In this activity, you will have a backend server for your own database.

Sounds cool, right?
Right?
Alright then, let's start.

You can attain backend and database servers in many ways. Technically, if you use NodeJS and MySQL or MongoDB or other databases, you will need to install *node*, *npm,* and that database of your choice. You will need to code a backend server that connects to that database server. And if you are not using a database server from the provider such as Atlas for MongoDB, you will also have to get your database up on *localhost* or a server such as *DigitalOcean.*

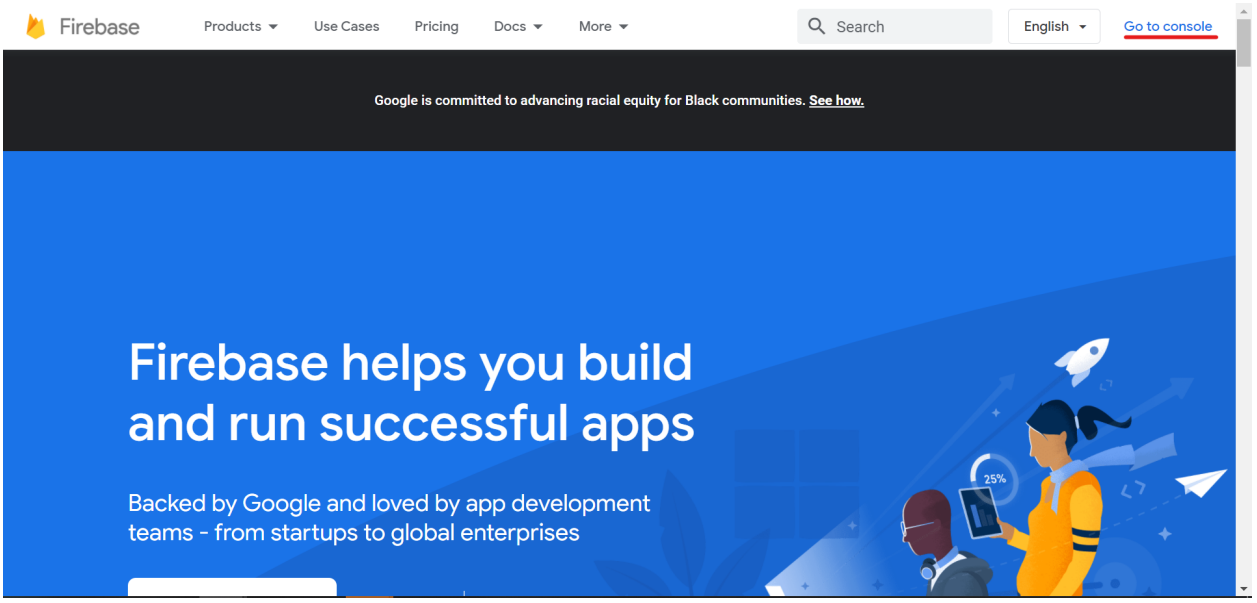
Luckily, you were born in the right generation! There are many X-as-a-service (XaaS) for you to get started with your project lightning fast⚡.

Table of Contents

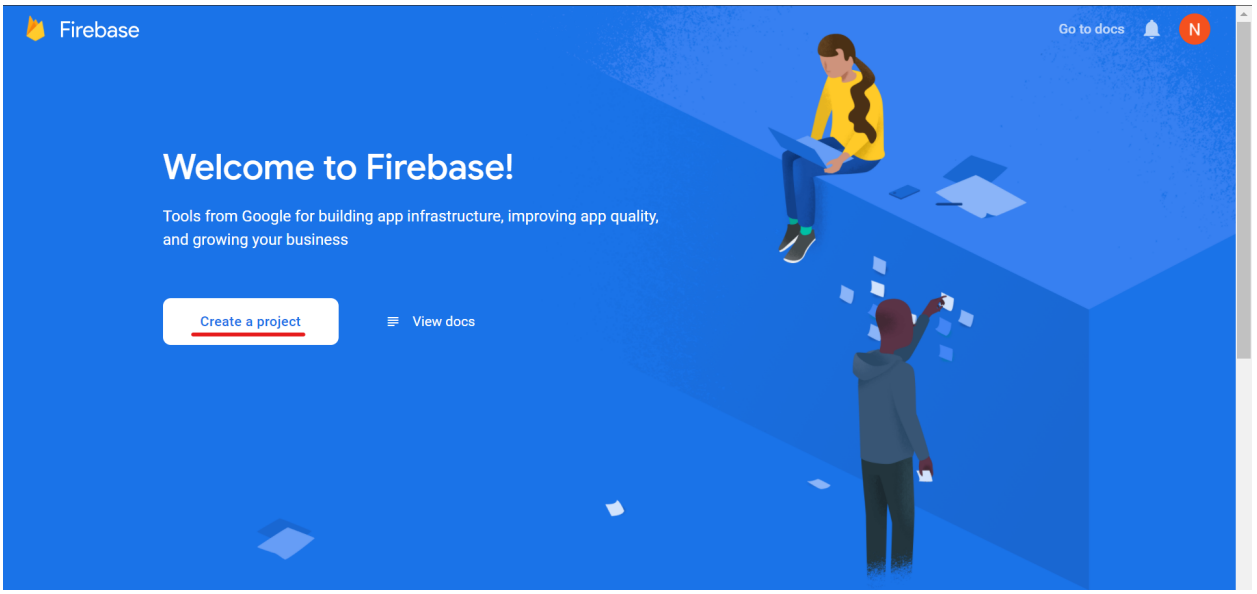
Create a Firebase Project

Today, you will use one of the easiest, **Firebase**🔥! Firebase is a backend-as-a-service (BaaS, thank me later on OS course). Shortly, to create a database and backend servers in BaaS, you'll "click" instead of "code."

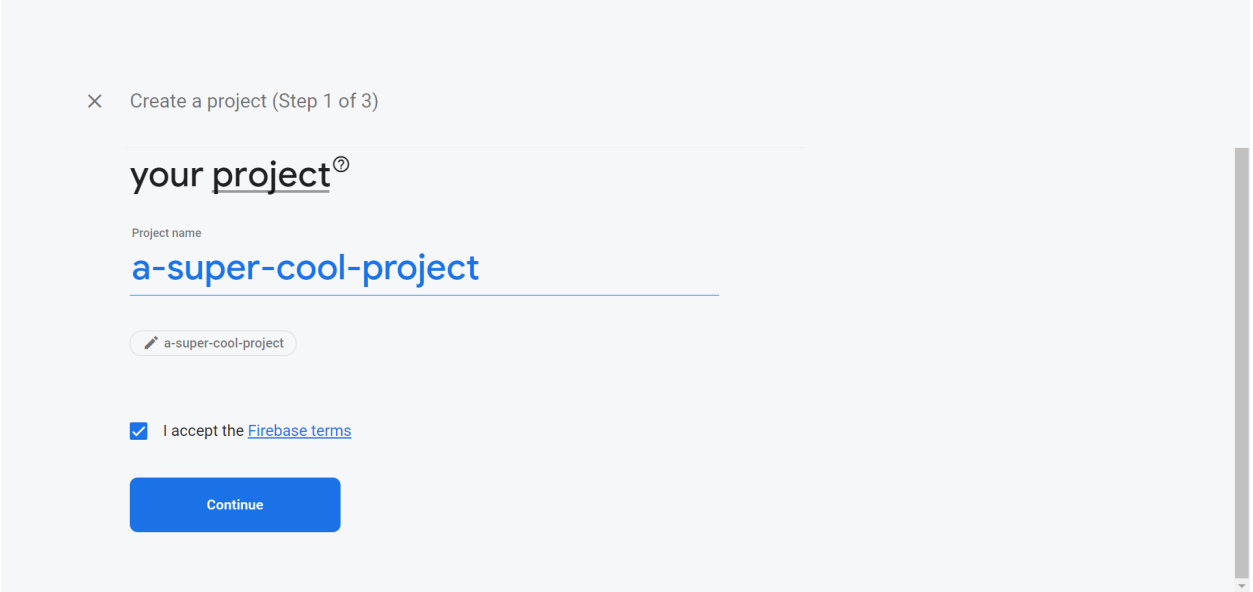
First, go to [the Firebase Official website](). Then, click "Go to console" in the upper right corner.



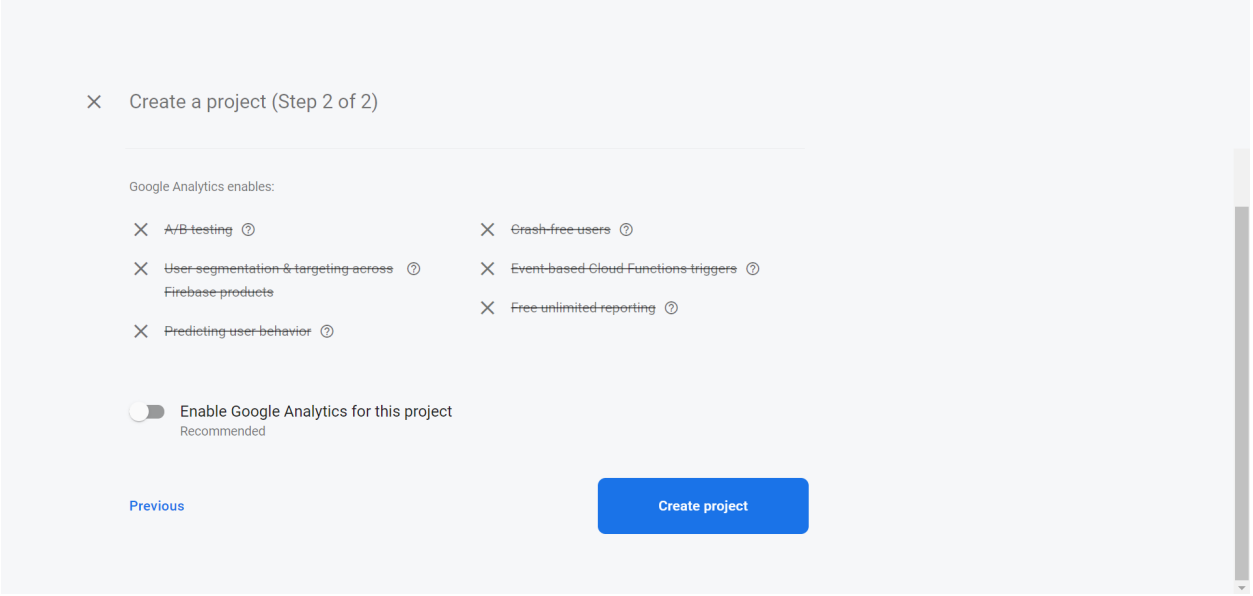
Second, create a new project. (You can "View docs" to see the full detail of Firebase).



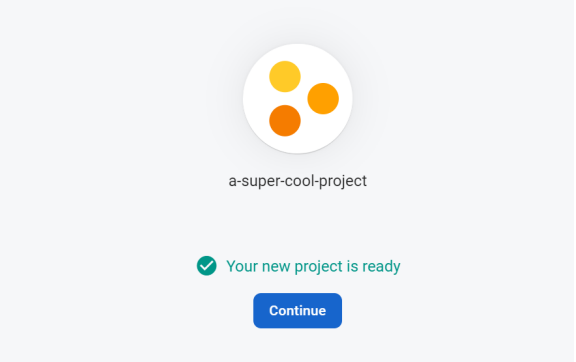
Third, name your project. (Sorry for making you type some inputs, at least it is not a code)



Fourth, disable Google Analytics. Google Analytics provides you insights of your website, but you do not need that in this project.

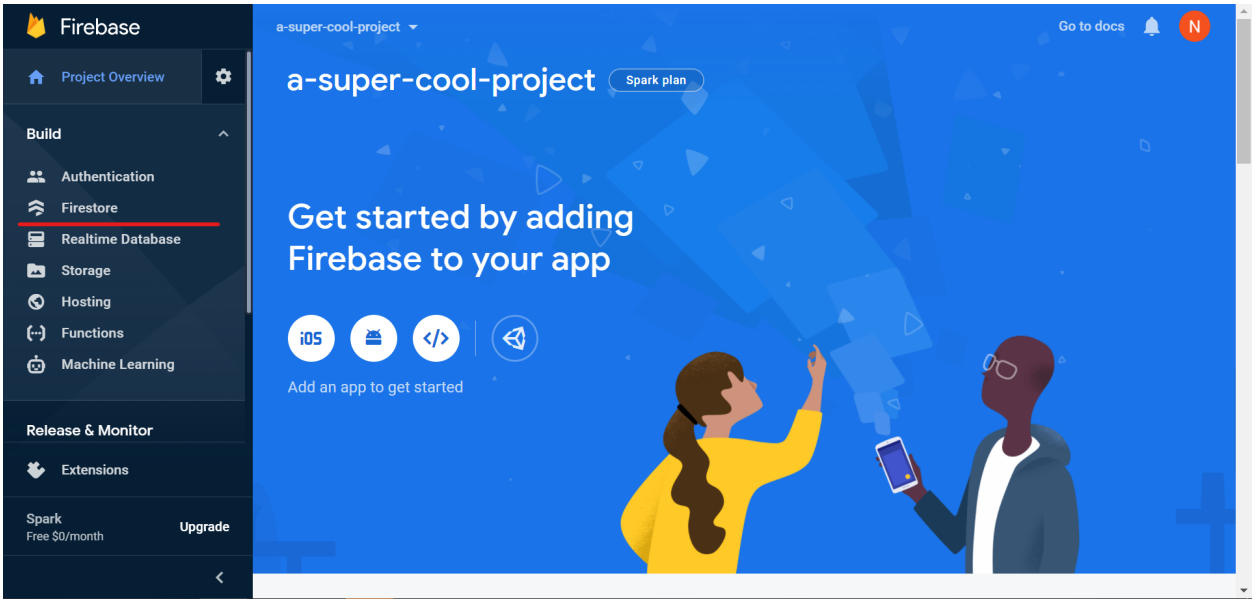


Congratulations! You have a brand new project.
Phew! Not that hard right?
Not a single code is written.

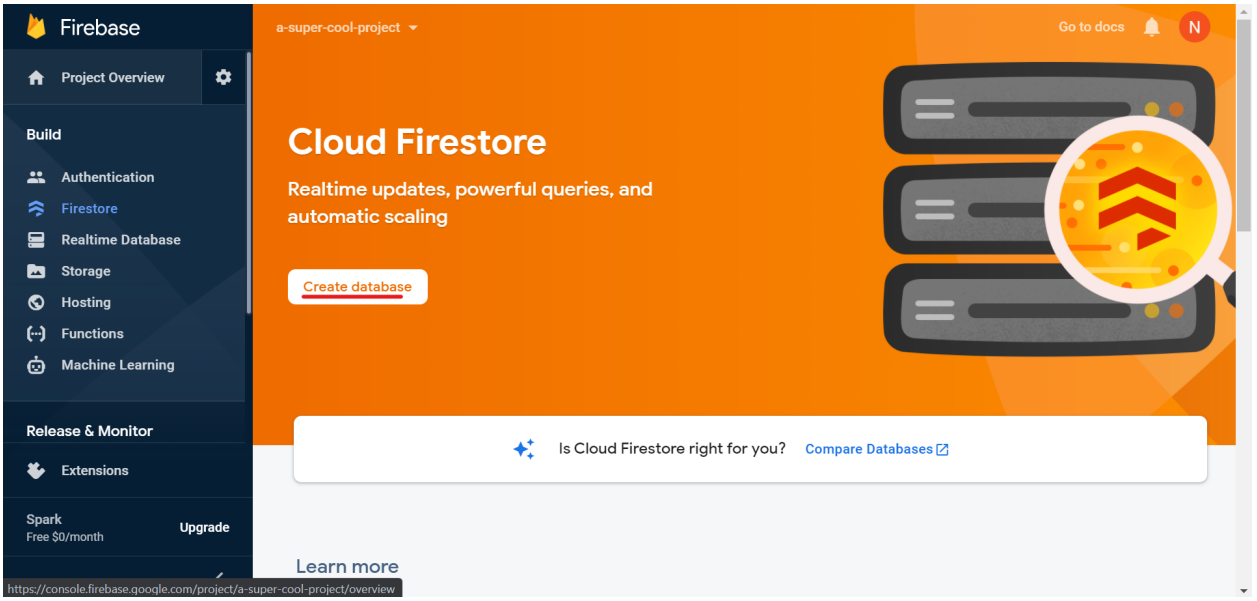


Create a Firestore

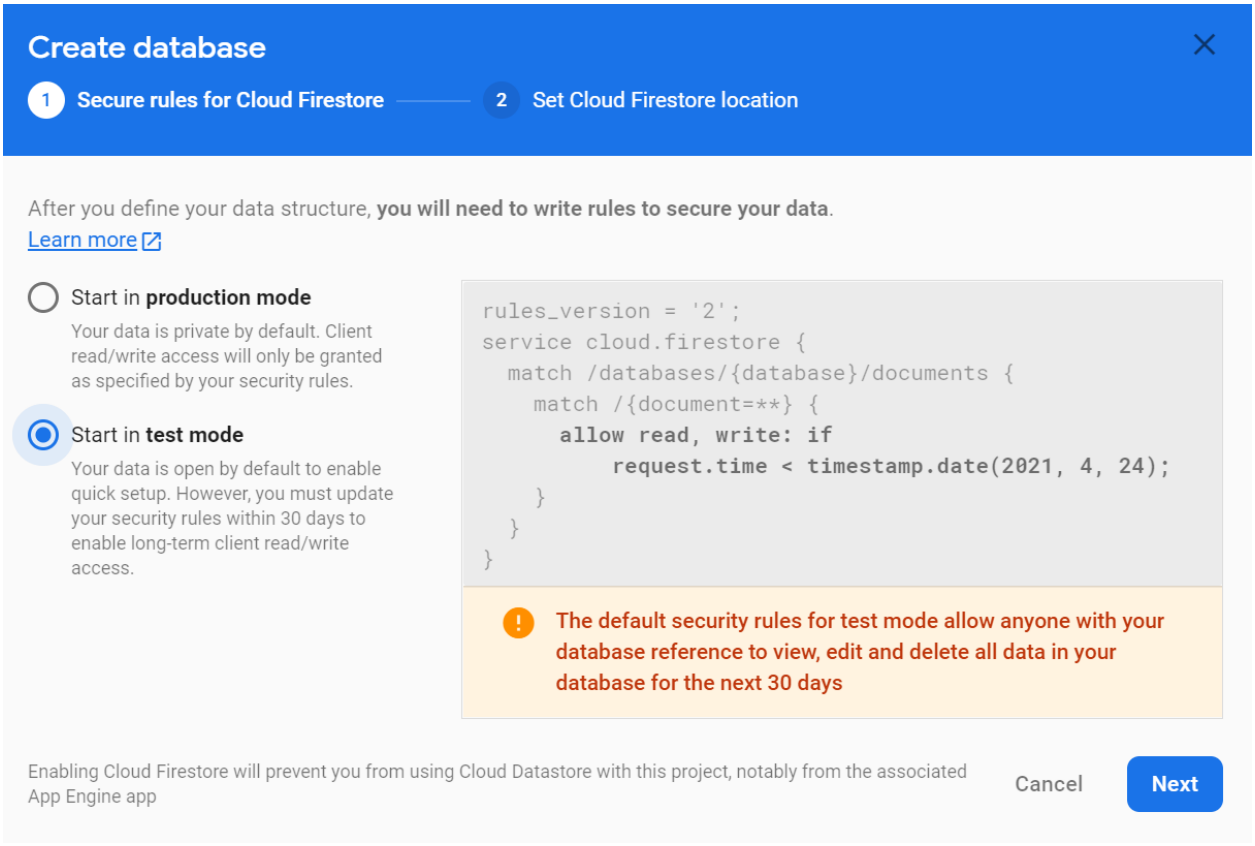
Now, you will create a new database using Firestore.



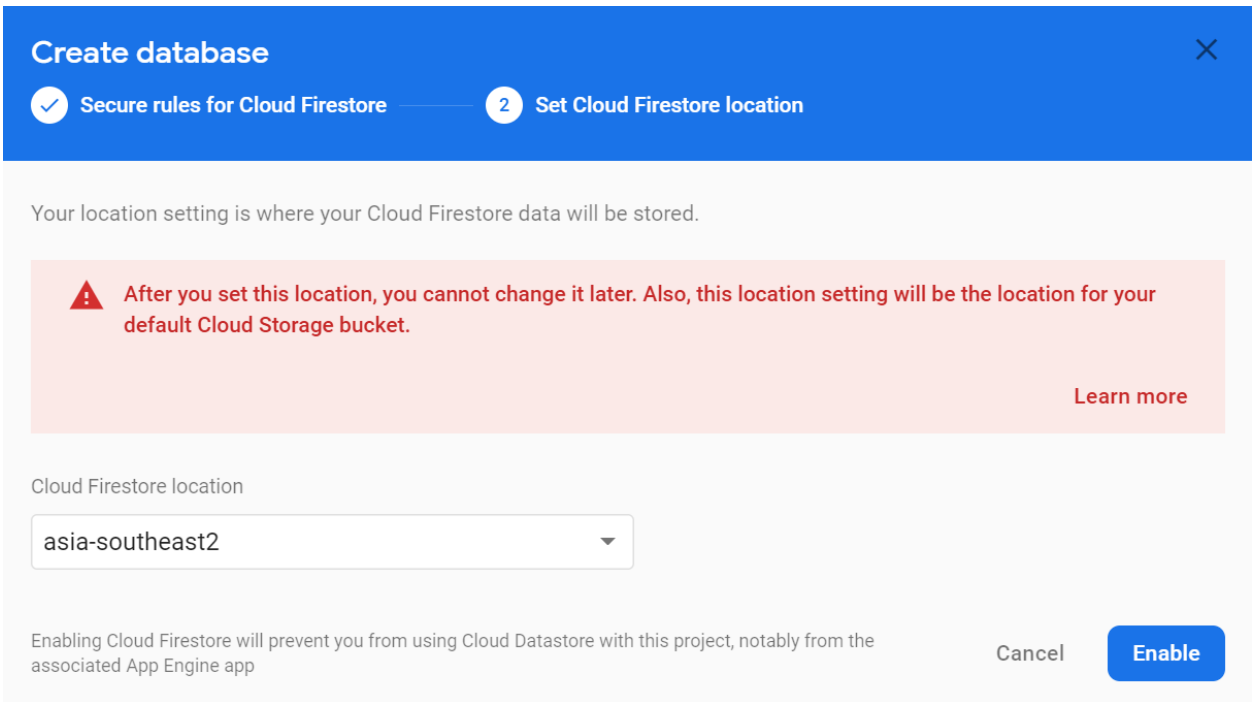
Click create a database



Set the RULES to test mode. Select "Start in **test mode**". This allows ANYONE to view, edit and delete all data in your database for the next 30 days. For more information about RULES, [you can view it on the official document.]



Select a location for your Cloud Firestore. But which one? Well, the best practice is to choose the location near the users. Say, you are here in Thailand, a land of smiles :). The nearest location should be asia-southeast.



Voila! A brand new database.
You are so smart.

In only a few pages, you can create a database and a backend server.
Pat your back.
Take a deep breath.
The following is about setting up a database.

Set up the Firestore Database

Now, you will set up your database. Unlike SQL, Firestore does not store data in a form of table. Firestore is NoSQL. There are two keywords you need to know: Collection and Document.

To simply put, Collection is a *list* and Document is a *dictionary* compared to Python. To illustrate, to create a library database with NoSQL, one possible structure is a Collection that stores Documents of books. A document may have a title, author and ISBN.

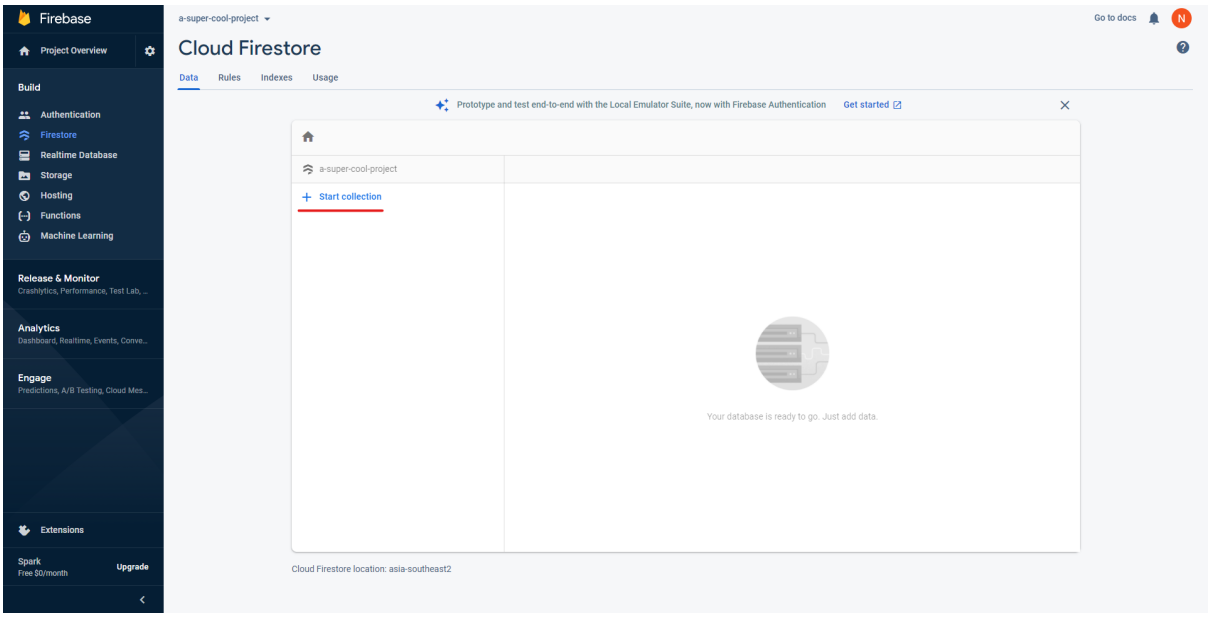
An instance of book may look like this

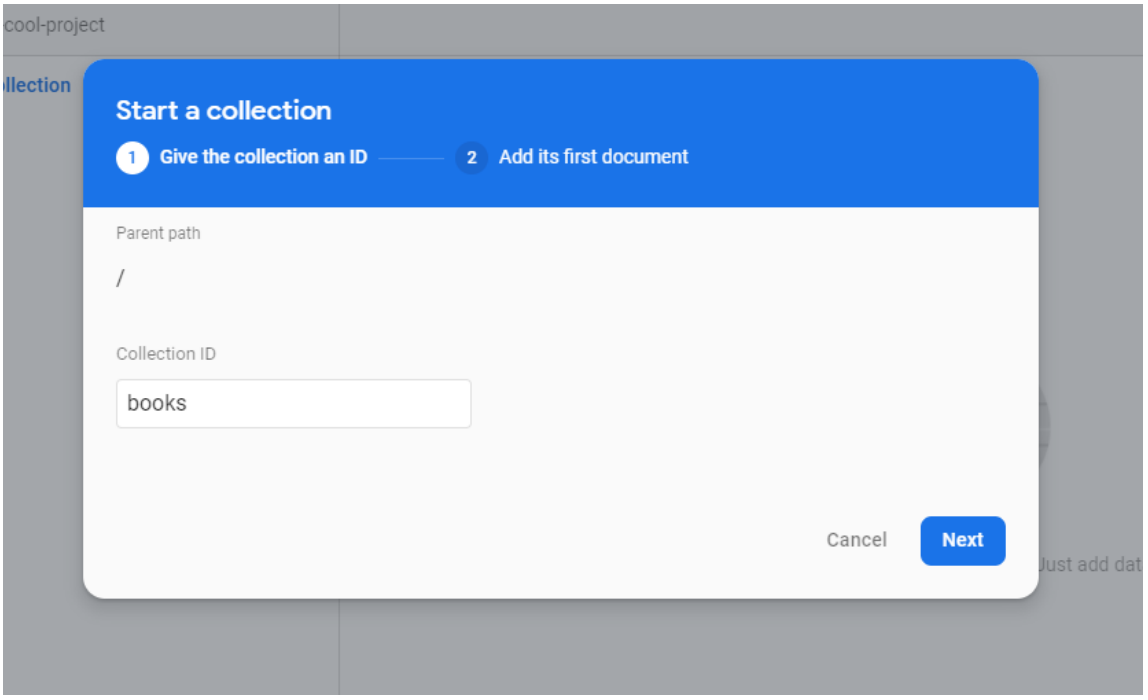
```
book1 = {
  title: "Ulysses",
  author: "James Joyce",
  ISBN: "9781408468418"
}
```

And a Collection of those books will be

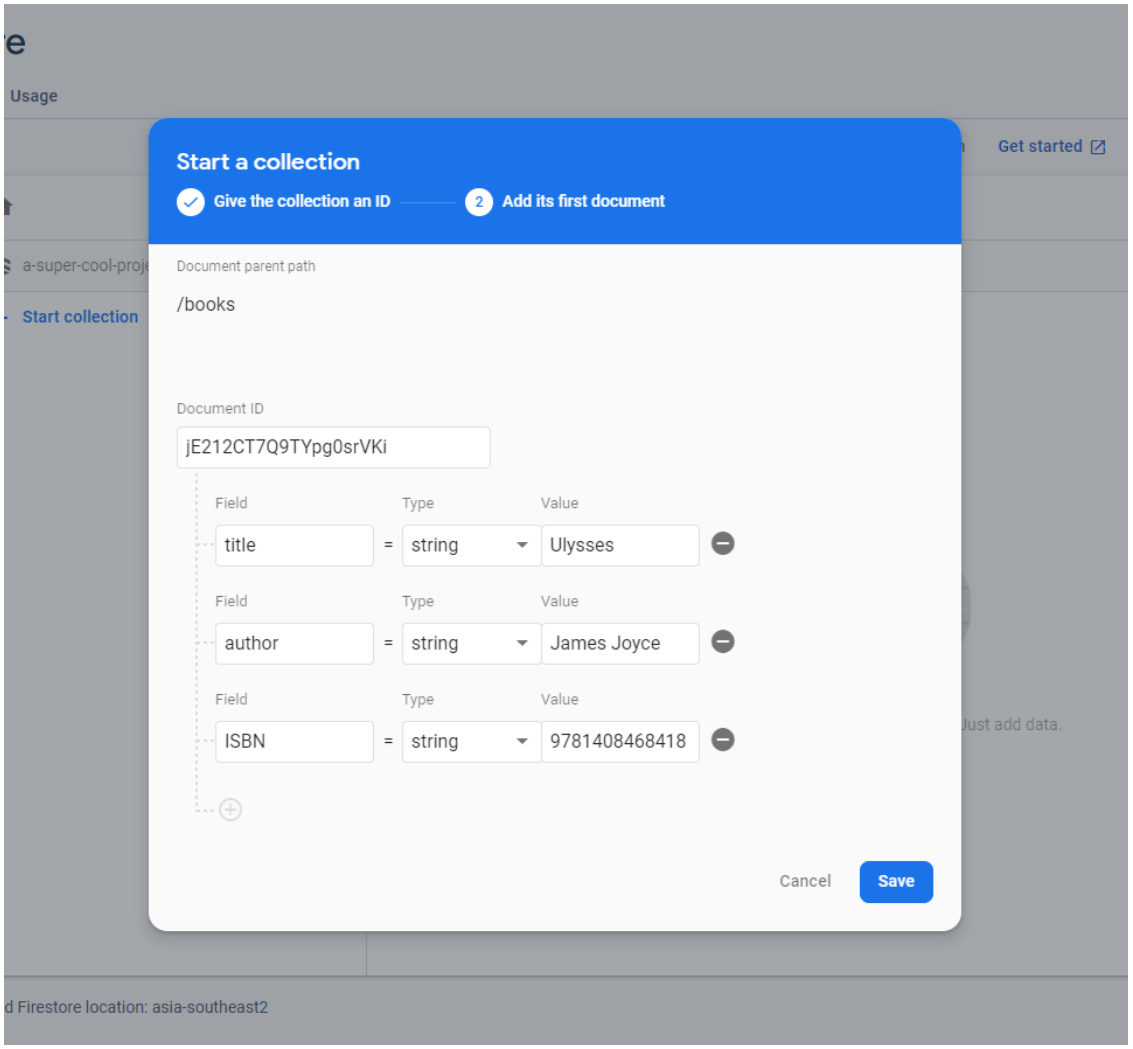
```
books = [book1, book2, book3,...]
```

First, go to the Firestore tab and create a new collection. Call it *books*.





Add the first document into the collection. Add all the fields with their corresponding types (ISBN may be a string since its value does not reflect ordinal quality). You can use Auto ID for each Document ID (see why you should not use incrementally increasing id).

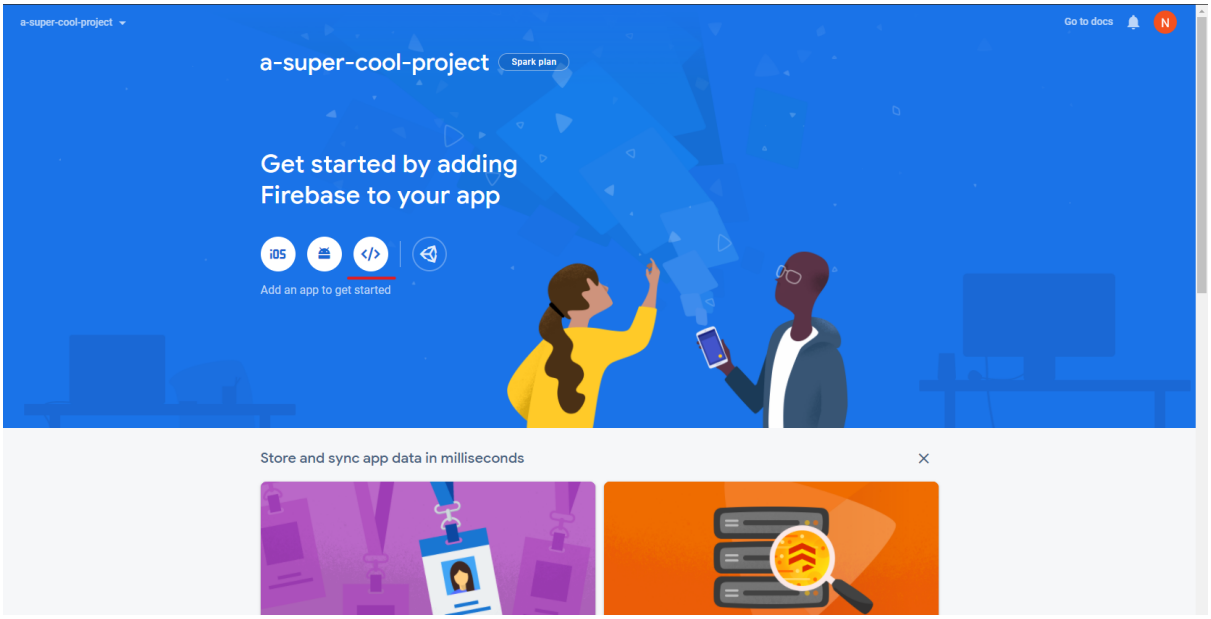


This is a very basic concept of NoSQL. Sorry for those experts in this field.

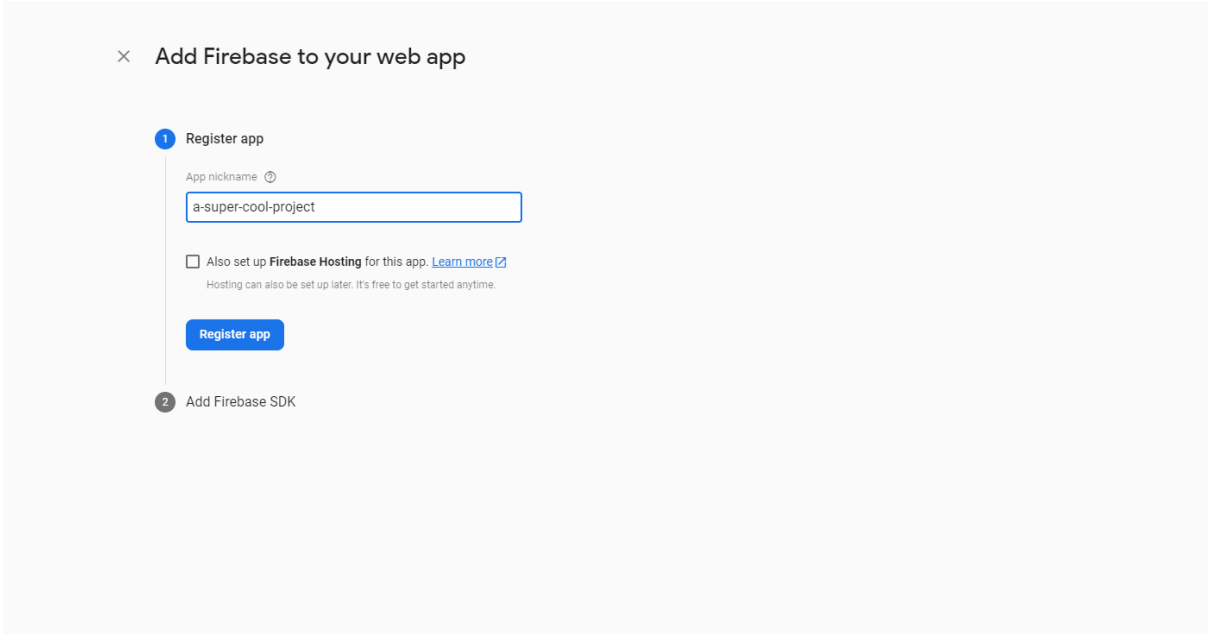
Good job! You have data in a database.
Now what?
Now, you will connect Firebase to your website.
Exciting right?
Come on then!

Create an App for Firebase Project

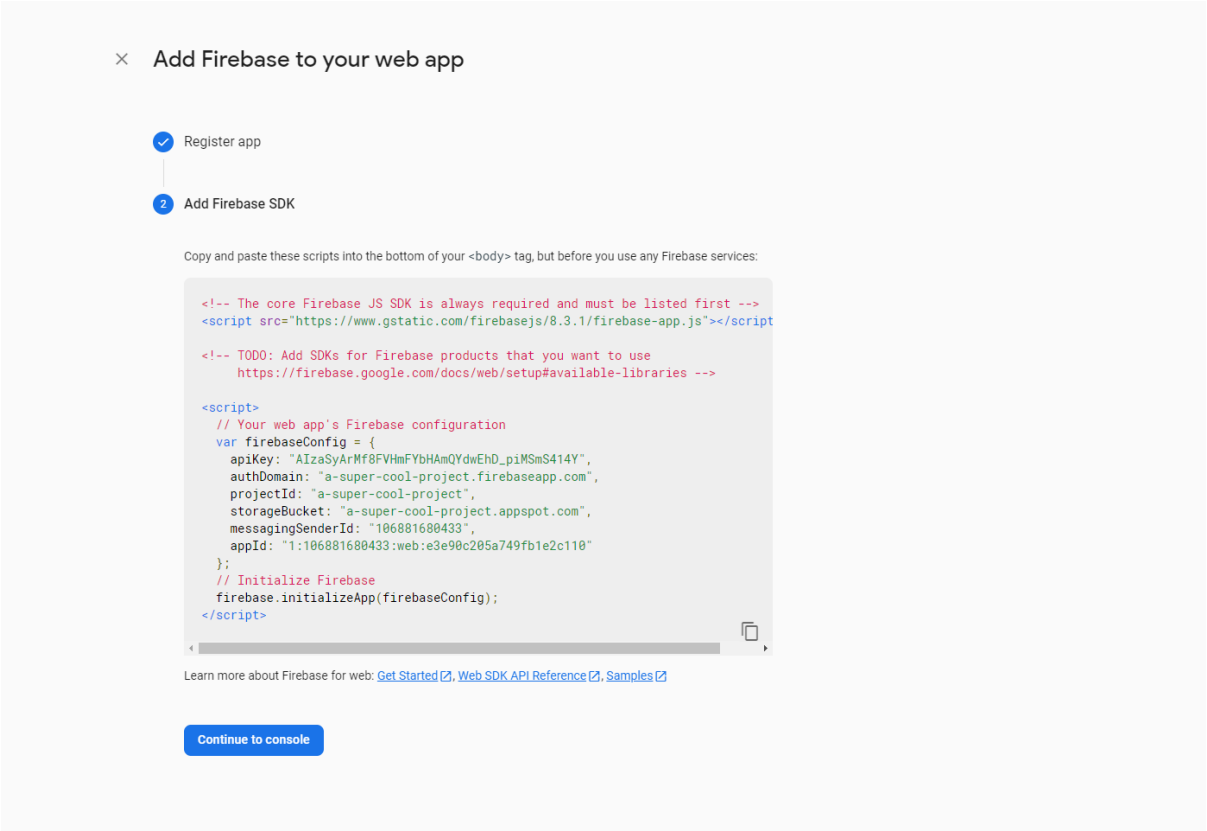
First of all, create an app for your Firebase project. In the overview page, click “</>”



Give it a name. You may use the name of your project.



Nice! You are done. If you have a frontend, you can do as it said to connect your Firebase to your frontend. For now, you can also click “continue to console”

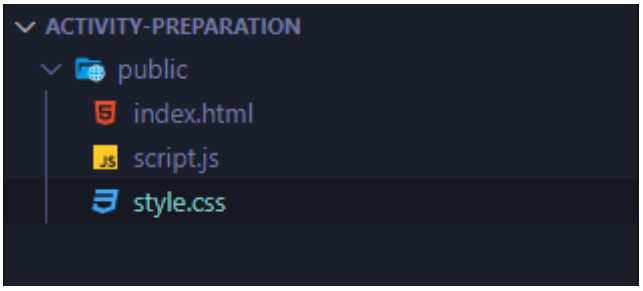


Connect Firebase and Firestore to the Frontend

Okay, it's been a very loooong journey with you. Backend is done. No coding as promised. Now what I did not promise is connecting your frontend with Firebase without coding.

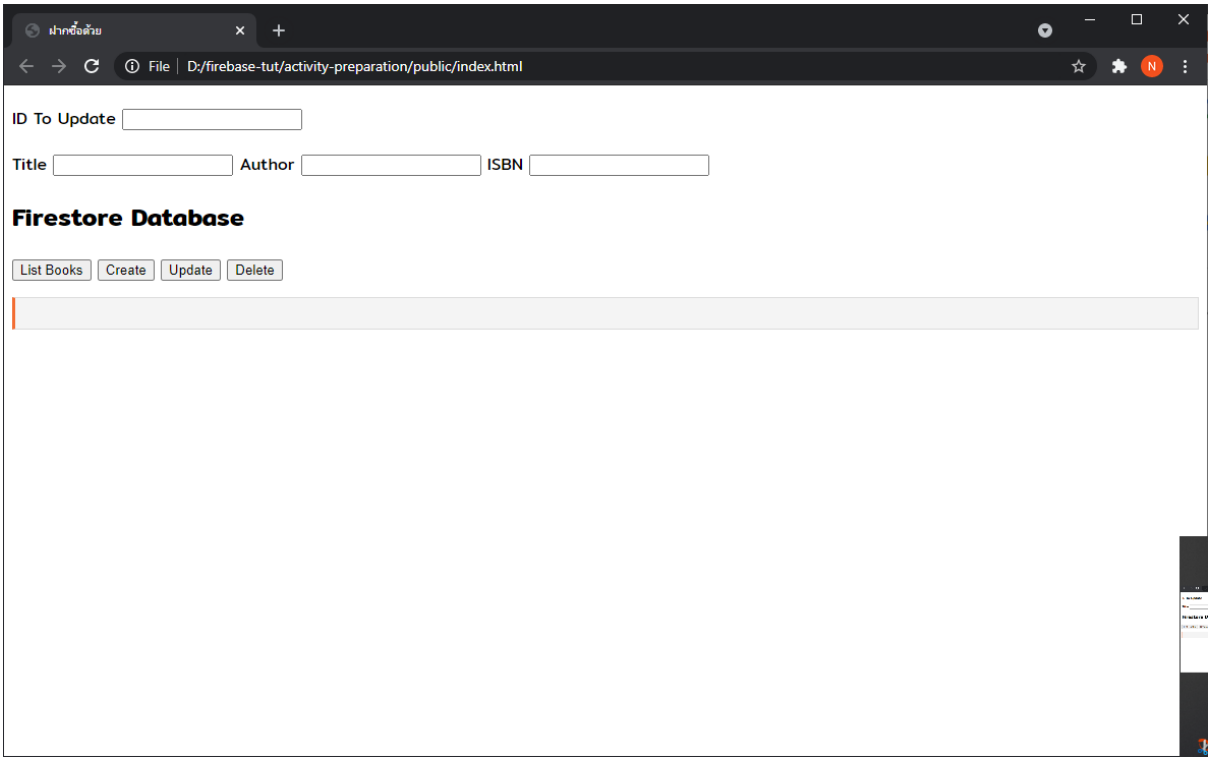
To make it as simple as possible, I will provide a starter project, so you don't have to do anything. You will connect Firebase and Firestore to this starter project (you can also do it yourself here is the video tutorial for you to get started. Ps. I encourage you to get familiar with learning these technical things by yourself as it is a very ESSENTIAL skill in your career.)

Now, this is a starter project. The project consists of HTML, CSS and JavaScript files.

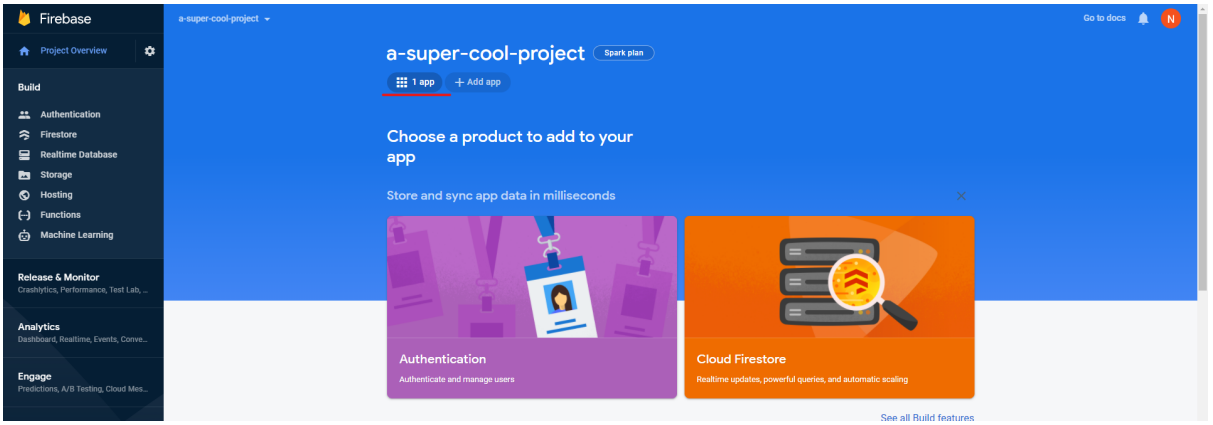


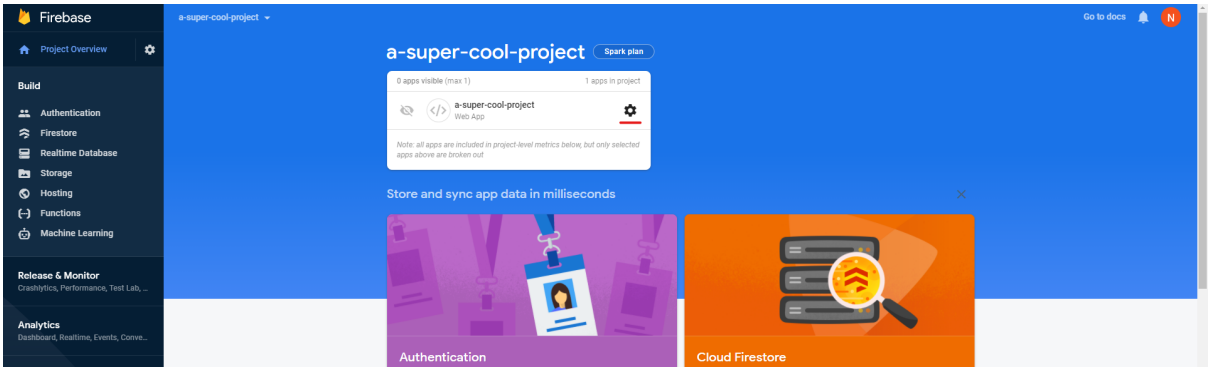
Open index.html with your favorite browser (mine is Chrome). This is a simple frontend that can list, create, update and delete data in your database.

This website is not functioning. That is unsurprising.



First, connect your frontend with your database. In the overview page, go to your app setting.





Scroll down, you will see the instructions about adding the app to your frontend. If you do not understand, that is totally fine.

```
<!-- The core Firebase JS SDK is always required and must be listed
first -->
<script
src="https://www.gstatic.com/firebasejs/8.3.1/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>
  // Your web app's Firebase configuration
  var firebaseConfig = {
    apiKey: "XXXXXXXXXXXXXXXXXXXXXXX",
    authDomain: "{PROJECT_ID}.firebaseapp.com",
    projectId: "{PROJECT_ID}",
    storageBucket: "{PROJECT_ID}.appspot.com",
    messagingSenderId: "XXXXXXXXXX",
    appId: "XXXXXXXXXXXXXXXXXXXXX"
  };
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);
</script>
```

First, you will copy the *SDK* stuff to your index.html <head />. Think of it like installing a core Firebase library to your website. At the src is a link to a JavaScript file (you can open the link to see the file). This is called content-delivery-network (CDN). You can search Google for that, or wait for the Network course in the third year (please do not wait, Google it).

```
<!-- The core Firebase JS SDK is always required and must be listed first
-->
<script
src="https://www.gstatic.com/firebasejs/8.3.1/firebase-app.js"></script>

<!-- TODO: Add SDKs for Firebase products that you want to use
     https://firebase.google.com/docs/web/setup#available-libraries -->
```

Now, you will initialize Firebase. Copy these portion of codes to your project.

```
<script>
  // Your web app's Firebase configuration
  var firebaseConfig = {
    apiKey: "XXXXXXXXXXXXXXXXXXXXXXX",
    authDomain: "{PROJECT_ID}.firebaseapp.com",
    projectId: "{PROJECT_ID}",
    storageBucket: "{PROJECT_ID}.appspot.com",
    messagingSenderId: "XXXXXXXXXX",
```

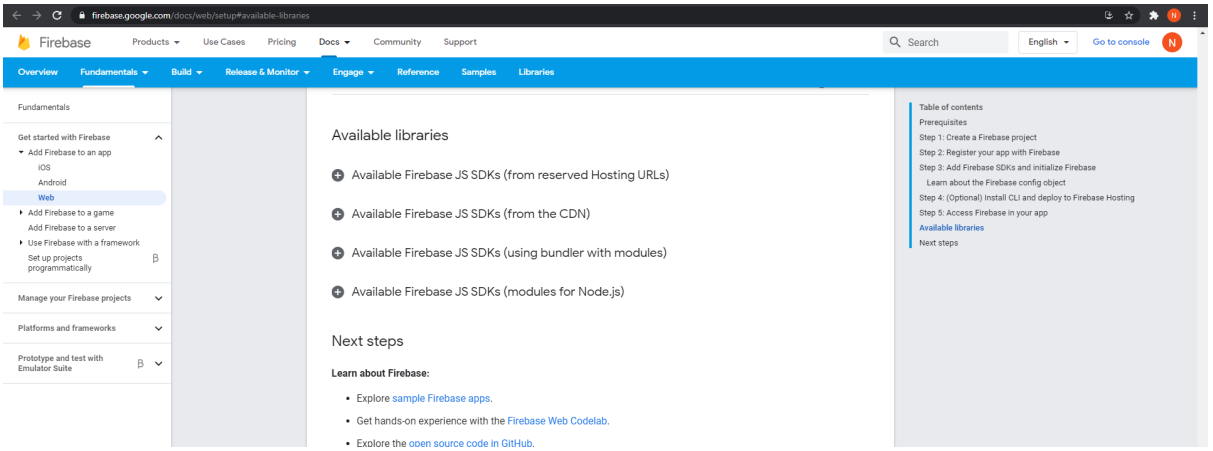
```
    appId: "XXXXXXXXXXXXXXXXXXXXX"
  };
  // Initialize Firebase
  firebase.initializeApp(firebaseConfig);
</script>
```

You should have the following result.
In index.html

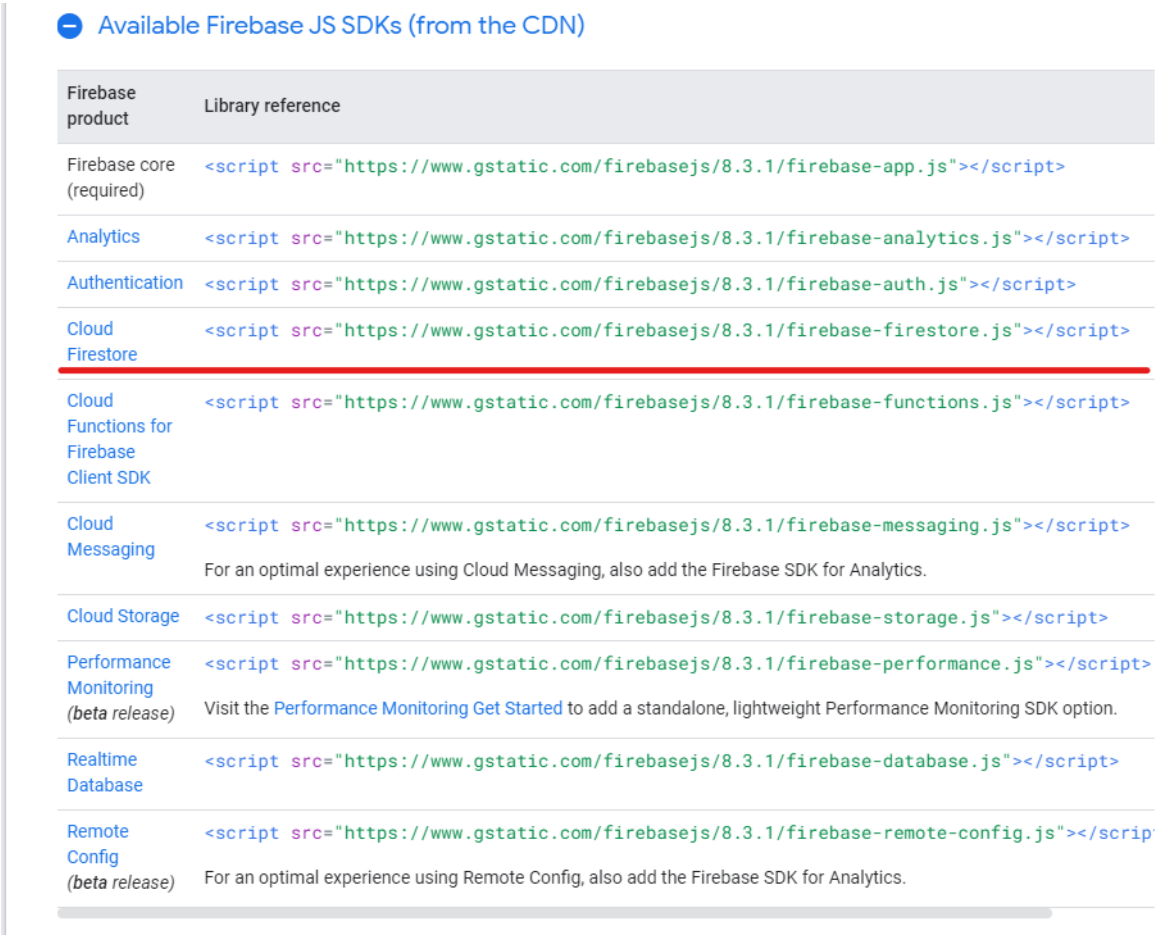
In script.js, make sure *firebaseConfig* is yours not exactly XXXX like this.

Now, there is a TODO telling you to add SDK products. Since you need to connect to Firestore, you should add "Firestore SDK" to the project.

Well, where is it? You can see a link in the comment tag, so click that. You use HTML, so you will need CDN.



There it is! Cloud Firestore🔥. Copy that line and put it below core Firebase SDK.



Available Firebase JS SDKs (from the CDN)

Firebase product	Library reference
Firebase core (required)	`<script src="https://www.gstatic.com/firebasejs/8.3.1/firebase-app.js"></script>`
Analytics	`<script src="https://www.gstatic.com/firebasejs/8.3.1/firebase-analytics.js"></script>`
Authentication	`<script src="https://www.gstatic.com/firebasejs/8.3.1/firebase-auth.js"></script>`
Cloud Firestore	`<script src="https://www.gstatic.com/firebasejs/8.3.1/firebase-firestore.js"></script>`
Cloud Functions for Firebase Client SDK	`<script src="https://www.gstatic.com/firebasejs/8.3.1/firebase-functions.js"></script>`
Cloud Messaging	`<script src="https://www.gstatic.com/firebasejs/8.3.1/firebase-messaging.js"></script>` For an optimal experience using Cloud Messaging, also add the Firebase SDK for Analytics.
Cloud Storage	`<script src="https://www.gstatic.com/firebasejs/8.3.1/firebase-storage.js"></script>`
Performance Monitoring *(beta release)*	`<script src="https://www.gstatic.com/firebasejs/8.3.1/firebase-performance.js"></script>` Visit the Performance Monitoring Get Started to add a standalone, lightweight Performance Monitoring SDK option.
Realtime Database	`<script src="https://www.gstatic.com/firebasejs/8.3.1/firebase-database.js"></script>`
Remote Config *(beta release)*	`<script src="https://www.gstatic.com/firebasejs/8.3.1/firebase-remote-config.js"></scrip` For an optimal experience using Remote Config, also add the Firebase SDK for Analytics.

Great! Now, you can use Firestore.

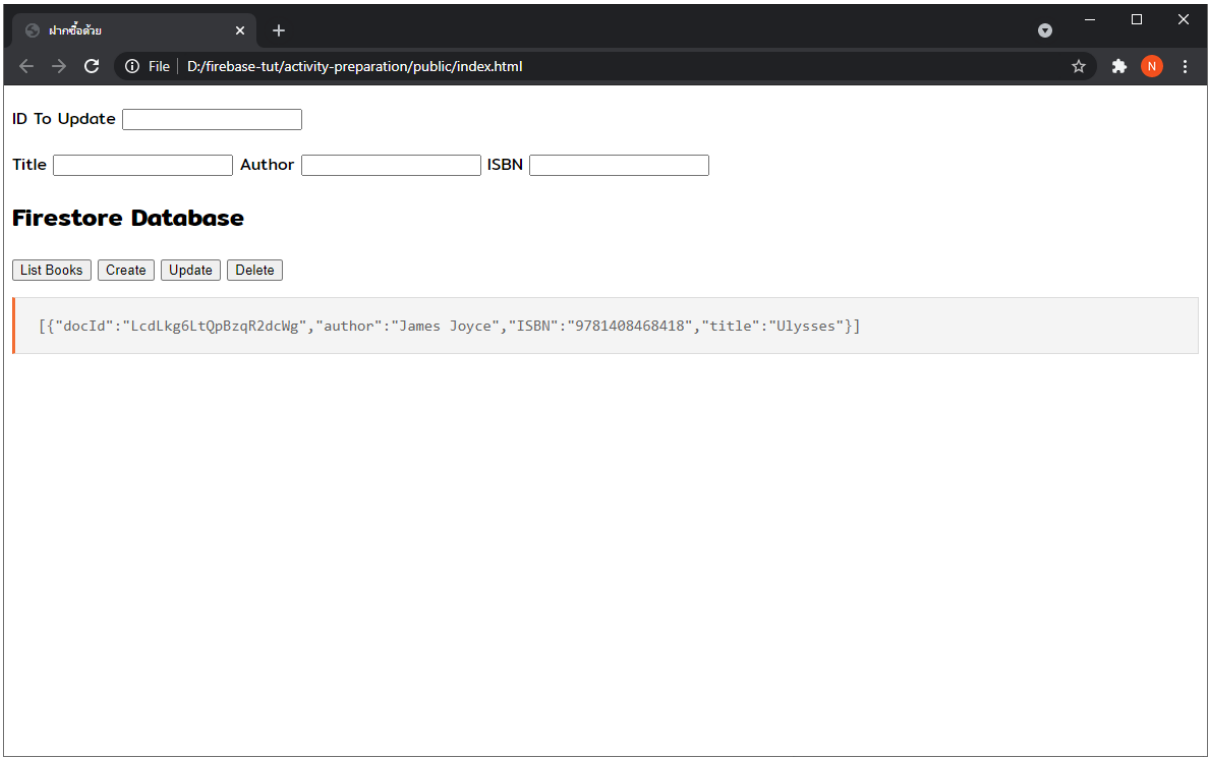
```
<head>
    <title>ฝากซื้อด้วย</title>
    <link rel="stylesheet" href="style.css"></link>
    <!-- The core Firebase JS SDK is always required and must be listed first -->
    <script src="https://www.gstatic.com/firebasejs/8.3.1/firebase-app.js"></script>

    <!-- 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/8.3.1/firebase-firestore.js"></script>
</head>
```

Assign a firestore to a variable.

```
/*
*   ASSIGN FIRESTORE TO A VARIABLE `db`
*   ENTER YOUR CODE HERE
*/
const db = firebase.firestore();
```

High five ✋!
Refresh the index.html opened in the browser.

Click List Books



*Create a new book by entering the fields: **Title, Author and ISBN.** Then click "Create". If you click "List Books" again, your new book should appear.*

*To update or delete, copy **docId** to **ID to Update.***
If you want to delete, click "Delete" and "List Books" again. That book with the corresponding id should be deleted.

If you want to update, enter values in fields you want to update. Click "Update" and "List Books" again. That book with the corresponding id should be updated with new values.

To Do Before Preparation Deadline

"Create a book with

Title: "ComEngEss",
Author: {Your Full Name},
ISBN: {Your Student ID}

Capture the frontend page with the list of this book and put it in the attachment slot"