1. 프로젝트 폴더 clone 하고 해당 폴더로 이동한다. (이 경우, desktop/push2 폴더)

2. `npm install`

3. `npm install -g firebase-tools`

4. `firebase login`

5. `firebase init functions`

6. `serviceAccountKey.json` 파일 다운받아서, functions 폴더 내부에 넣는다

> 참고링크:) `https://firebase.google.com/docs/admin/setup`

```javascript
//      functions/index.js 작성
const admin = require('firebase-admin');
const functions = require('firebase-functions');
const serviceAccount = require("./serviceAccountKey.json");

admin.initializeApp({
  credential: admin.credential.cert(serviceAccount),
  databaseURL: "https://ssafy-245804.firebaseio.com"
});

var db = admin.firestore();

exports.dbWrite = functions.firestore.document('posts/{any?}').onCreate( event
=> {
    const payload = {
        notification : {
            title: '포스트 생성되었습니다',
            body: '알림아 울려줘'
        }
    }
    console.log('글변경')

    db.collection('tokens').get().then((snapshot)=> {
      snapshot.forEach(doc => {
        console.log(doc.data()['test_token'])
        admin.messaging().sendToDevice(doc.data()['test_token'], payload)
      })
    })
  });
  //  (change, context)
```

8. 함수 deploy 과정 `firebase deploy --only functions --project ssafy-245804`

9. `App.vue` 의 `<script>` 에 작성

```
import Firebaseservice from './services/FirebaseService'

// export default 내부에 작성 (methods 와 mounted)

export default {
  name: 'App',
  store,
  data() {
    return {
      night: false,
    }
  },
  components: {
    'MainHeader' : MainHeader,
    'MainFooter' : MainFooter,
    'Advertisement' : Advertisement,
    BackToTop,
  },

  methods:{
    async check(){
      await Firebaseservice.notificationcheck()
    }
  },

  mounted() {
    this.check()  // 알림허용할지 확인
  }
}
```

10. `FirebaseService.js` 에 작성

```
import 'firebase/messaging'

// const config 수정
const config = {
  projectId: 'ssafy-245804',
  authDomain: 'ssafy-245804.firebaseapp.com',
  apiKey: 'AIzaSyAh53Ji2z1PjJFVo4EvJYe_bN9omsOiYMc',
  databaseURL: 'https://ssafy-245804.firebaseio.com',
  storageBucket: 'ssafy-245804.appspot.com',
  messagingSenderId : "452889627202",
  appId : "1:452889627202:web:b22e00319bd3559b"
}

const messaging = firebase.messaging()
```

```
// export default 에 추가

export default {
  notificationcheck(){
    Notification.requestPermission().then(function(permission){
      if (permission === 'granted') {
        console.log('알림이 허용되었어')
      } else {
        console.log('알림 거부하지마 ㅠㅠ')
      }
    })
  }, // notificationcheck 끝


    gettingtoken(){
    messaging.usePublicVapidKey('BH4H7pKz2nJHRkzwWzgUtSna-zCJzPM-
KeR_MemZjinDVVuEzd6FulRn_P4zIxZFObFkE_mvIOnKHrrBING55ZE')
    messaging.getToken().then(function(currentToken) {
      if (currentToken) {
        var test_token = currentToken
        console.log(test_token)
        firestore.collection('tokens').doc(test_token).set({
          test_token : test_token
        })
      } else {
        console.log('No Instance ID token available. Request permission to
generate one.');
      }
    }).catch(function(err) {
      console.log('An error occurred while retrieving token. ', err);
    })
  },
}
```

11. `MainHeader.vue` 에 `로그인함수 3가지 내부의 가장 마지막 줄에` 작성

```
// 아래처럼
async loginWithMail(){
    firebase.auth().signInWithEmailAndPassword(this.loginEmail,
this.loginPassword)
    .then((user)=>{
      this.$store.state.accessToken = user.user.refreshToken
      this.$store.state.user = user.user
      localStorage.setItem('accessToken', this.$store.state.accessToken)
      localStorage.setItem('user', JSON.stringify(user.user))
      alert(this.$store.state.user.email + "님 로그인 되었습니다")
      this.closeDialog()
      this.$router.replace('/')
```

```
        FirebaseService.addLog(this.$store.state.user.uid,"login with mail")

        // 로그인하면, token 얻기
        FirebaseService.gettingtoken()


    })
    .catch((error)=>{
      alert(error)
    })
  },
```

12. `public` 폴더 내부에, `firebase-messaging-sw.js` 만들고 작성

```javascript
// 서비스워커 파일

importScripts('https://www.gstatic.com/firebasejs/4.8.1/firebase-app.js');
importScripts('https://www.gstatic.com/firebasejs/4.8.1/firebase-messaging.js');

const config = {
  projectId: 'ssafy-245804',
  authDomain: 'ssafy-245804.firebaseapp.com',
  apiKey: 'AIzaSyAh53Ji2z1PjJFVo4EvJYe_bN9oms0iYMc',
  databaseURL: 'https://ssafy-245804.firebaseio.com',
  storageBucket: 'ssafy-245804.appspot.com',
  messagingSenderId: "452889627202",
  appId: "1:452889627202:web:b22e00319bd3559b"
}

firebase.initializeApp(config);

const messaging = firebase.messaging()

// If you would like to customize notifications that are received in the
// background (Web app is closed or not in browser focus) then you should
// implement this optional method.
// [START background_handler]
messaging.setBackgroundMessageHandler(function(payload) {
  console.log('[firebase-messaging-sw.js] Received background message ',
payload);
  // Customize notification here
  var notificationTitle = 'Background Message Title';
  var notificationOptions = {
    body: 'Background Message body.',
    icon: '/firebase-logo.png'
  };
```

```javascript
  return self.registration.showNotification(notificationTitle,
    notificationOptions);
});
// [END background_handler]
```