# Lab: Remote Databases

Problems for in-class lab for the "[JavaScript Applications" course @ SoftUni](https://softuni.bg/courses/js-applications)

**1. Firebase: All Books**

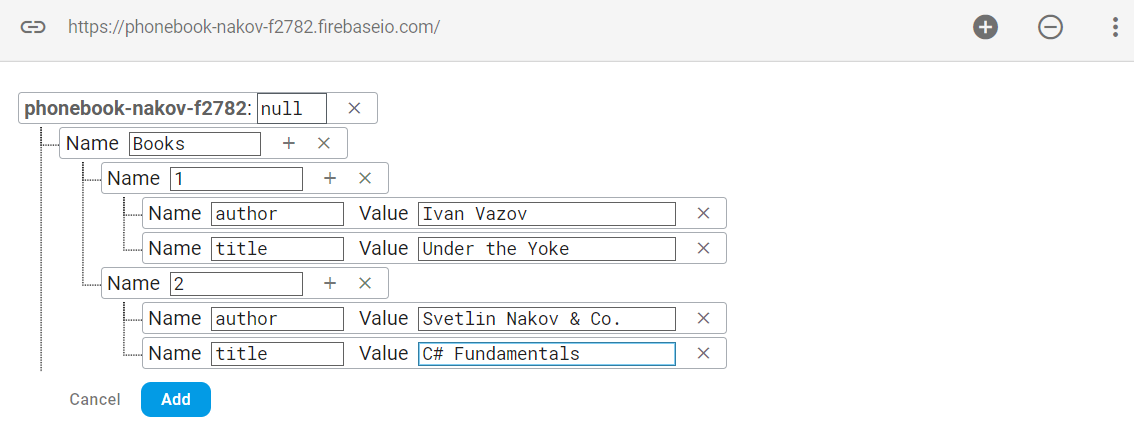
Firebase is a **mobile** and **web application** development **platform**.

Create a "**TestApp**" and then create the **following** structure:

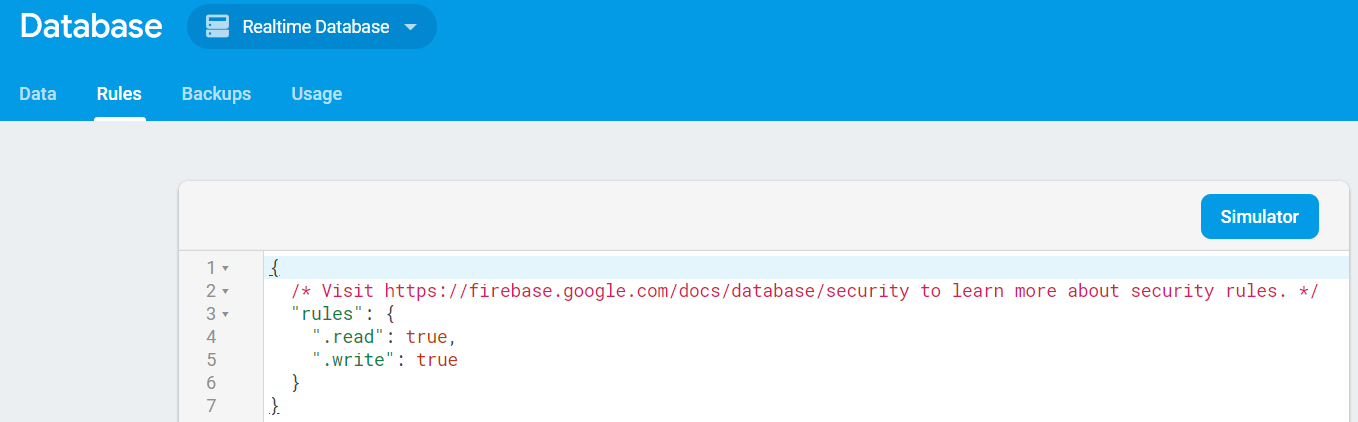


First task is to "**GET**" all books. To consume the request with **POSTMAN** your **url** should be the **following**: https://{databaseId}.firebaseio.com/.json.

**DatabaseId** is unique for every application. You can **find** yours from here:



We **should** also do one more configuration. Go to Database/Rules and set **.read** & **.write** actions to "**true**". This will allow us to **send** request with **POSTMAN**. Beware that now everyone can **manipulate** our database and even **delete** it. (this is for **testing** purposes only).

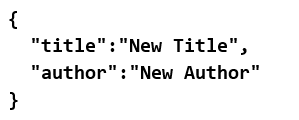


**2. Firebase: Get Book**

"**GET**" the Book with **id**: 1. Don’t forget the **.json** extension at the end (otherwise you will receive the whole **html**).

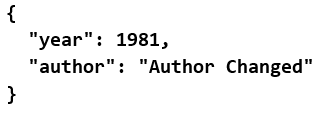
**3. Firebase: Create Book**

To **create** a book, we will have to send a "**POST**" request and the JSON body should be in the **following** format:



**4. Firebase: Patch Book**

The HTTP command "**PATCH**" **modifies** an existing HTTP **resource** (it can also create the resource if it does **not** exist). The JSON body should be in the **following** format:



**5. Firebase: Change Book Author**

The next task is to execute a "**PUT**" command (the difference is that with "**PUT"** we can update a resource **partially**). In our case we have to **change** the author’s name to "**New author was assigned**".

**REQUEST**: https://{databaseId}.firebaseio.com/Books/7/author/.json

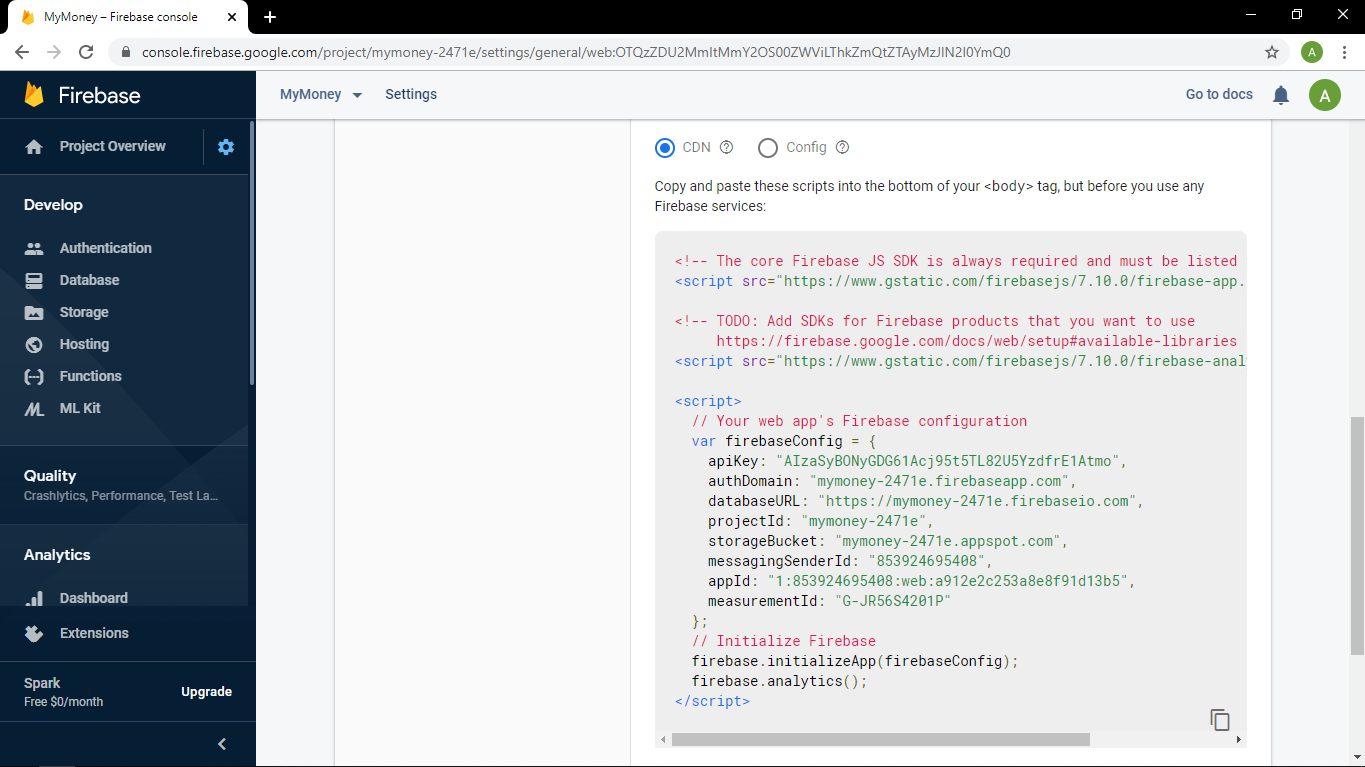
The JSON body should be in the **following** format:

"**New author was assigned**".

**6. Firebase: Initialize Project**

**Initializing** project is done with **config object** and then passed **to initializeApp** method.

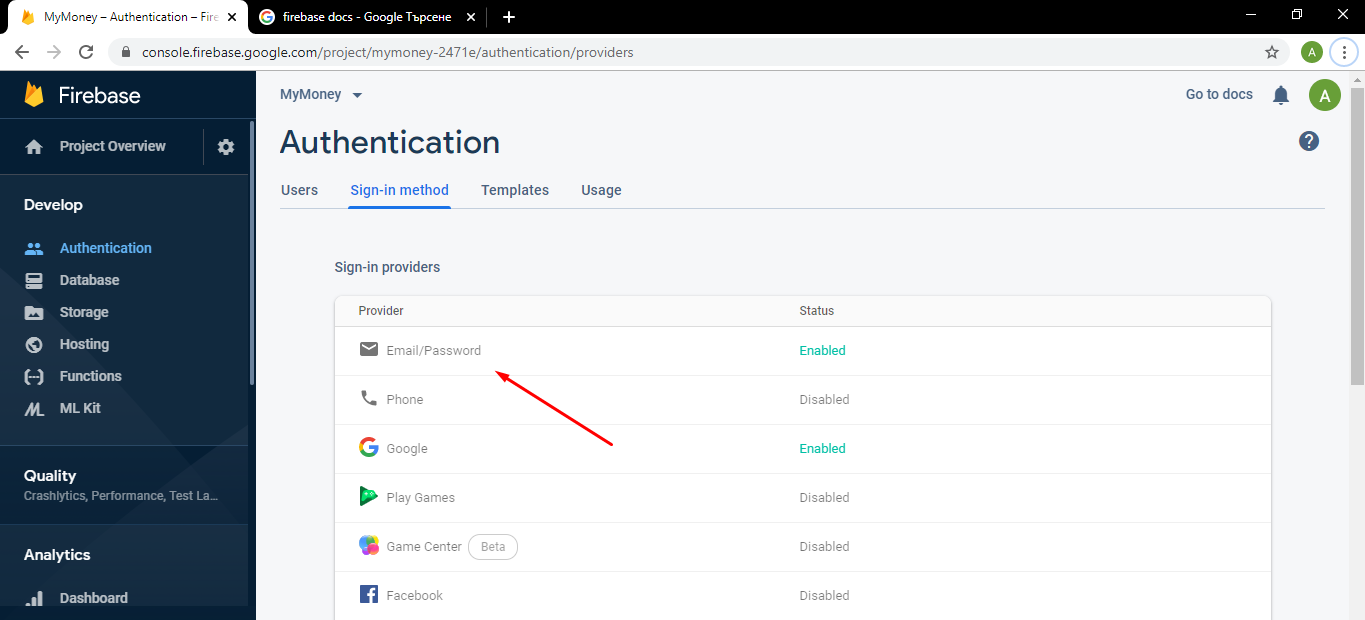
You can find your **config file** in your project, then **Project Settings** > **General**.



**7. Firebase: Register**

**Registering** in firebase is made by the **createUserWithEmailOrPassword** **method**.

To **allow** users to register with **their email and password**, you need to go to your project, then **Authentication (in the side bar)** > **Sign-in method.**



|  |
| --- |
| Create User |
| firebase.auth().createUserWithEmailAndPassword(email, password)  .catch((error) => {       // Handle Errors here.       let errorCode = error.code;       let errorMessage = error.message;       // ...  }); |

**8. Firebase: Login**

**Signing** user in firebase is made by the **signInWithEmailAndPassword method.**

|  |
| --- |
| Login User |
| firebase.auth().signInUserWithEmailAndPassword(email, password)  .catch((error) => {       // Handle Errors here.       let errorCode = error.code;       let errorMessage = error.message;       // ...  }); |

**8. Firebase: Get User Data (data observer)**

**Getting** the user data in firebase is made by the **onAuthStateChanged method.**

|  |
| --- |
| Get User Data |
| firebase.auth().onAuthStateChanged((user) => {      if (user) {          // User is signed in.          let displayName = user.displayName;          let email = user.email;          let emailVerified = user.emailVerified;          let isAnonymous = user.isAnonymous;          let uid = user.uid;          // ...      } else {          // User is signed out.          // ...      }  }); |

**8. Firebase: Logout User**

**Logging out** user in firebase is made by the **signOut method.**

|  |
| --- |
| Sign Out User |
| firebase.auth().signOut()      .then(() => {          // Sign-out successful.      })      .catch((error) => {          // An error happened.      }); |