

21 파이어스토어, 스토리지, FCM

모바일 앱을 개발할 때 앱에서 발생하는 데이터나 파일을 서버에 저장하는 기능을 자주 사용한다. 파이어베이스에서 이런 백엔드 기능을 서버리스로 구현할 수 있도록 파이어스토어와 스토리지 기능을 제공한다.

1. 클라우드 파이어스토어

파이어베이스는 클라우드 파이어스토어와 실시간 데이터베이스 2가지를 클라우드 기반으로 한 데이터베이스를 제공한다. 실시간 데이터베이스는 여러 클라이언트에서 상태를 실시간으로 동기화해야 하는 모바일 앱을 만드는 솔루션이다.

- 파이어스토어 사용 설정

파이어스토어를 사용하려면..

먼저 파이어베이스에서 데이터베이스를 만들어야 한다.

만들었으면 그래들 파일에 파이어스토어 라이브러리 등록.

- 파이어스토어 데이터 모델

파이어스토어는 NoSQL 데이터베이스라서 SQL 데이터베이스와 달리 테이블이나 행이 없다. 그 대신 컬렉션으로 정리되는 문서에 데이터가 저장된다.

각 문서에는 키-값 쌍의 데이터가 저장되며 모든 문서는 컬렉션에 저장된다.

- 파이어스토어 보안 규칙

파이어스토어에 저장된 데이터를 이용할 때는 다양한 보안 규칙을 설정할 수 있다. 읽는데 조건이 없지만 로그인했을 때만 쓸

수 있게 설정할 수도 있다.

- 데이터 저장하기

파이어베이스 콘솔에서 파이어스토어를 만들고 보안 규칙을 설정했다면 이제 안드로이드 앱에서 파이어스토어를 이용할 수 있다. 이때 가장 먼저 파이어스토어 객체를 얻어야 한다. 이렇게 얻은 `Firestore` 객체로 컬렉션을 선택하고 문서를 추가하거나 가져오는 작업을 한다.

- `add()` 함수로 데이터 저장하기

파이어스토어에 데이터가 문서 단위로 저장되고 문서는 컬렉션에 저장된다. 따라서 데이터를 저장하려면 먼저 컬렉션을 선택하고 문서를 작업하는 `CollectionReference` 객체를 얻어야 한다.

다음 코드에서 `db.collection("users")`는 `users`라는 컬렉션을 선택하는 구문이며 만약 해당 컬렉션이 없으면 새로 만듭니다. 그리고 `collection()` 함수의 반환값인 `CollectionReference` 객체의 `add()`, `set()`, `get()` 등의 함수로 문서 작업을 합니다.

- 객체 저장하기

앞선 예에서는 데이터를 `Map`으로 저장했는데 객체의 데이터를 저장할 수도 있습니다. 다음은 `Users` 컬렉션에 `User` 클래스의 객체를 저장하는 예입니다. 앞서 `Map`으로 저장한 문서의 필드는 `name`, `email`, `avg`였지만, 다음 예에서는 `name`, `email`, `avg`, `isAdmin`, `isTop`입니다. 이처럼 같은 컬렉션에 저장된 문서는 필드가 달라도 됩니다.

- 객체 저장하기

```
class User(val name: String, val email: String, val avg: Int,
    @JvmField val isAdmin: Boolean, val isTop: Boolean)
val user = User("kim", "kim@a.com", 20, true, true)
db.collection("users")
    .add(user)
```

- `set()` 함수로 데이터 저장하기

set() 함수는 신규 데이터뿐만 아니라 기존의 데이터를 변경할 때도 사용된다. add() 함수는 CollectionReference 객체에서 제공하므로 문서를 추가할 때 식별자가 자동으로 지정된다.

- 데이터 업데이트와 삭제

set() 함수는 문서 전체를 덮어쓰므로 만약 기존의 특정 필드값만 업데이트하려면 update() 함수이용해야 함.

- 데이터 불러오기

컬렉션의 문서를 가져올 때는 get() 함수 이용.

- get() 함수로 컬렉션의 전체 문서 가져오기
컬렉션을 지정한 다음 get() 함수 호출.

- 단일 문서 가져오기
document() 함수에 식별값으로 문서를 지정한 다음 get() 함수 호출.

- wherexxx() 함수로 조건 설정

2. 파이어베이스 스토리지

파이어베이스 스토리지는 앱의 파일을 저장하는 기능을 제공한다. 스토리지를 이용하면 사용자가 앱에서 사진을 선택하고 서버에 올린 후 다시 특정 시점에 내려받을 수 있도록 할 수 있다.

- 스토리지 사용 설정

스토리지를 이용하려면 파이어베이스 콘솔에서 스토리지를 시작해야 한다.

3. 이미지 공유 앱 만들기

루즈해서 건너 뛴 것

4. 파이어베이스 클라우드 메시징

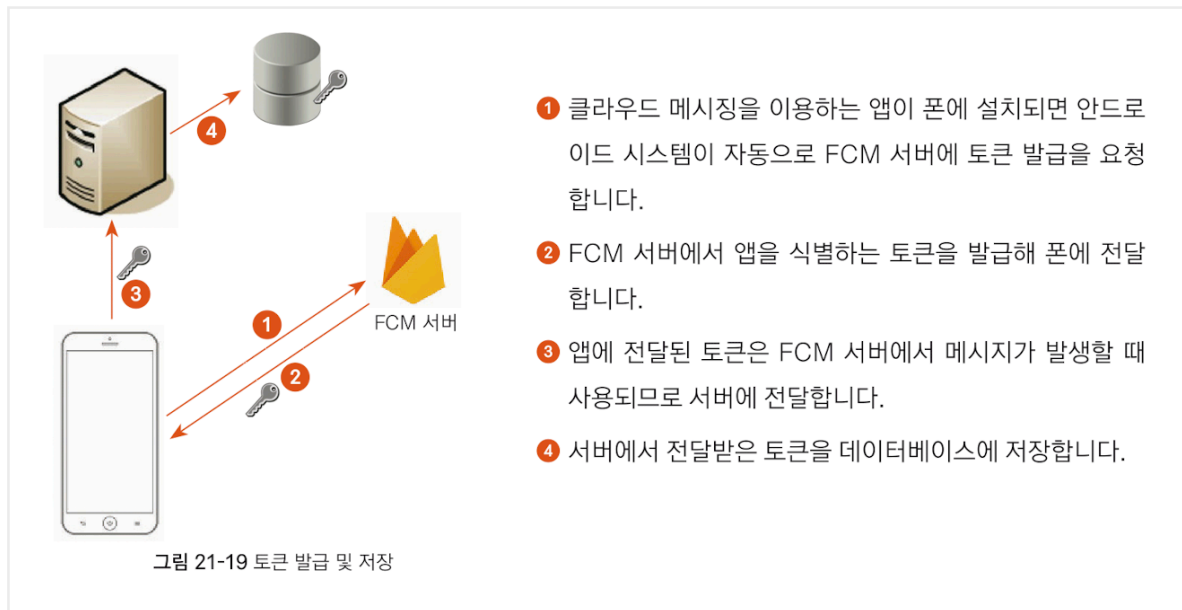
- 클라우드 메시징의 원리

파이어베이스 클라우드 메시징(FCM)은 서버에서 상황이나 데이터가 발생할 때 알림을 전달하는 기능이다. 서버에서 어떤 상황이 발생할 때 클라이언트에 데이터를 전달하는 것을 서버 푸시라고 한다.

항상 서버에 연결해 놓으려면 제약도 있고 효율도 떨어지므로 파이어베이스 클라우드 메시징을 생각할 수 있다. 이는 서버의 데이터를 앱에 직접 전달하지 않고 FCM 서버를 거쳐 앱에 전달하는 방식이다.

- 1단계 : 토큰 발급

FCM 서버에 전달된 데이터를 특정 기기의 앱에 전달하려면 클라우드 메시징을 이용하는 앱을 구분하는 식별값이 필요하다. 이 식별값을 토큰이라고 하며 안드로이드 시스템이 FCM 서버에 자동으로 의뢰해서 발급받는다.



- 2단계 : 서버에서 앱으로 데이터 전송



5. 서버에서 보내는 알림 받기

생략함 ㅅㄱ

● 정리

1. 파이어스토어는 파이어베이스에서 제공하는 데이터베이스이다.
2. 파이어스토어의 데이터는 문서 단위로 저장되며 문서 내에서는 키-값의 형태의 여러 필드가 저장된다.

3. 파이어베이스 스토리지에는 파일을 저장할 수 있다.
4. 파이어베이스 클라우드 메시징은 실시간 서버 푸시를 쉽게 구현할 수 있는 구글의 서비스이다.