

UNIVERSITY PARTNER



Distributed and Cloud System Programming (5CS022)

WORKSHOP 10

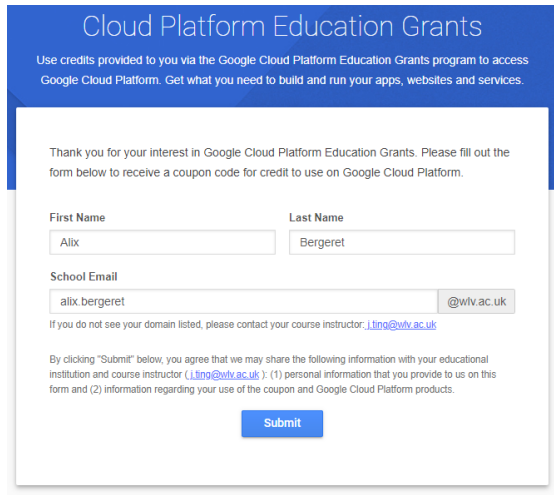
Student Id	: 2227486
Student Name	: Nayan Raj Khanal
Group	: L5CG4
Instructor	: Mr. Prabin Sapkota

Google Firestore

Part 1 – Getting started with Google Firestore

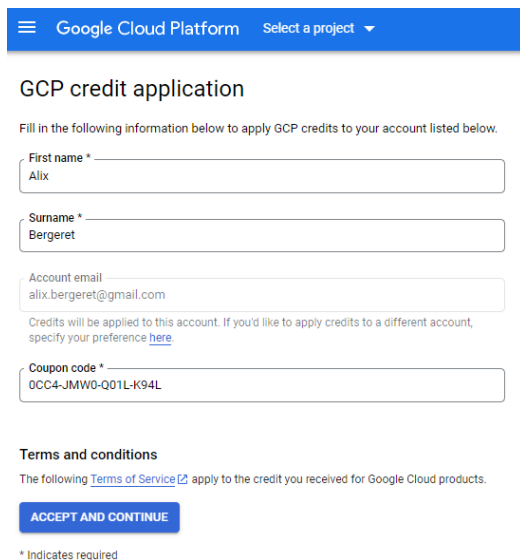
Register for Google Cloud

fill in the form with your details:



The screenshot shows the 'Cloud Platform Education Grants' registration form. At the top, a blue header contains the title and a brief description: 'Use credits provided to you via the Google Cloud Platform Education Grants program to access Google Cloud Platform. Get what you need to build and run your apps, websites and services.' Below this, a white box contains the form. It starts with a thank-you message and a request to fill out the form for a coupon code. The form fields include 'First Name' (Alix), 'Last Name' (Bergeret), and 'School Email' (alix.bergeret@wlv.ac.uk). A dropdown menu for the email domain is set to 'wlv.ac.uk'. A note mentions that if the domain is not listed, users should contact their course instructor at l.ting@wlv.ac.uk. A 'Submit' button is at the bottom of the form. A disclaimer at the bottom of the form states that by clicking 'Submit', users agree to share their information with their educational institution and course instructor for the purpose of providing credits and information about Google Cloud Platform products.

A second email will allow you to verify your email address, and once that's done, you will receive a third email containing your coupon. Click on that link to redeem your coupon, and fill the form:



The screenshot shows the 'GCP credit application' form. At the top, a blue header contains the Google Cloud Platform logo and a 'Select a project' dropdown. Below this, the title 'GCP credit application' is displayed. The form instructions state: 'Fill in the following information below to apply GCP credits to your account listed below.' The form fields include 'First name *' (Alix), 'Surname *' (Bergeret), 'Account email' (alix.bergeret@gmail.com), and 'Coupon code *' (0CC4-JMW0-Q01L-K94L). A note below the email field states: 'Credits will be applied to this account. If you'd like to apply credits to a different account, specify your preference [here](#).' Below the coupon code field, there is a 'Terms and conditions' section with a link to the 'Terms of Service' and a note that the terms apply to the credit received for Google Cloud products. At the bottom, there is a blue 'ACCEPT AND CONTINUE' button. A footnote at the bottom left states: '* Indicates required'.

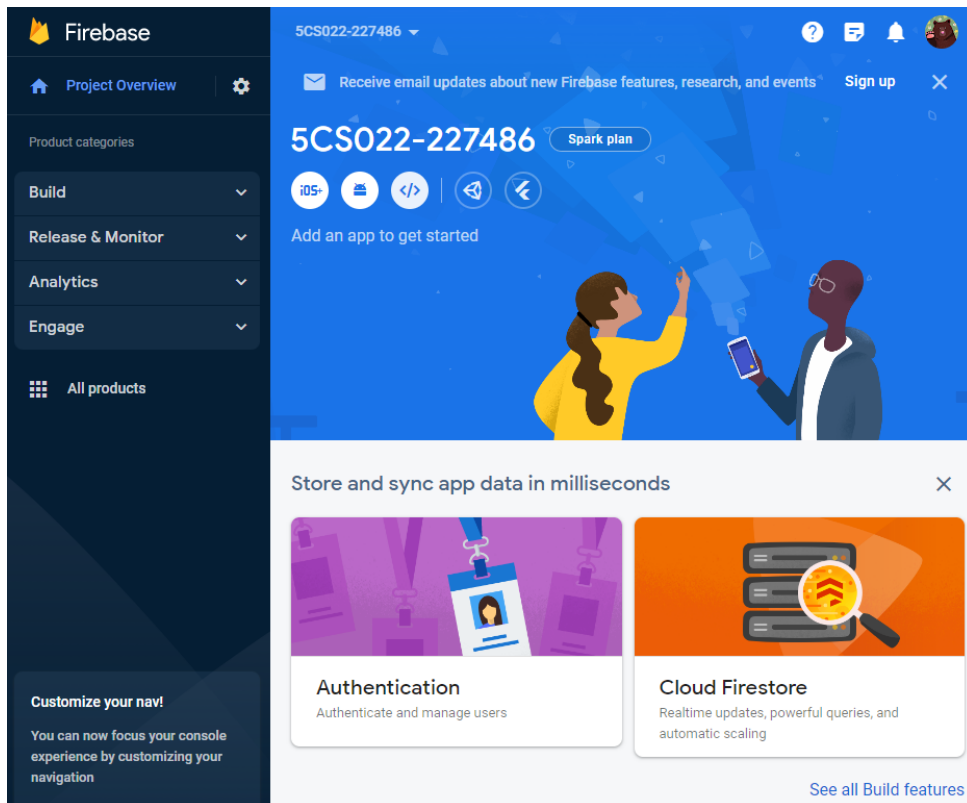
You may have to agree to the Terms of Service (if it's the first time you use Google Cloud), and should then arrive on the Billing Account Overview page:

Access Firebase

Next, access the **Firebase console**: <https://console.firebase.google.com/>

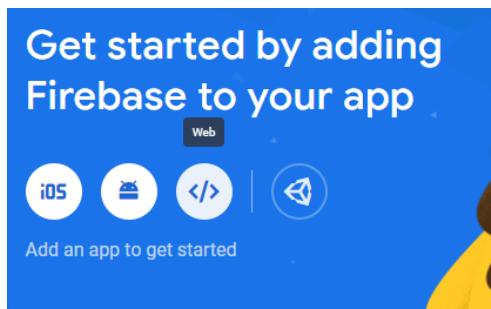
- Press the “Add project” button
- On the following screen, **enter a new project name (maybe “5CS022” + your student number)** and press “Continue”.
- On the “Google Analytics”, **deselect** “Enable Google Analytics for this project”, as we don’t need it for this workshop. Press “Continue”.

- Your Firebase project will be created (it takes a few seconds). When ready, press “Continue” ... you should arrive on the Firebase Console:




Note: you can come back to your Firebase Console any time, by clicking on the following link then selecting your project: <https://console.firebase.google.com/>

The last thing we need to do is “Add Firebase to an app” (see the very large blue section on the homepage of your Firebase console.) **Select the “web” option**, as we will be accessing our Firestore from JavaScript:





Give your app any name you like and press "Register app".



Add Firebase to your web app

1 Register app

App nickname 

☐ Also set up **Firebase Hosting** for this app. [Learn more](#) 

Hosting can also be set up later. There is no cost to get started anytime.

Register app

2 Add Firebase SDK

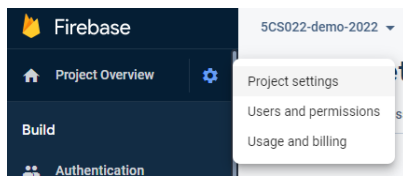
On the next screen, copy and paste the JavaScript code extract into a text file somewhere. It is very important as it contains YOUR connection details. We will use it later when writing our JavaScript. Mine looks like this, BUT USE YOUR OWN:

```
<script type="module">
  // Import the functions you need from the SDKs you need
  import { initializeApp } from "https://www.gstatic.com/firebasejs/9.21.0/firebase-app.js";
  // TODO: Add SDKs for Firebase products that you want to use
  // https://firebase.google.com/docs/web/setup#available-libraries

  // Your web app's Firebase configuration
  const firebaseConfig = {
    apiKey: "AIzaSyD0BUVPpj3tfP5NbY8GhfgFF1scAmddIZE",
    authDomain: "cs022-227486.firebaseio.com",
    projectId: "cs022-227486",
    storageBucket: "cs022-227486.appspot.com",
    messagingSenderId: "38380240244",
    appId: "1:38380240244:web:11e0c5a7a9870aa075c021"
  };

  // Initialize Firebase
  const app = initializeApp(firebaseConfig);
</script>
```

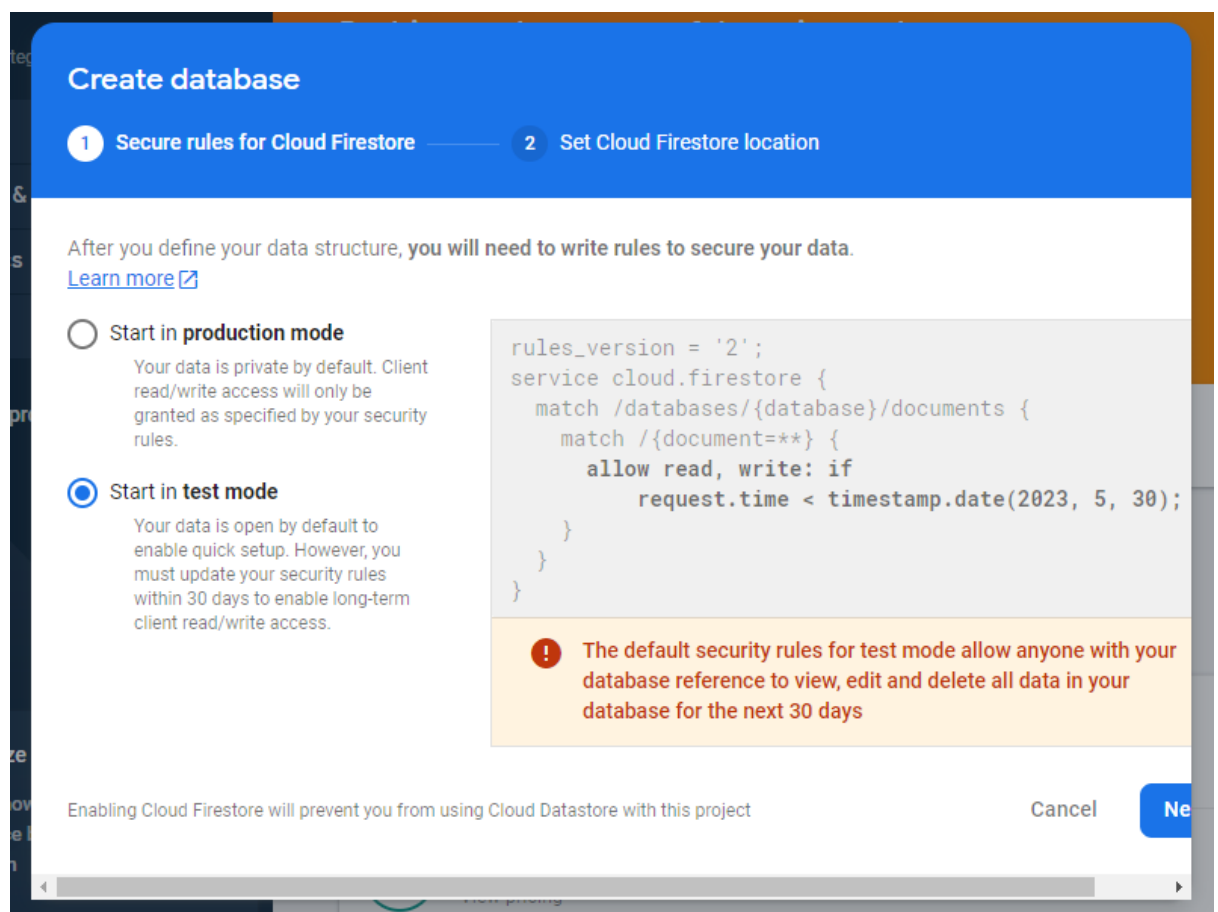
Note: if you lose the credentials above, you can simply retrieve them from your [Firebase Console](#). Click on your project, then “Project Overview”, then “Project settings”:



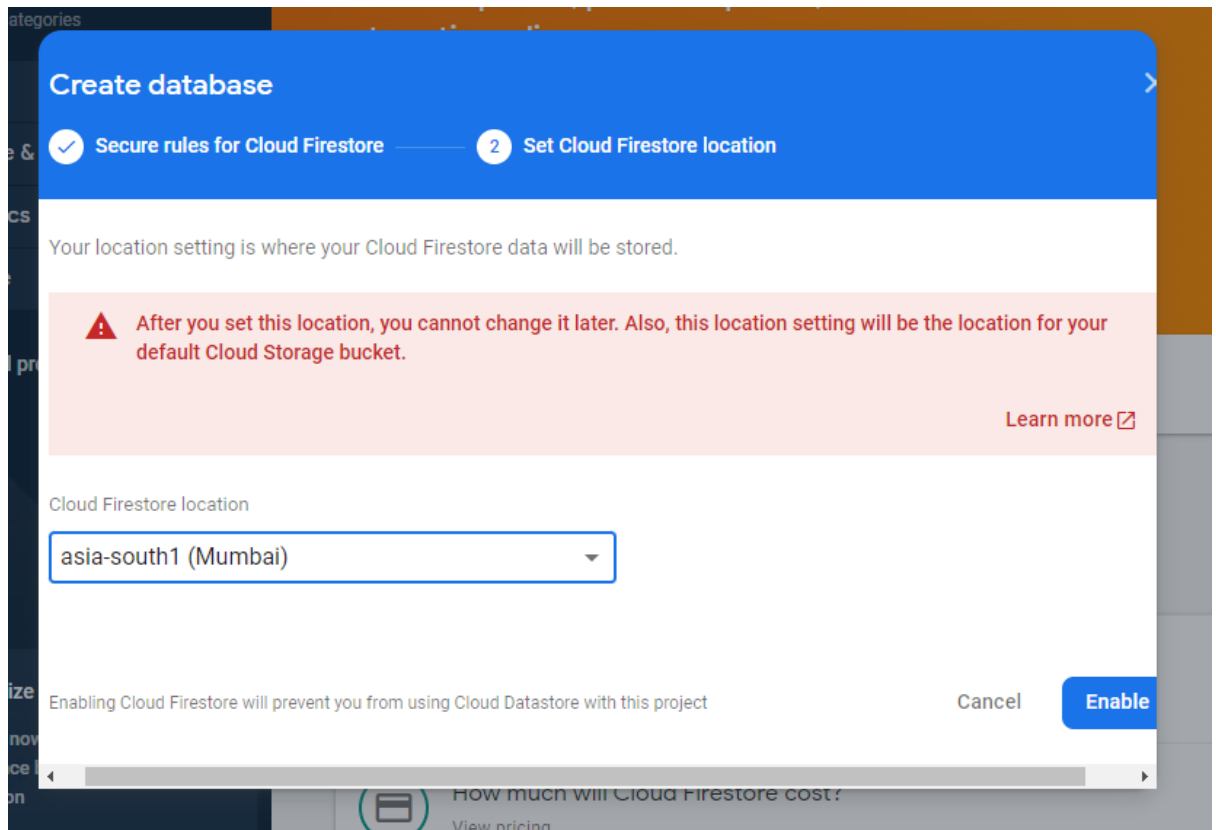
Create a Firestore

Next, we need to **create a Firestore** (NoSQL database):

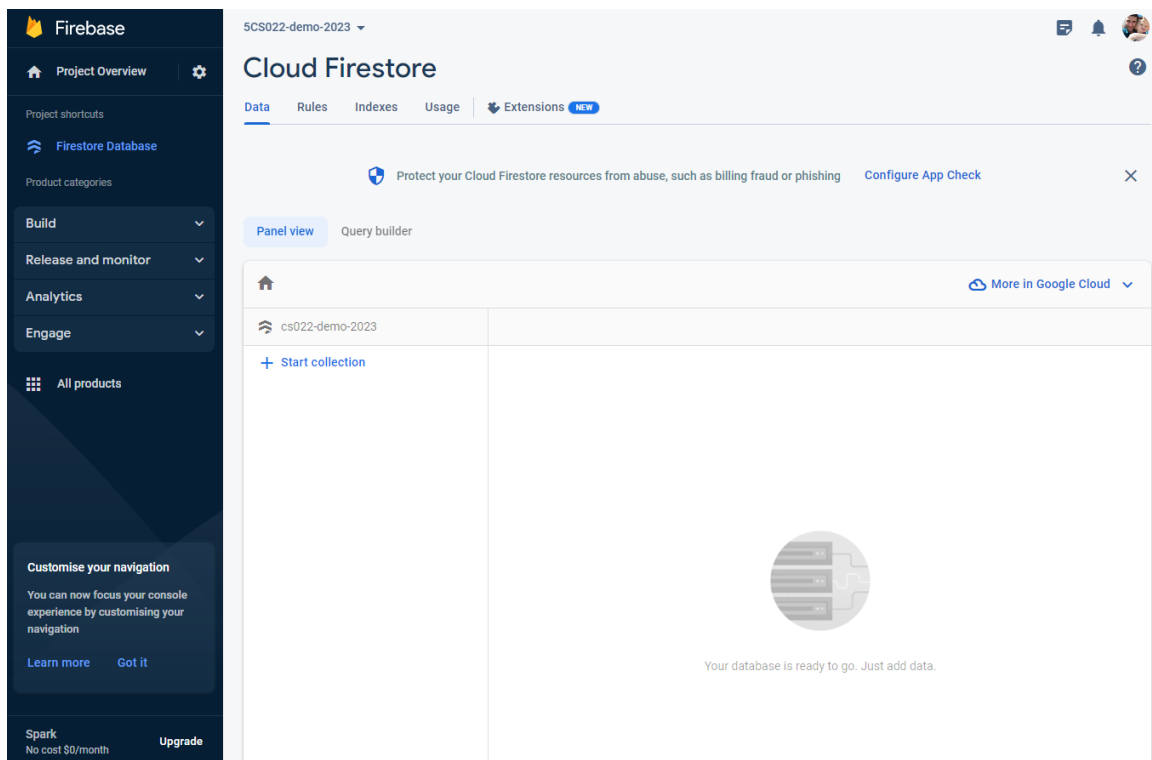
- In your [Firebase console](#), in the menu on the left select **Firestore Database** (under “Build”).
- Click on the “Create database” button in the middle of the screen.
- On the next screen, choose “**Start in test mode**”. This will create an open rule that allows for easy access of your data later on. Obviously, this is not secure!



- On the next screen, set the location of your Firestore to somewhere nearby. This will impact performance and cannot be changed later! Press “Enable”.



You should now be in your [Firestore console](#):



Explore the tabs at the top:

- **“Data”** is where you can add your collections and documents (see next section)
- **“Rules”** allows you to control access to your data. Because we chose “test mode” earlier, it’s open by default.
- **“Indexes”** allows you to create document indexes, just like in a relational database. You will only need this if you use complex queries later on (e.g. if you sort on 2 fields)
- **“Usage”** is self-explanatory!

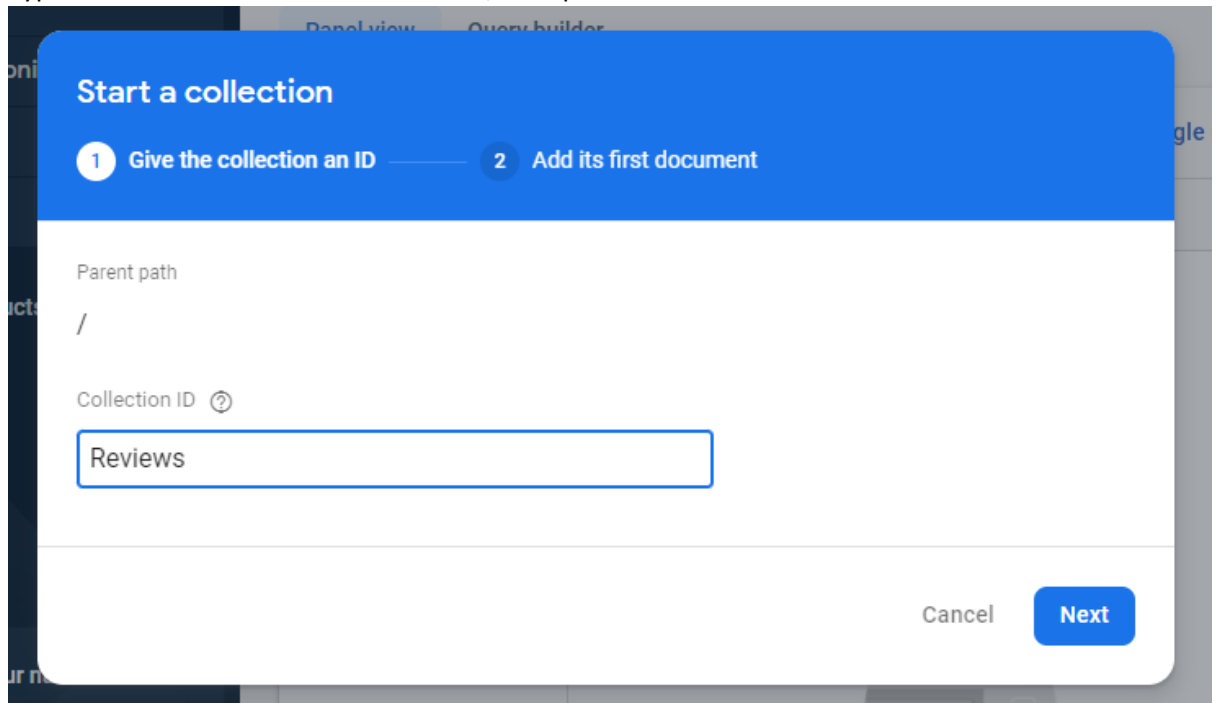
Note that you can access the above via 2 different consoles:

- <https://console.cloud.google.com/firestore/>
- <https://console.firebase.google.com/>

Part 2 – Creating your collections and documents

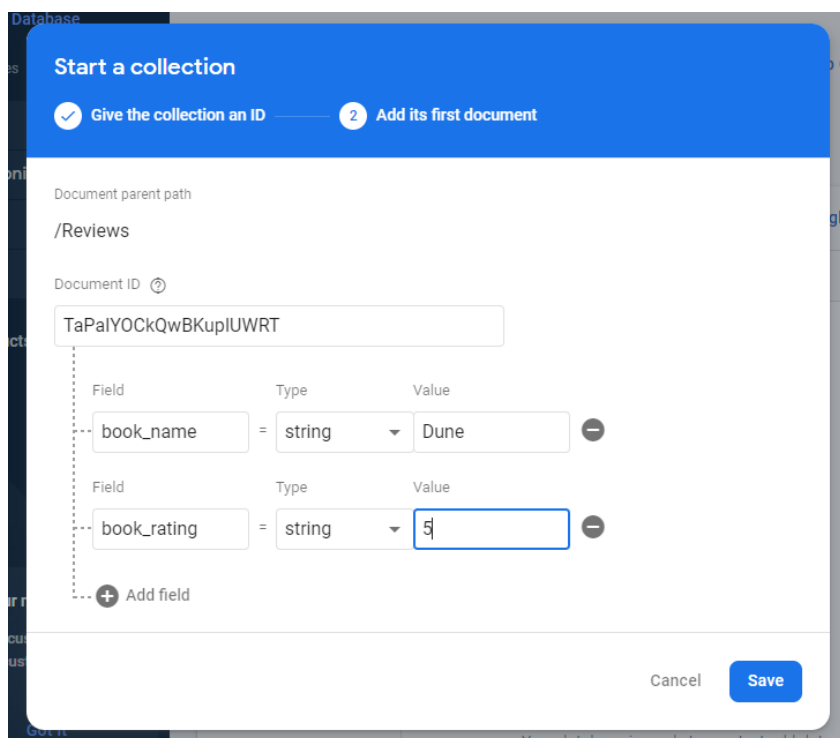
Let's create a "Reviews" **collection** with a couple of **documents**.

1. In your Firestore console, under "Data", press "+ Start Collection"
2. Type "Reviews" in the Collection ID field, then press "Next".



The screenshot shows the 'Start a collection' dialog in the Firestore console. The dialog has a blue header with the title 'Start a collection' and two steps: '1 Give the collection an ID' and '2 Add its first document'. The 'Parent path' is set to '/'. The 'Collection ID' field contains the text 'Reviews'. At the bottom right, there are 'Cancel' and 'Next' buttons.

3. Add a first Document:
 - a. Press "Auto-ID" to populate the Document ID field.
 - b. Create 2 "book_name" and "book_rating" fields (with values), as follow, and press "Save":



The screenshot shows the 'Start a collection' dialog in the Firestore console, now in the '2 Add its first document' step. The 'Document parent path' is set to '/Reviews'. The 'Document ID' field is populated with an auto-generated ID: 'TaPaIYOckQwBKuplUWRT'. Below this, there are two field-value pairs: 'book_name' with type 'string' and value 'Dune', and 'book_rating' with type 'string' and value '5'. At the bottom right, there are 'Cancel' and 'Save' buttons.

4. You should now see your **collection** and single **document**, as follow:

The screenshot shows the Google Cloud Firestore console interface. At the top, the project ID '5CS022-227486' is displayed. The main heading is 'Cloud Firestore'. Below it, there are tabs for 'Data', 'Rules', 'Indexes', 'Usage', and 'Extensions' (marked as 'NEW'). A security warning banner indicates to 'Protect your Cloud Firestore resources from abuse, such as billing fraud or phishing' with a 'Configure App Check' link. Below the banner, there are two view options: 'Panel view' (selected) and 'Query builder'. The main content area shows a breadcrumb path: 'Home > Reviews > TaPaIYOckQwB..'. To the right of the path is a link 'More in Google Cloud'. Below the path, there is a table-like structure with three columns. The first column shows the project ID 'cs022-227486' and a '+ Start collection' button. The second column shows the collection name 'Reviews' and a '+ Add document' button. The third column shows the document ID 'TaPaIYOckQwBKupIUWRT' and a '+ Start collection' button. Below the document ID, there is a '+ Add field' button. The document content is displayed as a JSON object:

```
{  "book_name": "Dune",  "book_rating": "5"}
```

5. Add a few more **documents** in your “Reviews” collection, specifying the same “book_name” and “book_rating” fields every time (but with different values, put your favourite books in there!)

Part 3 – Accessing your data from JavaScript

Let's create a simple web page that will access the data from part 2.

1. Using your favourite coding text editor or IDE, create a **new HTML file**, e.g. **index.html**, with the following starter template:

```
<!doctype html>
<html lang="en">
  <head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <!-- Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet"></head>

    <title>My Firebase app</title>
  </head>
  <body>
    <div class="container">
      <h1 id="mainTitle">My books</h1>
      <table class="table table-striped">
        <tbody id="reviewList">
          </tbody>
        </table>
      </div>

      <!-- jQuery -->
      <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

      <!-- Bootstrap JavaScript -->
      <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"></script>

    </body>
  </html>
```

2. Okay let's connect to our Firestore. Create a **new file** called **"myscripts.js"** with the following code:

```
// Import required Firebase services
import { initializeApp } from
'https://www.gstatic.com/firebasejs/9.18.0/firebase-app.js';
import { Firestore,
  getFirestore,
  onSnapshot,
  query,
  collection,
  orderBy,
  addDoc } from
'https://www.gstatic.com/firebasejs/9.18.0/firebase-firestore.js'

// Your web app's Firebase configuration
const firebaseConfig = {
  apiKey: "use your own",
  authDomain: " use your own ",
  projectId: "use your own",
```

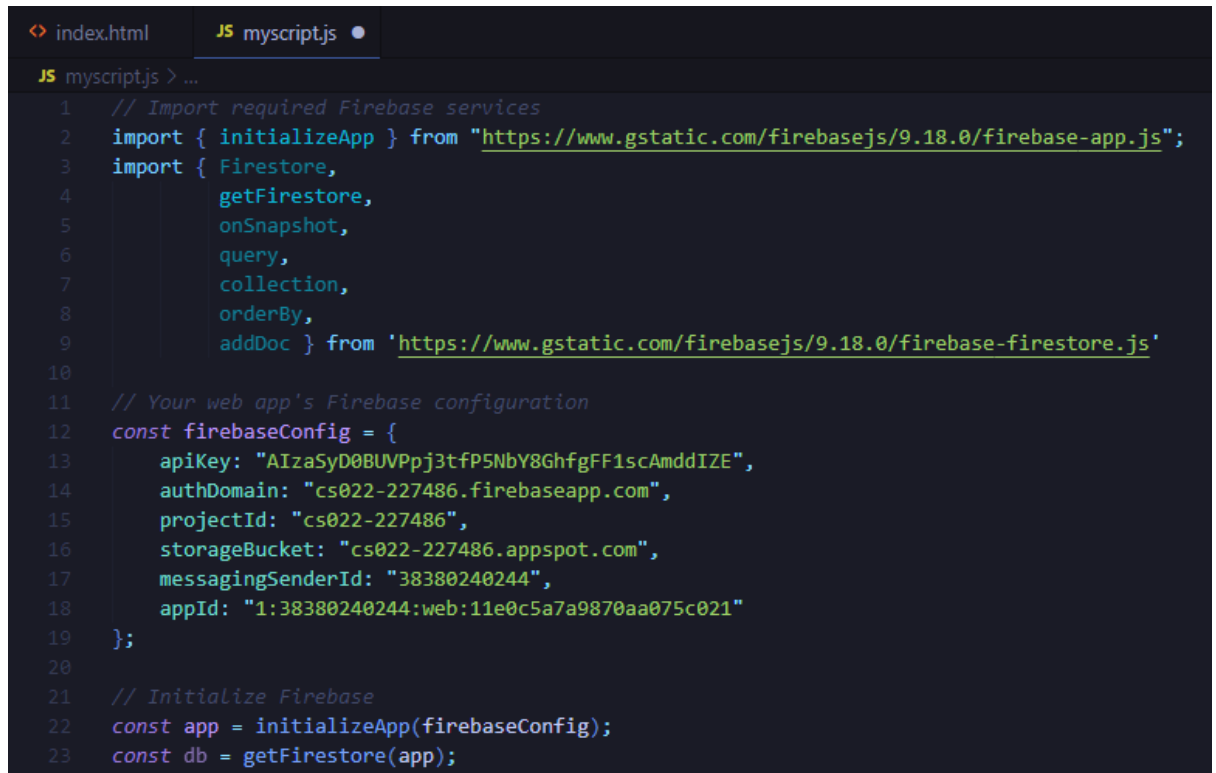
```

    storageBucket: "use your own",
    messagingSenderId: "use your own",
    appId: "use your own"
  };

  // Initialize Firebase
  const app = initializeApp(firebaseConfig);
  const db = getFirestore(app);

```

Note: please use **your own details**, see Part 1 in this document.



```

< index.html JS myscript.js
JS myscript.js > ...
1  // Import required Firebase services
2  import { initializeApp } from "https://www.gstatic.com/firebasejs/9.18.0/firebase-app.js";
3  import { Firestore,
4         getFirestore,
5         onSnapshot,
6         query,
7         collection,
8         orderBy,
9         addDoc } from "https://www.gstatic.com/firebasejs/9.18.0/firebase-firestore.js";
10
11 // Your web app's Firebase configuration
12 const firebaseConfig = {
13   apiKey: "AIzaSyD0BUVPpj3tFP5NbY8GhfgFF1scAmdIzE",
14   authDomain: "cs022-227486.firebaseio.com",
15   projectId: "cs022-227486",
16   storageBucket: "cs022-227486.appspot.com",
17   messagingSenderId: "38380240244",
18   appId: "1:38380240244:web:11e0c5a7a9870aa075c021"
19 };
20
21 // Initialize Firebase
22 const app = initializeApp(firebaseConfig);
23 const db = getFirestore(app);

```

- Then in the **same** "myscript.js" file, **add** the following code:

```

// Get a live data snapshot (i.e. auto-refresh) of our Reviews collection
const q = query(collection(db, "Reviews"), orderBy("book_name"));
const unsubscribe = onSnapshot(q, (snapshot) => {

  // Empty HTML table
  $('#reviewList').empty();

  // Loop through snapshot data and add to HTML table
  var tableRows = '';
  snapshot.forEach((doc) => {
    tableRows += '<tr>';
    tableRows += '<td>' + doc.data().book_name + '</td>';
    tableRows += '<td>' + doc.data().book_rating + '/5</td>';
    tableRows += '</tr>';
  });
  $('#reviewList').append(tableRows);

  // Display review count
  $('#mainTitle').html(snapshot.size + " book reviews in the list");
});

```

```
<> index.html JS myscript.js X
JS myscript.js > ...
1 // Import required Firebase services
2 import { initializeApp } from "https://www.gstatic.com/firebasejs/9.18.0/firebase-app.js";
3 import { Firestore,
4         getFirestore,
5         onSnapshot,
6         query,
7         collection,
8         orderBy,
9         addDoc } from "https://www.gstatic.com/firebasejs/9.18.0/firebase-firestore.js"
10
11 // Your web app's Firebase configuration
12 const firebaseConfig = {
13   apiKey: "AIzaSyD0BUVPpj3tFp5NbY8GhfgFF1scAmdDIZE",
14   authDomain: "cs022-227486.firebaseio.com",
15   projectId: "cs022-227486",
16   storageBucket: "cs022-227486.appspot.com",
17   messagingSenderId: "38380240244",
18   appId: "1:38380240244:web:11e0c5a7a9870aa075c021"
19 };
20
21 // Initialize Firebase
22 const app = initializeApp(firebaseConfig);
23 const db = getFirestore(app);
24
25 // Get a live data snapshot (i.e. auto-refresh) of our Reviews collection
26 const q = query(collection(db, "Reviews"), orderBy("book_name"));
27 const unsubscribe = onSnapshot(q, (snapshot) => {
28
29   // Empty HTML table
30   $('#reviewList').empty();
31
32   // Loop through snapshot data and add to HTML table
33   var tableRows = '';
34   snapshot.forEach((doc) => {
35     tableRows += '<tr>';
36     tableRows += '<td>' + doc.data().book_name + '</td>';
37     tableRows += '<td>' + doc.data().book_rating + '/5</td>';
38     tableRows += '</tr>';
39   });
40   $('#reviewList').append(tableRows);
41
42   // Display review count
43   $('#mainTitle').html(snapshot.size + " book reviews in the list");
44 });
45
```

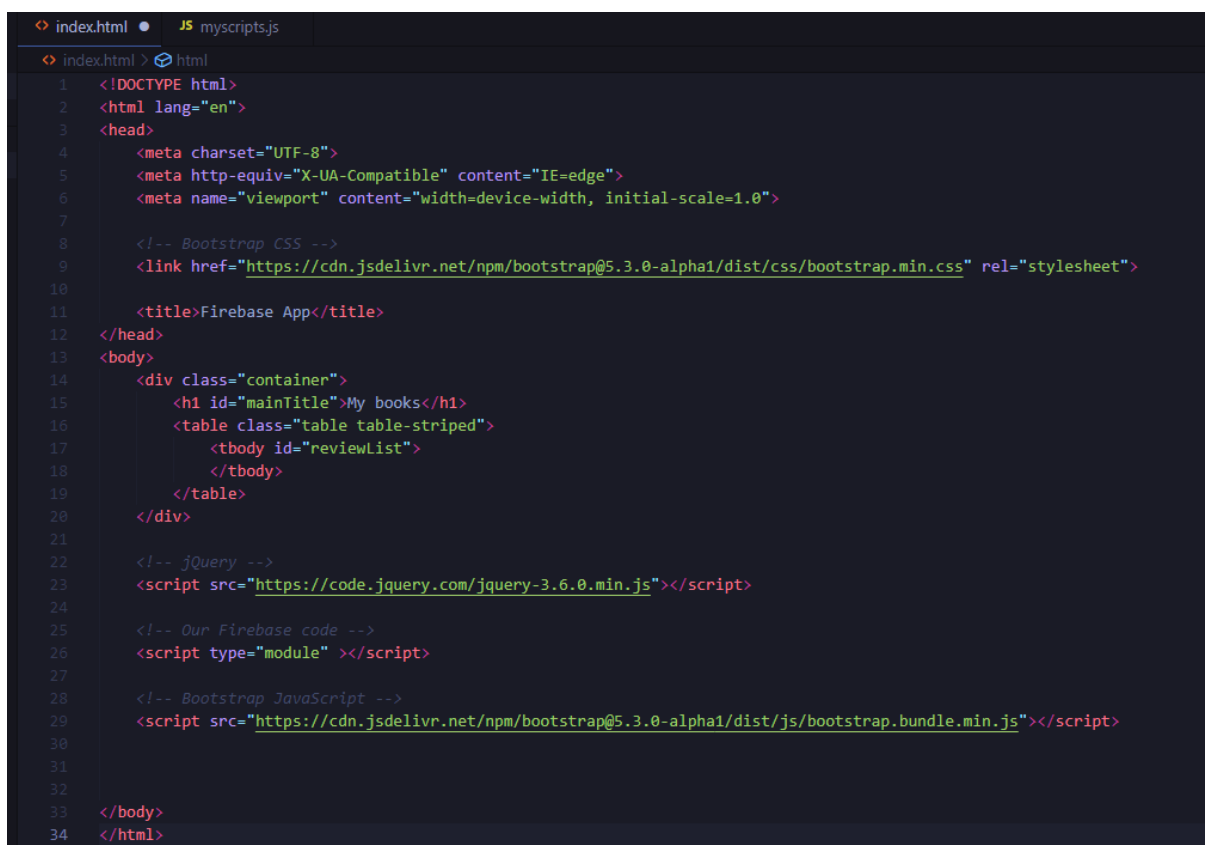
4. Finally, add an **empty HTML table** (to display your results) and include **your JavaScript file** in **your index.html file**, like this:

```
<div class="container">
  <h1 id="mainTitle">My books</h1>
  <table class="table table-striped">
    <tbody id="reviewList">
    </tbody>
  </table>
</div>

<!-- jQuery -->

<!-- Our Firebase code -->
<script type="module" src="myscripts.js"></script>

<!-- Bootstrap JavaScript -->
```



```
index.html JS myscripts.js
index.html > html
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7
8   <!-- Bootstrap CSS -->
9   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
10
11   <title>Firebase App</title>
12 </head>
13 <body>
14   <div class="container">
15     <h1 id="mainTitle">My books</h1>
16     <table class="table table-striped">
17       <tbody id="reviewList">
18       </tbody>
19     </table>
20   </div>
21
22   <!-- jQuery -->
23   <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
24
25   <!-- Our Firebase code -->
26   <script type="module" ></script>
27
28   <!-- Bootstrap JavaScript -->
29   <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"></script>
30
31
32
33 </body>
34 </html>
```

```
index.html X JS myscripts.js
index.html > {} "index.html" > html > body > script
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7
8   <!-- Bootstrap CSS -->
9   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
10
11   <title>Firebase App</title>
12 </head>
13 <body>
14   <div class="container">
15     <h1 id="mainTitle">My books</h1>
16     <table class="table table-striped">
17       <tbody id="reviewList">
18       </tbody>
19     </table>
20   </div>
21
22   <!-- jQuery -->
23   <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
24
25   <!-- Our Firebase code -->
26   <script type="module" >
27     // Import required Firebase services
28 import { initializeApp } from "https://www.gstatic.com/firebasejs/9.18.0/firebase-app.js";
29 import { Firestore,
30   getFirestore,
31   onSnapshot,
32   query,
33   collection,
34   orderBy,
35   addDoc } from 'https://www.gstatic.com/firebasejs/9.18.0/firebase-firestore.js'
36
37 // Your web app's Firebase configuration
38 const firebaseConfig = {
39   apiKey: "AIzaSyD0BUVPpj3tP5NbY8GhfgFF1scAmdIzE",
40   authDomain: "cs022-227486.firebaseio.com",
41   projectId: "cs022-227486",
42   storageBucket: "cs022-227486.appspot.com",
43   messagingSenderId: "38380240244",
44   appId: "1:38380240244:web:11e0c5a7a9870aa075c021"
45 };
46
47 // Initialize Firebase
48 const app = initializeApp(firebaseConfig);
49 const db = getFirestore(app);
50
```

5. All done. Browse to your file, and you should see a nice HTML table displaying your data from Part 2.

3 book reviews in the list

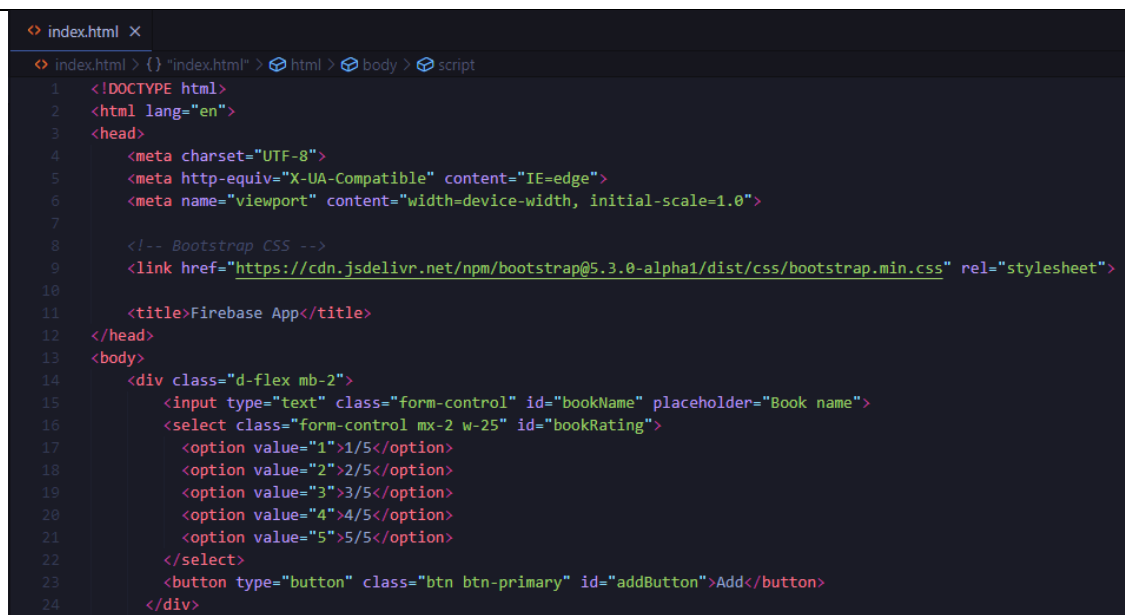
Dune	5/5
OnePiece	5/5
TheGodfather	5/5

Part 4 – Adding more data from JavaScript

Finally, we would like to be able to **add new reviews directly from our web page**.

1. **Add** the following HTML in **index.html**, just **above your HTML table**:

```
<div class="d-flex mb-2">
  <input type="text" class="form-control" id="bookName" placeholder="Book name">
  <select class="form-control mx-2 w-25" id="bookRating">
    <option value="1">1/5</option>
    <option value="2">2/5</option>
    <option value="3">3/5</option>
    <option value="4">4/5</option>
    <option value="5">5/5</option>
  </select>
  <button type="button" class="btn btn-primary" id="addButton">Add</button>
</div>
```



```
index.html X
index.html > {} "index.html" > html > body > script
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta http-equiv="X-UA-Compatible" content="IE=edge">
6   <meta name="viewport" content="width=device-width, initial-scale=1.0">
7
8   <!-- Bootstrap CSS -->
9   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/css/bootstrap.min.css" rel="stylesheet">
10
11   <title>Firebase App</title>
12 </head>
13 <body>
14   <div class="d-flex mb-2">
15     <input type="text" class="form-control" id="bookName" placeholder="Book name">
16     <select class="form-control mx-2 w-25" id="bookRating">
17       <option value="1">1/5</option>
18       <option value="2">2/5</option>
19       <option value="3">3/5</option>
20       <option value="4">4/5</option>
21       <option value="5">5/5</option>
22     </select>
23     <button type="button" class="btn btn-primary" id="addButton">Add</button>
24   </div>
```


2. Add the following code in your existing **myscripts.js** file:

```
// Add button pressed
$("#addButton").click(function() {

    // Add review to Firestore collection
    const docRef = addDoc(collection(db, "Reviews"), {
        book_name: $("#bookName").val(),
        book_rating: parseInt($("#bookRating").val())
    });

    // Reset form
    $("#bookName").val('');
    $("#bookRating").val('1');
});
```



```
index.html x
index.html > {} "index.html" > html > body > script
85 // Add button pressed
86 $("#addButton").click(function() {
87
88     // Add review to Firestore collection
89     const docRef = addDoc(collection(db, "Reviews"), {
90         book_name: $("#bookName").val(),
91         book_rating: parseInt($("#bookRating").val())
92     });
93
94     // Reset form
95     $("#bookName").val('');
96     $("#bookRating").val('1');
97 });
98
99
100 </script>
101
102 <!-- Bootstrap JavaScript -->
103 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha1/dist/js/bootstrap.bundle.min.js"></script>
104
105
106
107 </body>
108 </html>
```

3. All done! You should now be able to add reviews to your list (which will be refreshed automatically).



Book name 1/5

5 book reviews in the list

Dune	5/5
Nayan	5/5
OnePiece	5/5
TheFinal	3/5
TheGodfather	5/5