1. 프로젝트 생성

프로젝트명 : SamplePush

패키지명 : com.example.samplepush

2. FCM 설정

(1) https://console.firebase.google.com

> [ 프로젝트 만들기 ]

> 구글 계정으로 로그인

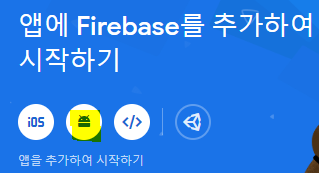
> 프로젝트명 : SamplePush

- 어날리틱스 위치 : 대한민국

(2) 안드로이드에 추가

` FCM 푸시 서비스는 안드로이드, iOS, 웹앱 모두 사용가능

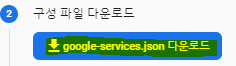
` 가운데 안드로이드 선택 [ Android 앱에 Firebase 추가 ]



` Android 패키지 이름 : 위에 프로젝트 생성시 패키지명(?) - com.example.samplepush

> [ 앱 등록 ]

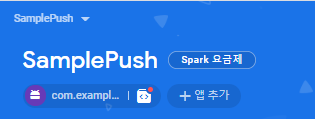
` 구성파일 다운로드 : 프로젝트 / app 아래 저장



` Firebase SDK 추가 : [다음]

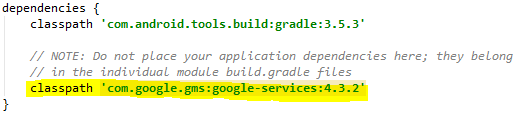
` [ 이 단계 건너뛰기 ]

` 앱 추가 확인

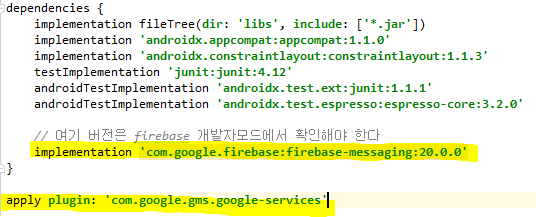


3. 앱에서 푸시 서비스를 이용하기 위한 설정

(1) Gradle Script / build.gradle (Project: SamplePush)



(2) Gradle Script / build.gradle (Module:app)



* android { defaultConfig { minSdkVersion 16 } } } 버전 16으로 변경

(3) [Sync Now] 선택

` 오류 발생시 google-services.json 파일 다시 다운로드

` 프로젝트 폴더 안에 build 폴더와 app / build 폴더 삭제 후 [Sync Project with Gradle Files]로 다시 반영

4. FCM 사용하는 서비스 생성

(1) 서비스 생성

` New > Service > Service : MyMessagingService

` Service 상속을 FirebaseMessagingService 상속으로 변경

|  |
| --- |
| **public class** MyMessagingService **extends** FirebaseMessagingService {  **public** MyMessagingService() {  }   **private static final** String ***TAG*** = **"FMS"**;    @Override  *// 새로운 토큰이 확인했을 때 호출됨* **public void** onNewToken(String token) {  **super**.onNewToken(token);   Log.*d*(***TAG***, **"onNewToken 호출됨 : "** + token);  }   @Override  *// 새로운 메세지를 받았을 땓 호출됨* **public void** onMessageReceived(RemoteMessage remoteMessage) {  Log.*d*(***TAG***, **"onMessageReceived 호출됨."**);   *//----------------------------------  // (2) 메세지 수신 기능 추가  // - 푸시 메시지 받았을 때 그 내용 확인한 후 엑티비티 쪽으로 보내는 메소드 호출* String from = remoteMessage.getFrom();  Map<String, String> data = remoteMessage.getData();  String contents = data.get(**"contents"**);  Log.*d*(***TAG***, **"from : "** + from + **", contents : "** + contents);   sendToActivity(getApplicationContext(), from, contents);  }   *//---------------------------------------------  // (2) 메세지 수신 기능 추가  // - 엑티비티 쪽으로 데이타를 보내기 위해 인텐트 객체를 만들고 startActivity() 호출* **private void** sendToActivity(Context context, String from, String contents) {  Intent intent = **new** Intent(context, MainActivity.**class**);  intent.putExtra(**"from"**, from);  intent.putExtra(**"contents"**, contents);  intent.addFlags(Intent.***FLAG\_ACTIVITY\_NEW\_TASK*** | Intent.***FLAG\_ACTIVITY\_SINGLE\_TOP*** | Intent.***FLAG\_ACTIVITY\_CLEAR\_TOP***);   context.startActivity(intent);  }  } |

(2) 서비스 등록확인 및 추가 : AndroidManifest.xml

|  |
| --- |
| <**service  android:name=".MyMessagingService"  android:enabled="true"  android:exported="true"  android:stopWithTask="false"**>  <**intent-filter**>  <**action android:name="com.google.firebase.MESSAGING\_EVENT"**/>  </**intent-filter**> </**service**> |

(3) 권한 허용 : AndroidManifest.xml

|  |
| --- |
| <**uses-permission android:name="android.permission.INTERNET"**/> |

5. activity\_main.xml ( 화면생성 )

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:app="http://schemas.android.com/apk/res-auto"  xmlns:tools="http://schemas.android.com/tools"  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:orientation="vertical"  tools:context=".MainActivity" >  <Button  android:id="@+id/button"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:text="인스턴스 id 확인하기" />  <TextView  android:id="@+id/textView"  android:layout\_width="match\_parent"  android:layout\_height="0dp"  android:layout\_weight="1"  android:textSize="30sp" />  <ScrollView  android:layout\_width="match\_parent"  android:layout\_height="0dp"  android:layout\_weight="2"  android:background="@android:color/holo\_blue\_bright">  <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:orientation="vertical">  <TextView  android:id="@+id/textView2"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:textSize="20sp" />  </LinearLayout>  </ScrollView>  </LinearLayout> |

6. MainActivity.java

|  |
| --- |
| **public class** MainActivity **extends** AppCompatActivity {   TextView **textView**;  TextView **textView2**;   @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_main***);   **textView** = findViewById(R.id.***textView***);  **textView2** = findViewById(R.id.***textView2***);   *// 등록 id 확인을 위한 리스터 설정하기* FirebaseInstanceId.*getInstance*().getInstanceId().addOnSuccessListener(**this**, **new** OnSuccessListener<InstanceIdResult>() {  @Override  **public void** onSuccess(InstanceIdResult result) {  String newToken = result.getToken();   println(**"등록 id : "** + newToken);  }  });   Button button = findViewById(R.id.***button***);  button.setOnClickListener(**new** View.OnClickListener() {  @Override  **public void** onClick(View v) {  String instanceId = FirebaseInstanceId.*getInstance*().getId();   println(**"확인된 인스턴스 id : "** + instanceId);  }  });  }   **public void** println(String data) {  **textView2**.append(data + **"\n"**);  Log.*d*(**"FMS"**, data);  }   *//--------------------------------------------------  // 서비스로부터 인텐트를 받았을 때의 처리* @Override  **protected void** onNewIntent(Intent intent) {  println(**"onNewIntent 호출됨"**);  **if** (intent != **null**) {  processIntent(intent);  }   **super**.onNewIntent(intent);  }   **private void** processIntent(Intent intent) {  String from = intent.getStringExtra(**"from"**);  **if** (from == **null**) {  println(**"from is null."**);  **return**;  }   String contents = intent.getStringExtra(**"contents"**);  println(**"DATA : "** + from + **", "** + contents);  **textView**.setText(**"["** + from + **"]로부터 수신한 데이터 : "** + contents);  }  } |

[ 데이타 보내기 ]

1. 웹 서버 프로그램에서 jsp 파일 생성

Device id :

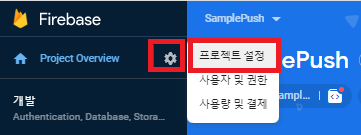
FCM 프로젝트 키값 :

|  |
| --- |
| <%@ page language=*"java"* contentType=*"text/html; charset=UTF-8"*  pageEncoding=*"UTF-8"*%>    <%@ page import=*'java.net.\*, java.io.\*'* %>  <%@ page import=*'org.json.simple.\*'* %>    <%  // FCM 서버에서 보낸 토큰값 : FirebaseInstanceId.getInstance().getToken()  // 콘솔에 출력하고는 그 값을 복사해서 사용해야 한다  // 실제적으로는 이 값을 DB에 넣고 jsp에서는 DB에서 가지고 와서 사용  //userDeviceIdKey is the device id you will query from your database  String userDeviceIdKey = "SamplePush 실행 후 Run 콘솔에서 id 값 복사";    //Method to send Notifications from server to client end.  // firebase.google.com 사이트에서 해당 프로젝트 > 설정 > 클라우드 메세지 > 서버 키 토큰 복사  String AUTH\_KEY\_FCM = "FCM 프로젝트 키값 복사";  String API\_URL\_FCM = "https://fcm.googleapis.com/fcm/send";  String authKey = AUTH\_KEY\_FCM; // You FCM AUTH key  String FMCurl = API\_URL\_FCM;  URL url = **new** URL(FMCurl);  HttpURLConnection conn = (HttpURLConnection) url.openConnection();  conn.setUseCaches(**false**);  conn.setDoInput(**true**);  conn.setDoOutput(**true**);  conn.setRequestMethod("POST");  conn.setRequestProperty("Authorization","key="+authKey);  conn.setRequestProperty("Content-Type","application/json");  JSONObject json = **new** JSONObject();  json.put("to",userDeviceIdKey.trim());  JSONObject info = **new** JSONObject();  info.put("title", "나의캐톡"); // Notification title  info.put("body", "당신은 사랑받기위해"); // Notification body  json.put("notification", info);  OutputStreamWriter wr = **new** OutputStreamWriter(conn.getOutputStream());  System.out.println(">" + json.toString());  wr.write(json.toString());  wr.flush();  conn.getInputStream();  %> |

2. 키값 복사해서 MainActivity.java 소스에 키부분에 복사 ( 2군데 )

- FCM 개발자 콘솔 : <https://console.firebase.google.com/>

- 프로젝트 설정



- 두번째 탭 [클라우디 메시징]

- 서버키를 복사

- 소스중

String AUTH\_KEY\_FCM = "FCM 프로젝트 키값 복사";

3. 실행

(1) 프로젝트 SamplePush 실행 ( 휴대폰에 연결 실행 )

- 화면에 id값 받아서 출력될때까지 기다림

- [Run] 탭에서 출력된 등록 id 값을 복사

D/FMS: 등록 id : AAAAAAAAAAAAAAAAAAAAAAAA

- 소스중

String userDeviceIdKey = "SamplePush 실행 후 Run 콘솔에서 id 값 복사";

(2) jsp 파일을 톰캣으로 구동

- 단어 전송 -> SamplePush에 출력

[참고]

맨 처음에 실행하면 null 값이 나올 수 있음

다시 한번 실행하면 된다