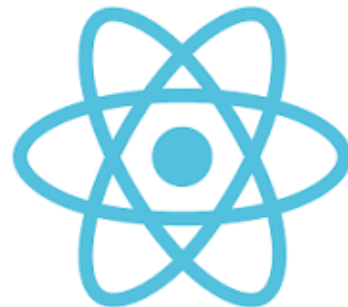


Agencia de
Aprendizaje
a lo largo
de la vida

ESPECIALIZACIÓN REACT



Agencia de
Aprendizaje
a lo largo
de la vida

Clase 14. REACT JS

CRUD FIREBASE 1

CRUD

CRUD

Si no conoces `Firestore` no te preocupes, aquí viene un ejemplo para practicar. `Firestore` es un servicio de `Firebase` (creado por Google) para almacenar `collection` (base de datos no relacionales) y utilizarlas de forma gratuita .

CREAR PROYECTO

Crear Proyecto

Dirigete a <https://firebase.google.com/> (opens new window)accede con alguna cuenta de Gmail y listo!

Pinchamos en añadir proyecto

Nos vamos a database

Comenzamos en modo de prueba

Listo

#Configurar

CREAR PROYECTO

Crear Proyecto

Dirigete a <https://firebase.google.com/> (opens new window)accede con alguna cuenta de Gmail y listo!

Pinchamos en añadir proyecto

Nos vamos a database

Comenzamos en modo de prueba

Listo

#Configurar

CONFIGURAR

En este ejemplo utilizaremos un proyecto limpio de React, pero es importante instalar Firebase con:

```
npm i firebase
```

Luego crear un archivo en la carpeta src llamado firebase.js

```
import firebase from 'firebase/app'
import 'firebase/firestore'

const firebaseConfig = {
  apiKey: "xxx",
  authDomain: "xxx",
  databaseURL: "xxx",
  projectId: "xxx",
  storageBucket: "xxx",
  messagingSenderId: "xxx",
  appId: "xxx"
};

// Initialize Firebase
firebase.initializeApp(firebaseConfig);

export {firebase};
```

CONFIGURAR

En este ejemplo utilizaremos un proyecto limpio de React, pero es importante instalar Firebase con:

```
npm i firebase
```

Luego crear un archivo en la carpeta src llamado firebase.js

```
import firebase from 'firebase/app'
import 'firebase/firestore'

const firebaseConfig = {
  apiKey: "xxx",
  authDomain: "xxx",
  databaseURL: "xxx",
  projectId: "xxx",
  storageBucket: "xxx",
  messagingSenderId: "xxx",
  appId: "xxx"
};

// Initialize Firebase
firebase.initializeApp(firebaseConfig);

export {firebase};
```

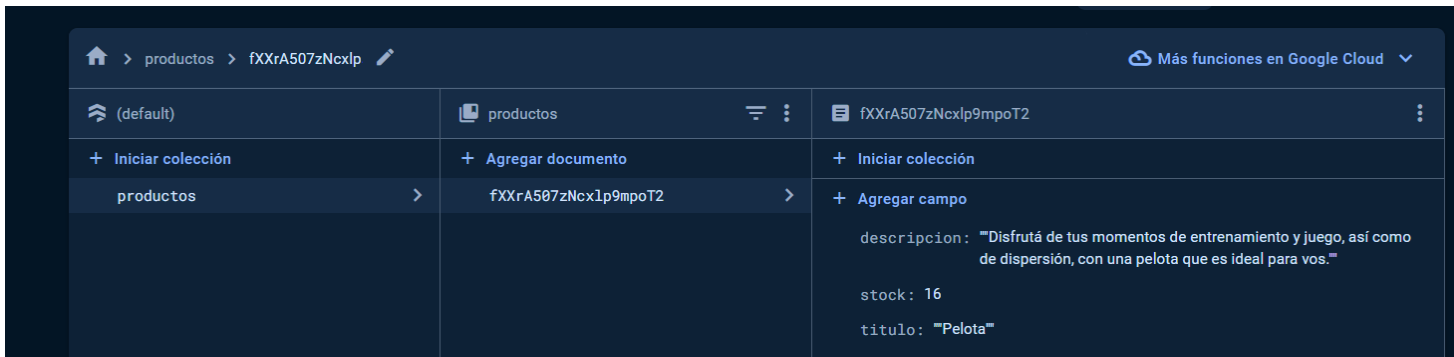

Bootstrap

Podemos seguir utilizando Bootstrap de una manera sencilla utilizando su CDN:

```
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh" crossorigin="anonymous">
```

Database

En la consola de firebase configuraremos una nueva colección:



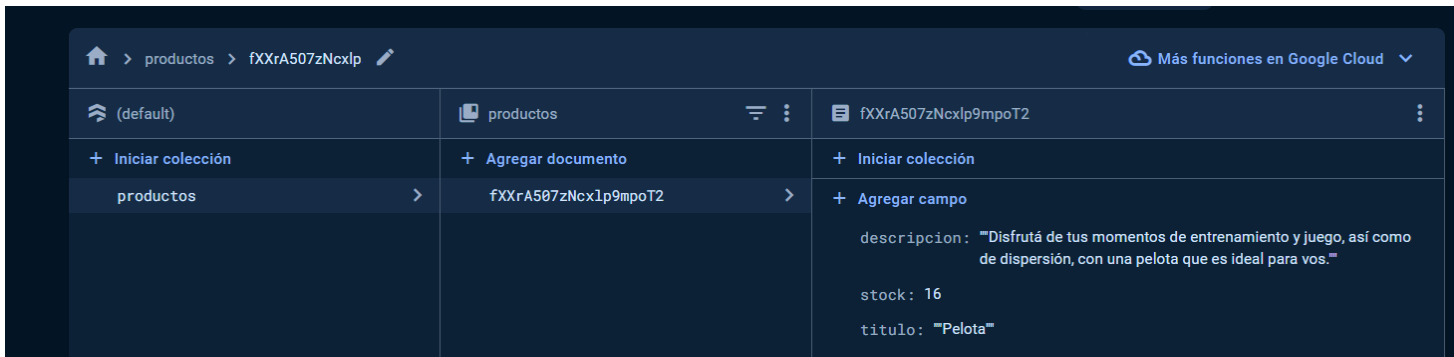
Bootstrap

Podemos seguir utilizando Bootstrap de una manera sencilla utilizando su CDN:

```
<link rel="stylesheet" href="https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css" integrity="sha384-Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh" crossorigin="anonymous">
```

Database

En la consola de firebase configuraremos una nueva colección:



LEER documentos

https://firebase.google.com/docs/firestore/quickstart?authuser=0#read_data

```
import {firebase} from './firebase'
```

```
React.useEffect(() => {  
  
  const obtenerDatos = async () => {  
    const db = firebase.firestore()  
    try {  
      const data = await db.collection('tarefas').get()  
      const arrayData = data.docs.map(doc => ({id: doc.id, ...doc.data()}))  
      console.log(arrayData)  
    } catch (error) {  
      console.log(error)  
    }  
  }  
  obtenerDatos()  
  
}, [])
```

Ahora con State

```
const [tareas, setTareas] = React.useState([])

React.useEffect(() => {

  const obtenerDatos = async () => {
    const db = firebase.firestore()
    try {
      const data = await db.collection('tareas').get()
      const arrayData = data.docs.map(doc => ({id: doc.id, ...doc.data()}))
      console.log(arrayData)
      setTareas(arrayData)
    } catch (error) {
      console.log(error)
    }
  }

  obtenerDatos()

}, [])
```

Ahora con State

```
<div className="container mb-2">
  <div className="row">
    <div className="col-md-6">
      <h3>Lista de Tareas</h3>
      <ul className="list-group">
        {
          tareas.map(item => (
            <li className="list-group-item" key={item.id}>
              <span>{item.name}</span>
              <button
                className="btn btn-danger btn-sm float-right"
              >
                Eliminar
              </button>
              <button
                className="btn btn-warning btn-sm float-right mr-2"
              >
                Editar
              </button>
            </li>
          ))
        }
      </ul>
    </div>
    <div className="col-md-6">
      formulario
    </div>
  </div>
</div>
```

AGREGAR documentos

<https://firebase.google.com/docs/firestore/manage-data/add-data?authuser=0>

```
const [tarea, setTarea] = React.useState('')

const agregar = async (e) => {
  e.preventDefault()
  if(!tarea.trim()){
    console.log('sin texto')
    return
  }
  console.log(tarea)
}
```

```
<h3>Formulario</h3>
<form onSubmit={agregar}>
  <input
    type="text"
    className="form-control mb-2"
    placeholder='Ingrese Tarea'
    value={tarea}
    onChange={e => setTarea(e.target.value)}
  />
  <button
    type='submit'
    className="btn btn-dark btn-block btn-sm"
  >
    Agregar
  </button>
</form>
```

AGREGAR documentos

<https://firebase.google.com/docs/firestore/manage-data/add-data?authuser=0>

```
try {  
  const db = firebase.firestore()  
  const nuevaTarea = {  
    name: tarea,  
    fecha: Date.now()  
  }  
  const data = await db.collection('tareas').add({  
    name: tarea,  
    fecha: Date.now()  
  })  
  setTareas([  
    ...tareas,  
    {id: data.id, ...nuevaTarea }  
  ])  
  setTarea('')  
} catch (error) {  
  console.log(error)  
}
```