**Get data from firebase**

1. Install firebase => npm install firebase
2. Create database and get it’s API config from google
3. Make a new js file to initialize this database

import firebase from "firebase";

import "firebase/firestore";

*// Your web app's Firebase configuration*

const firebaseConfig = {

  apiKey: "AIzaSyAYzh4wMkZfryhpkJq3nAIf5fDyhJ3KDm4",

  authDomain: "smoothies-2a5a1.firebaseapp.com",

  databaseURL: "https://smoothies-2a5a1.firebaseio.com",

  projectId: "smoothies-2a5a1",

  storageBucket: "smoothies-2a5a1.appspot.com",

  messagingSenderId: "926768269280",

  appId: "1:926768269280:web:cc860bb16dea4136ffcd85"

};

*// Initialize Firebase*

const db = firebase.initializeApp(firebaseConfig).firestore();

export default db;

**in the file you want to import the data from Firebase**

import db from "@/firebase/init.js";

created(){

      db.collection('collection-name').get()

      .then(allDocuments => {

       allDocuments.forEach (doc =>{

          let smoothie = doc.data();

          smoothie.id = doc.id

          this.smoothies.push(smoothie)

          console.log(this.smoothies)

        })

      })

    }

**Get the two ID’s Firebase**

created() {

            db.collection('collection-name').get()

                .then(allDocuments => {

                    allDocuments.forEach(doc => {

*// Get the Document Outside Main ID*

                        console.log(doc.id)

                    })

                })

        }

*//When Select document, Use Main outside ID*

db.collection("allsmoothies").doc(mainID)

created() {

            db.collection('collection-name').get()

                .then(allDocuments => {

                    allDocuments.forEach(doc => {

*// Get the Document inside ID*

                        console.log(doc.data().id)

                    })

                })

        }

**Delete object from database (firebase)**

deleteSmoothi(id) {

*// delete the object which contain this ID in just database*   
 db.collection('allsmoothies').doc(id).delete()

}

deleteSmoothi(id){

*// delete the object which contain this ID in database & Frontend Side*

db.collection('allsmoothies').doc(id).delete().then(()=>{

    this.smoothies = this.smoothies.filter(smoothie => {

        return smoothie.id != id

    })

})

}

**Set Data To Database (firebase)**

*//Add this data and then Route to Home*

db.collection("allsmoothies").add({

     title: this.title,

     ingredients: this.ingredients,

slug: this.slug

}).then(()=>this.$router.push('/'))

**Users Sign up and authentication**

import firebase from 'firebase';

signup() {

    if (this.alias && this.email && this.password) { *//check of fields are filled*

        this.slug = slugify(this.alias, {

            replacement: '-',

            remove: /[$\*\_+~.()'"!\-:@]/,

            lower: true

        })

        let ref = db.collection('users').doc(this.slug) *//select document with specific id*

        ref.get().then(doc => {

            if (doc.exists) { *//exist is a firebase object to check if this doc exists*

                this.feedback = "This User alReady Here"

            } else { *// not exists so we will add new user*

                firebase.auth().createUserWithEmailAndPassword(this.email, this.password)

*//firebase function to add email and username*

                .then(user =>{

*//user set() function to set document with specific id*

                    ref.set({

                        alias:this.alias,

                        geoLocation:null,

                        user\_id: user.user.uid

                    })

                })

                .then(()=> this.$router.push({name:'GMap'}))

                .catch(error => {this.feedback = error.message})

            }

        })

    } else {

        this.feedback = "Please fill all fields !"

    }

} *// end SignUp function*