Ajouter ce code a la fin de la fonction afficher ensuite commenter .

|  |
| --- |
| **liid=li.getAttribute("data-id")**  **alert('li id = ' + liid)** |

|  |
| --- |
| **Ajouter des personnes** |

**Pour ajouter des éléments tester le javascript ci-dessous:**

|  |
| --- |
| **db.collection('personnes').add({**  **Nom: "juan",**  **prenom: "Rodrigez",**  **age: 48**  **});** |

**Q20.** Quelle “id” a pris votre document ajouté et pourquoi ?

**Documentation**

[**https://firebase.google.com/docs/firestore/manage-data/add-data**](https://firebase.google.com/docs/firestore/manage-data/add-data)

**Pour automatiser le bouton il faut tester le code ci-dessous:**

|  |
| --- |
| **const form = document.querySelector('#ajouter'); //selectionné l'élément form du HTML**    **form.addEventListener('submit', (e) => { //Actionner l'évenement submit de la form ajouter**  **e.preventDefault(); // Empêcher l'exécution de submit pour laisser la fonction s'exécuter**    **alert(form.nom.value)**    **});** |

**1.Écrire un nom**

**2.Cliquer sur ajouter**

|  |
| --- |
| **https://lh6.googleusercontent.com/NmCRhXH2YaHLR0ScknfJ4ENV3FLz_re9oYun9oFdrhoD599lID5ravGUfgV0q5WxUNBqVM8Gqzdh5v1g2xkRLQeSvWCIolfbpG4z0BGBnkVsoVmzEBXBfnRintZLIwucUVoWRYIl** |

**Q21.**Expliquer le fonctionnement de ce code.

Pour ajouter toutes les informations à la base de données il faut utiliser le code ci-dessous.

|  |
| --- |
| //--------------------------------------function ADD DATA --------------------------------------  form.addEventListener('submit', (e) => { //Actionner l'évenement submit de la form ajouter    e.preventDefault(); // Empécher l'éxécution de submmit pour laisser la fonction firebase s'éxécuter    db.collection('personnes').add({     //----mettre les données de la forme dans les variable à ajouter------      Nom: form.nom.value,      prenom: form.prenom.value,      age: form.age.value    });    //------Vider les formes ------    form.nom.value="";    form.prenom.value="";    form.age.value="";  });  //-------------------------------------------------------------------------------------------------- |

**regarder sur votre base de données si l’élément a bien été ajouté ensuite rafraîchir la page.**

**Remarque:** On verra ultérieurement comment ajouter l’élément sans rafraîchir la page en temp réel

**Q22.**Ajouter le code javascript qui permet d’ajouter des personnes à celui qui permet d’afficher les personnes et coller l’ensemble ci-dessous.

|  |
| --- |
|  |

**Enregistrer votre travail ou faire un git commit pour le sauvegarder**

|  |
| --- |
| **Supprimer  des personnes** |

Pour supprimer un élément, il faut obtenir son identifiant.

Les identifiant sont attribués précédemment dans la fonction **affiche()**

|  |
| --- |
| **li.setAttribute('data-id', doc.id)** |

**étape 1: Ajouter un événement à la div qui permet de supprimer l’élément.**

**Ajouter le code à la fin de la fonction affiche comme suit:**

|  |
| --- |
| **function affiche(doc){**    **let li = document.createElement('li'); // Créer une balise li**  **let supprimer = document.createElement('div'); // créer une balise div**    **li.setAttribute('data-id', doc.id) // donner l'identifiant de la personne comme attribut a la liste**  **li.textContent=doc.data().Nom + ' ' + doc.data().prenom + ' ' +doc.data().age + ' , id=  ' + doc.id //mettre les données dans la liste**  **supprimer.textContent='X' // Ecrite dans la balise supprimer**    **li.appendChild(supprimer) // intégrer le 'x' dans la balise li**    **ul.appendChild(li);//intégrer la balise li dans la balise ul présente sur le html**    **//-------------function delete data -----------------------**  **supprimer.addEventListener('click',(e)=>{**  **alert('Click DONE')**    **})**  **//-------------------------------------------------------------------**    **}** |

**etape 2:** il faut obtenir l’identifiant de l’élément cliqué

remplacer le code**alert('Click DONE')** par ce code ci dessous.

|  |
| --- |
| **let id = e.target.parentElement.getAttribute('data-id');**  **alert(id)** |

**Q23.** expliquer le rôle de la méthode **e.target**

**Documentation 1/target**

[**https://developer.mozilla.org/fr/docs/Web/API/Event/target**](https://developer.mozilla.org/fr/docs/Web/API/Event/target)

**Documentation 2/target**

[**https://www.w3schools.com/jsref/tryit.asp?filename=try\_dom\_event\_target**](https://www.w3schools.com/jsref/tryit.asp?filename=try_dom_event_target)

**Q24.** il faut savoir que l’élément **‘div’** cliqué est à l'intérieur de la balise **‘li’**

qui obtient l’identifiant de la donnée

|  |
| --- |
| https://lh4.googleusercontent.com/kJ3VZHFCnKRGgYzOiwx1KbaXVdIivh3NNXR0FzbEX5yuV4YQKw83E4d0JD9OZMNpBOVVAx0oq7F-lgQPeLyEr9LVaGVxB9EZSbjjkGXAo0eRzw3veEB1QTIbRKBxZtCQOi0k6eiV |

Expliquer le rôle de la méthode parentElement.getAttribute('data-id')

**Documentation 1/parentElement**

[**https://developer.mozilla.org/fr/docs/Web/API/Node/parentElement**](https://developer.mozilla.org/fr/docs/Web/API/Node/parentElement)

**Documentation 2/parentElement**

[**https://www.w3schools.com/jsref/tryit.asp?filename=tryjsref\_node\_parentelement**](https://www.w3schools.com/jsref/tryit.asp?filename=tryjsref_node_parentelement)

**Documentation 3/getAttribute**

[**https://developer.mozilla.org/fr/docs/Web/API/Element/getAttribute**](https://developer.mozilla.org/fr/docs/Web/API/Element/getAttribute)

**Documentation 4/getAttribute**

[**https://www.w3schools.com/jsref/tryit.asp?filename=tryjsref\_node\_parentelement**](https://www.w3schools.com/jsref/tryit.asp?filename=tryjsref_node_parentelement)

**Q25.** Réaliser le code qui permet de supprimer cette élément

**Documentation:**

[**https://firebase.google.com/docs/firestore/manage-data/delete-data**](https://firebase.google.com/docs/firestore/manage-data/delete-data)

**il faut rafraichir la page après la suppression de l’élément**