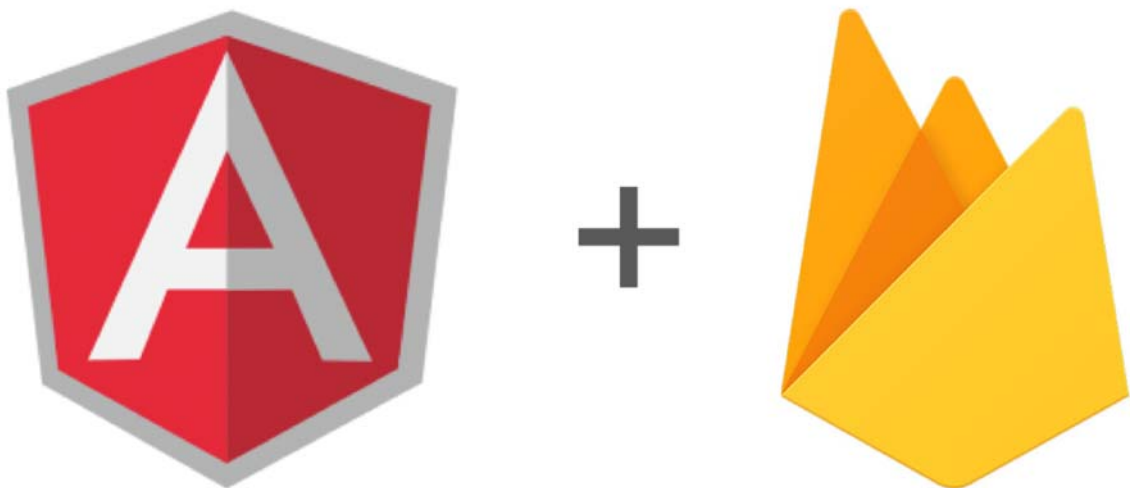


13.30 Angular Deploy - **Firebase**

Firebase를 사용한 Angular 애플리케이션 배포



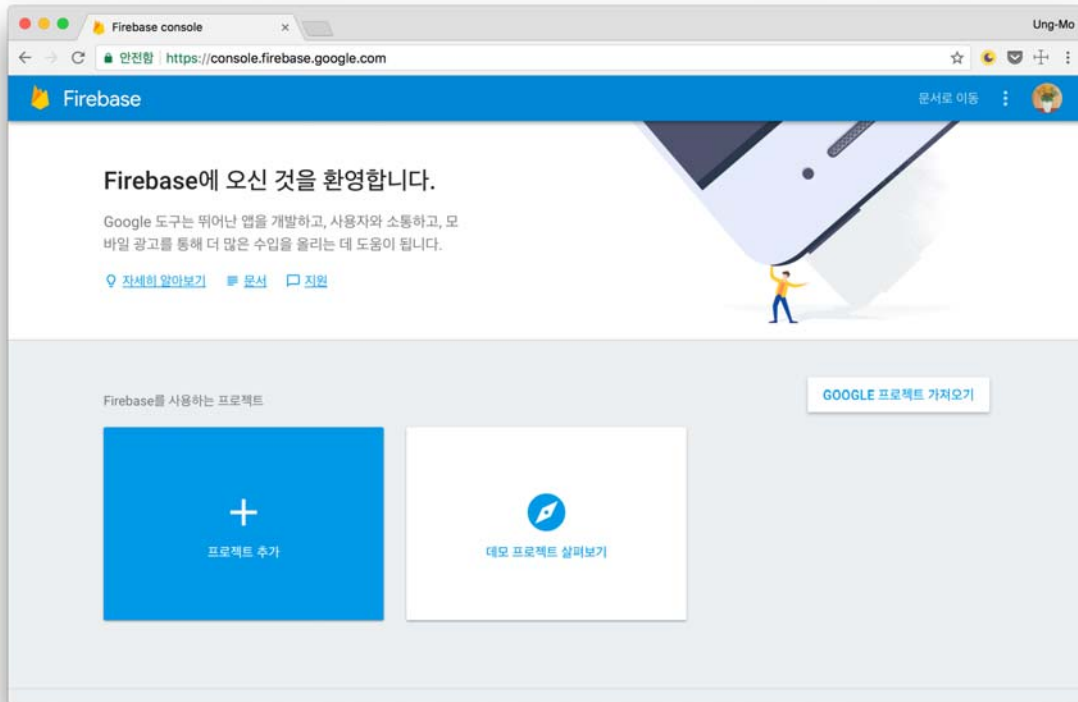
1. Firebase 소개

파이어베이스는 모바일 앱 및 웹 애플리케이션 개발에 필요한 데이터베이스, 사용자 인증, 호스팅, 스토리지, 크래시리포팅(문제보고), 퍼포먼스, 광고 등을 API의 형태로 제공하는 백엔드 서비스(BaaS, Backend as a service)이다. 파이어베이스는 유료 서비스이지만 1GB의 스토리지와 월 10GB의 트래픽은 무료로 제공된다.

이번 장에서는 파이어베이스의 사용자 인증, 데이터베이스, 호스팅 기능을 사용하여 Angular 애플리케이션을 배포하여 보자.

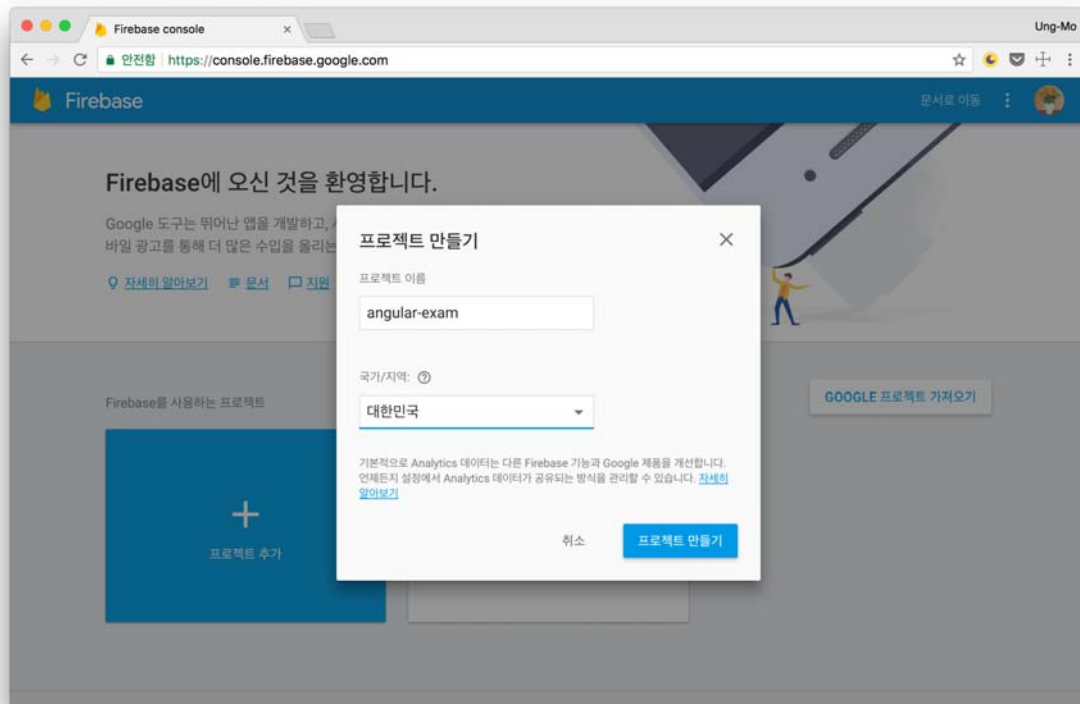
2. Firebase 프로젝트 생성

구글 계정을 준비하고 **Firebase Console**로 이동한다.



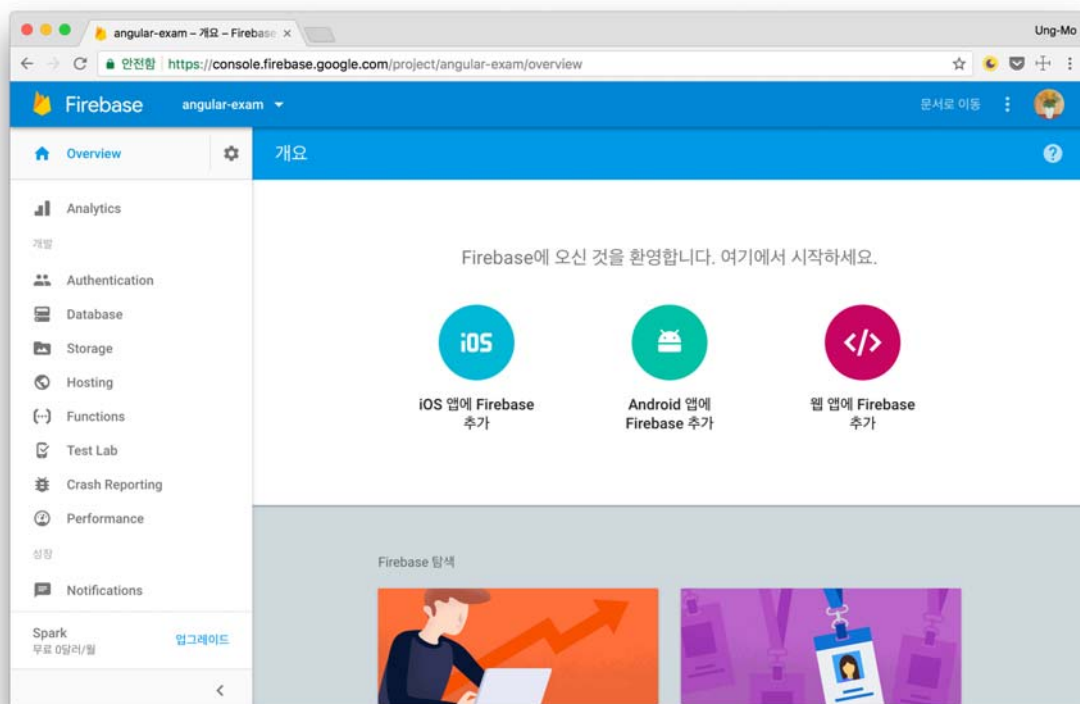
Firebase Console

‘프로젝트 추가’를 클릭하고 ‘프로젝트 이름’과 ‘국가/지역’을 입력한 후, ‘프로젝트 만들기’를 클릭하여 새로운 프로젝트를 생성한다.



Firebase 프로젝트 생성

정상적으로 프로젝트가 생성되면 아래의 프로젝트 개요 화면으로 이동한다.



Firebase 프로젝트 개요

3. Firebase CLI 설치

Firebase CLI는 Firebase 프로젝트 관리, 조회, 배포를 위한 여러 가지 도구를 제공한다. Firebase CLI를 설치하기 위해서 사전에 Node.js가 설치가 필요하다.

- Node.js 설치와 npm 업데이트

npm을 사용하여 Firebase CLI를 설치하도록 한다.

BASH

```
$ npm install -g firebase-tools
$ firebase --version
3.9.2
```

Firebase CLI가 정상적으로 설치되었으면 Google 계정으로 로그인한다.

BASH

```
$ firebase login
```

로그인을 실시하면 로컬 컴퓨터를 Firebase 계정에 연결하고 프로젝트에 대한 액세스를 허용한다. 정상적으로 로그인되었는지 테스트하려면 `firebase list`를 실행하여 Firebase 프로젝트 목록이 표시되는지 확인한다. 이 목록이 Firebase 콘솔 상의 프로젝트와 일치해야 한다.

BASH

```
$ firebase list
```

Name			Project ID / Instance		Permissions	
Firebase Demo Project			fir-demo-project		Viewer	
angular-exam			angular-exam		Owner	

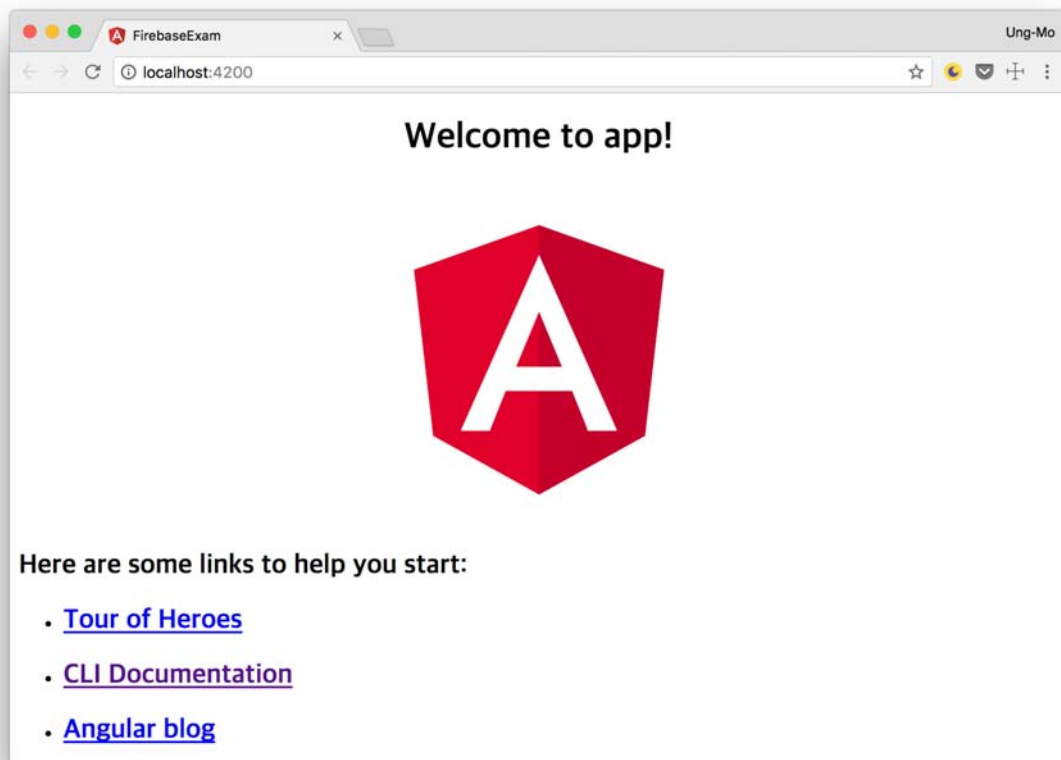
4. Deploy

4.1 Firebase Hosting을 사용한 로컬 프로젝트 Deploy 설정

이제 Firebase Hosting을 사용하여 로컬 프로젝트를 배포하도록 하자. 기존의 로컬 프로젝트를 사용하여도 좋지만 firebase와의 배포 테스트를 위해 새로운 Angular 프로젝트를 생성한다.

BASH

```
$ ng new firebase-exam  
$ cd firebase-exam  
$ ng serve -o
```



Angular 프로젝트 생성

Firebase와 연동할 프로젝트에는 firebase.json 파일이 필요하다. 이것을 생성하기 위해 아래의 명령어를 실행한다.

실시간 데이터베이스와 호스팅을 사용할 것이므로 Database와 Hosting을 선택(space)한 후, 엔터를 입력하여 다음으로 진행한다.

BASH

```
$ cd firebase-exam
$ firebase init

? Which Firebase CLI features do you want to setup for this folder? Press Space to select
features, then Enter to confirm your choices. (Press <space> to select)
  ☐ Database: Deploy Firebase Realtime Database Rules
  ☐ Functions: Configure and deploy Cloud Functions
> ☒ Hosting: Configure and deploy Firebase Hosting sites
```

Firebase Console에서 생성한 Firebase 프로젝트를 선택한 후, 엔터를 입력하여 다음으로 진행한다.

BASH

```
=== Project Setup

First, let's associate this project directory with a Firebase project.
You can create multiple project aliases by running firebase use --add,
but for now we'll just set up a default project.

? Select a default Firebase project for this directory:
  [don't setup a default project]
  Firebase Demo Project (fir-demo-project)
> angular-exam (angular-exam)
  [create a new project]
```

실시간 데이터베이스와 관련된 설정 파일의 이름을 묻는 질문에는 기본 설정인 database.rules.json을 그대로 사용할 것이므로 엔터를 입력하여 다음으로 진행한다.

BASH

```
=== Database Setup

Firebase Realtime Database Rules allow you to define how your data should be
structured and when your data can be read from and written to.
```

```
? What file should be used for Database Rules? (database.rules.json)
```

다음은 호스팅 설정에 관한 질문이 이어진다. 로컬 프로젝트의 빌드 결과물은 dist 폴더에 생성되므로 dist를 입력하고 엔터를 입력하여 다음으로 진행한다.

```
BASH
```

```
=== Hosting Setup
```

```
Your public directory is the folder (relative to your project directory) that will contain Hosting assets to be uploaded with firebase deploy. If you have a build process for your assets, use your build's output directory.
```

```
? What do you want to use as your public directory? (public) dist
```

호스팅 설정에서 SPA 설정 여부를 묻는 질문에는 y를 입력하고 엔터를 입력하여 다음으로 진행한다.

```
BASH
```

```
? Configure as a single-page app (rewrite all urls to /index.html)? (y/N) y
```

dist/index.html를 덮어 쓸것인지에 대한 질문에는 N를 입력하고 엔터를 입력하여 다음으로 진행한다.

```
BASH
```

```
? File dist/index.html already exists. Overwrite? (y/N) N
```

프로젝트 폴더에 firebase.json, .firebaserc, database.rules.json이 생성되고 Firebase 초기화가 종료된다.

```
BASH
```

```
i Writing configuration info to firebase.json...
i Writing project information to .firebaserc...
```

✓ Firebase initialization complete!

생성된 firebase.json의 내용은 아래와 같다.

JSON

```
{
  "database": {
    "rules": "database.rules.json"
  },
  "hosting": {
    "public": "dist",
    "ignore": [
      "firebase.json",
      "**/.*",
      "**/node_modules/**"
    ],
    "rewrites": [
      {
        "source": "**",
        "destination": "/index.html"
      }
    ]
  }
}
```

4.2 Firebase CLI로 로컬 프로젝트 실행

로컬 프로젝트를 파이어베이스 CLI로 실행하여 보자. 먼저 로컬 프로젝트를 빌드하고 파이어베이스 CLI 내장 테스트 서버를 실행한다.

BASH

```
$ ng build
$ firebase serve
=== Serving from '/Users/leeungmo/Desktop/firebase-exam'...
```



```
i  hosting: Serving hosting files from: dist
✓  hosting: Local server: http://localhost:5000
```

<http://localhost:5000>에 접속하여 애플리케이션의 동작을 확인한다.

4.3 Firebase Hosting에의 Deploy

프로젝트 빌드 결과물을 Firebase 호스팅을 통해 원격 배포하기 위해서는 아래의 명령을 실행한다.

```
BASH

$ firebase deploy

=== Deploying to 'angular-exam'...

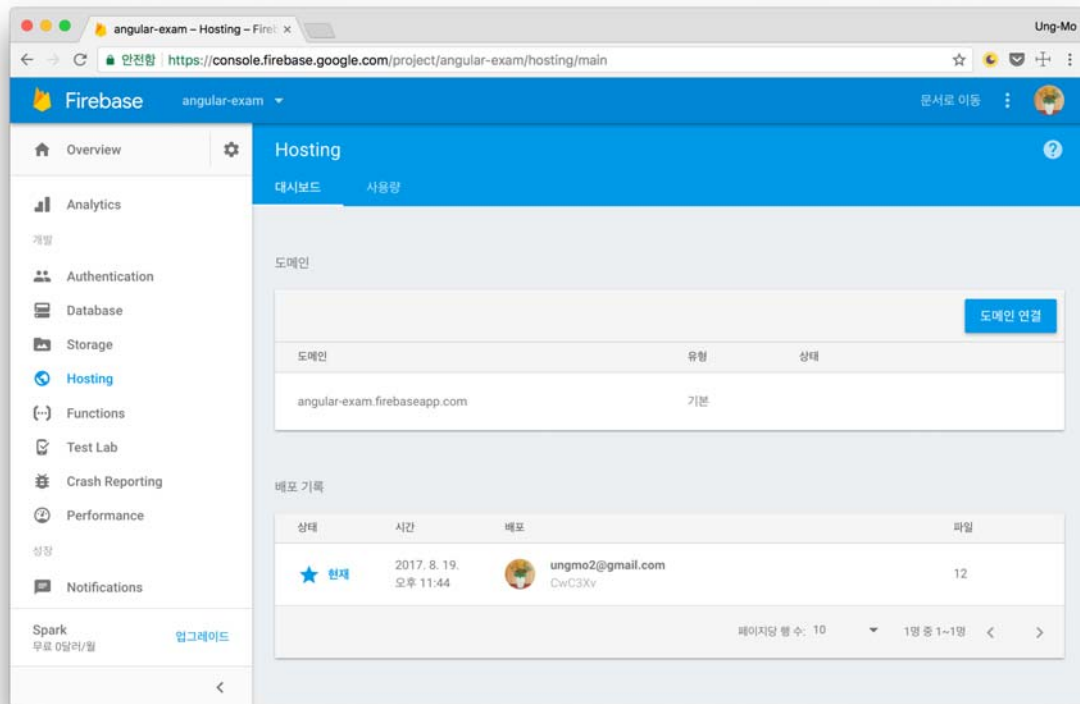
i  deploying database, hosting
✓  database: rules ready to deploy.
i  hosting: preparing dist directory for upload...
✓  hosting: 12 files uploaded successfully
i  starting release process (may take several minutes)...

✓  Deploy complete!

Project Console: https://console.firebase.google.com/project/angular-exam/ov
erview
Hosting URL: https://angular-exam.firebaseio.com
```

로컬 프로젝트의 빌드 결과물이 Firebase Console에서 생성한 Firebase 프로젝트에 배포되었다. 호스팅된 결과물은 <https://angular-exam.firebaseio.com>에서 확인할 수 있다.

Firebase Console에서 호스팅 결과를 확인하기 위해서는 메뉴의 'Hosting'을 클릭한다.



호스팅 결과 확인

Reference

- [Firebase](#)
- [Firebase Console](#)
- [Firebase CLI](#)