NOTIFICATIONS

We will try to generate a notification now.This notification will be displayed in the notification panel

Create an empty activity **NotificationActivity**

Open **NotificationActivity.java** and add below contents

**package** in.nic.kerala.training;  
  
  
**import** android.annotation.SuppressLint;  
**import** android.app.Notification;  
**import** android.app.NotificationChannel;  
**import** android.app.NotificationManager;  
**import** android.app.PendingIntent;  
**import** android.content.Context;  
**import** android.content.Intent;  
**import** android.graphics.Color;  
**import** android.os.Build;  
**import** android.os.Bundle;  
**import** android.support.annotation.Nullable;  
**import** android.support.v4.app.NotificationCompat;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.TextView;  
  
**public class** NotificationActivity **extends** AppCompatActivity{  
 Button **btn**;  
 @Override  
 **protected void** onCreate(@Nullable Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_notification***);  
 **btn**=(Button)findViewById(R.id.btun);  
 View includedLayout = findViewById(R.id.***head***);  
  
 TextView txttitle= (TextView)includedLayout.findViewById(R.id.***txttitile***);  
 txttitle.setText(**"Notification"**);  
  
 **btn**.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View v) {  
 addNotification();  
 }  
 });  
 }  
  
 **private void** addNotification() {  
  
 NotificationManager notificationManager = (NotificationManager) getSystemService(Context.***NOTIFICATION\_SERVICE***);  
 String NOTIFICATION\_CHANNEL\_ID = **"my\_channel\_id\_01"**;  
  
 **if** (Build.VERSION.***SDK\_INT*** >= Build.VERSION\_CODES.***O***) {  
 @SuppressLint(**"WrongConstant"**) NotificationChannel notificationChannel = **new** NotificationChannel(NOTIFICATION\_CHANNEL\_ID, **"My Notifications"**, NotificationManager.***IMPORTANCE\_MAX***);  
  
 *// Configure the notification channel.* notificationChannel.setDescription(**"Channel description"**);  
 notificationChannel.enableLights(**true**);  
 notificationChannel.setLightColor(Color.***RED***);  
 notificationChannel.setVibrationPattern(**new long**[]{0, 1000, 500, 1000});  
 notificationChannel.enableVibration(**true**);  
 notificationManager.createNotificationChannel(notificationChannel);  
 }  
  
  
 NotificationCompat.Builder notificationBuilder = **new** NotificationCompat.Builder(**this**, NOTIFICATION\_CHANNEL\_ID);  
  
 notificationBuilder.setAutoCancel(**true**)  
 .setDefaults(Notification.***DEFAULT\_ALL***)  
 .setWhen(System.*currentTimeMillis*())  
 .setSmallIcon(R.drawable.***notification***)  
 .setTicker(**"Hearty365"**)  
 .setContentTitle(**"Notifications Example"**)  
 .setContentText(**"This is a test notification"**)  
 .setContentInfo(**"Info"**);  
  
 Intent notificationIntent = **new** Intent(**this**, SqlActivity.**class**);  
 PendingIntent contentIntent = PendingIntent.*getActivity*(**this**, 0, notificationIntent,  
 PendingIntent.***FLAG\_UPDATE\_CURRENT***);  
 notificationBuilder.setContentIntent(contentIntent);  
  
   
 NotificationManager manager = (NotificationManager) getSystemService(Context.***NOTIFICATION\_SERVICE***);  
 manager.notify(0, notificationBuilder.build());  
  
  
 }  
}

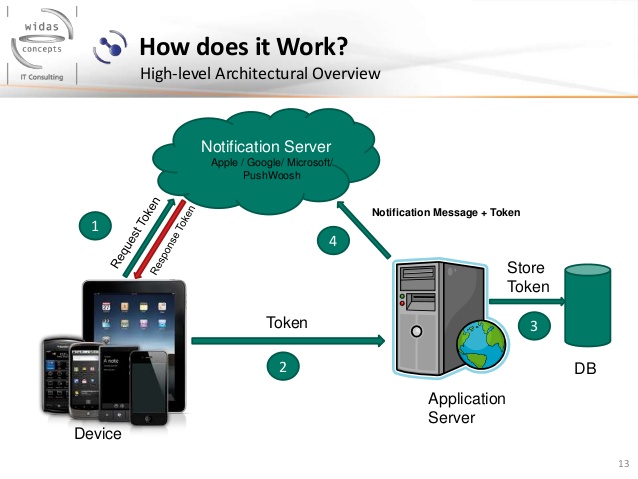
Open **activity\_notification.xml** and add below code

*<?***xml version="1.0" encoding="utf-8"***?>*<**RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"**>  
  
 <**include  
 android:id="@+id/head"  
 layout="@layout/activity\_title"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="wrap\_content"** />  
  
 <**Button  
 android:id="@+id/btun"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerInParent="true"  
 android:layout\_marginTop="10dp"  
 android:background="@drawable/buttoncustom"  
 android:text="Show Notification"  
 android:textColor="#000000"** />  
  
 <**LinearLayout  
 android:id="@+id/in"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="49dp"  
 android:layout\_alignParentBottom="true"  
 android:background="@drawable/fooo"  
 android:orientation="horizontal"** />  
  
</**RelativeLayout**>

**Push Notification**

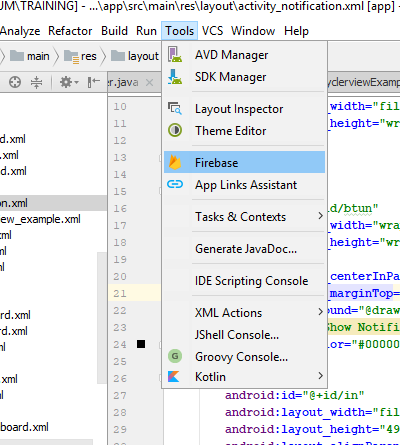
A notification is a message you can display to the user outside of your application's normal UI. When you tell the system to issue a notification, it first appears as an icon in the **notification area**. To see the details of the notification, the user opens the **notification drawer**. Both the notification area and the notification drawer are system-controlled areas that the user can view at any time.

Push Notification Architecture

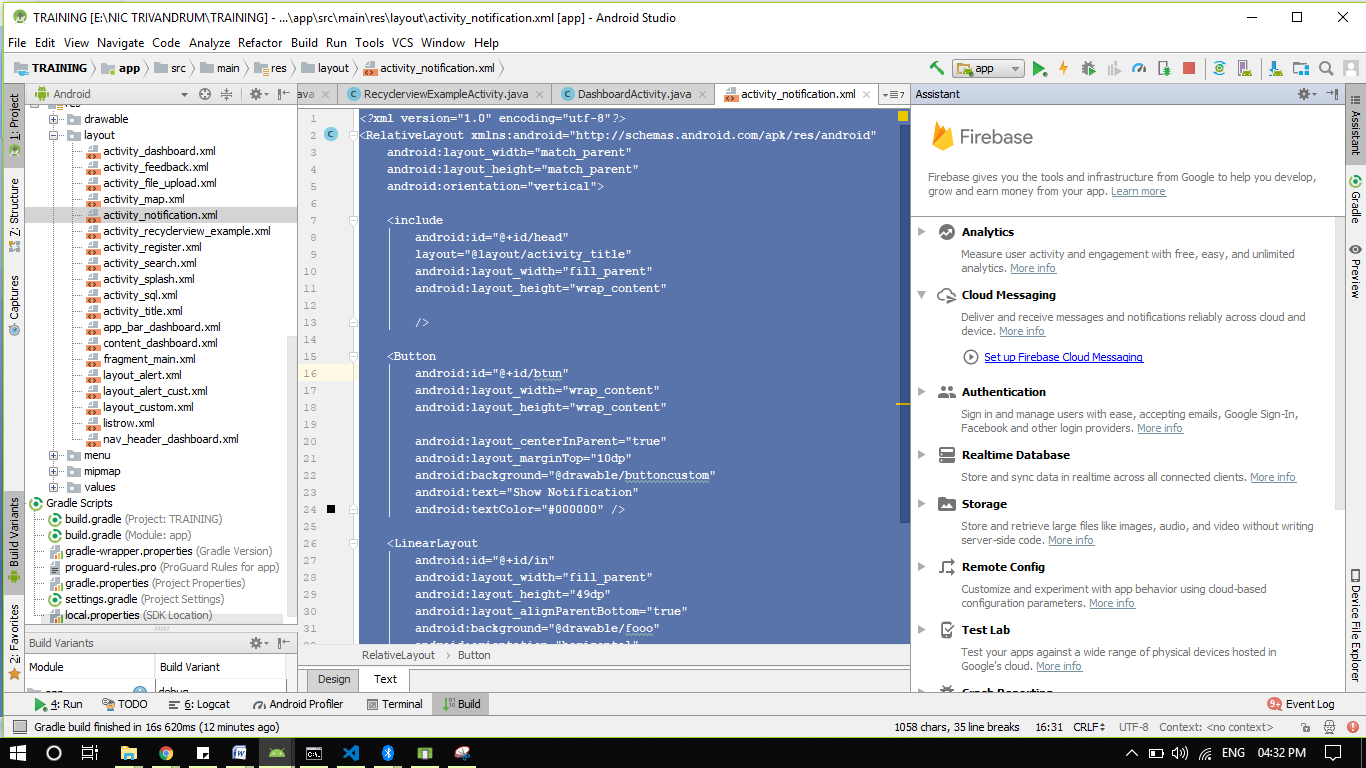


For implementing Notification using **FCM (Firebase Cloud Messaging)** follow these steps:

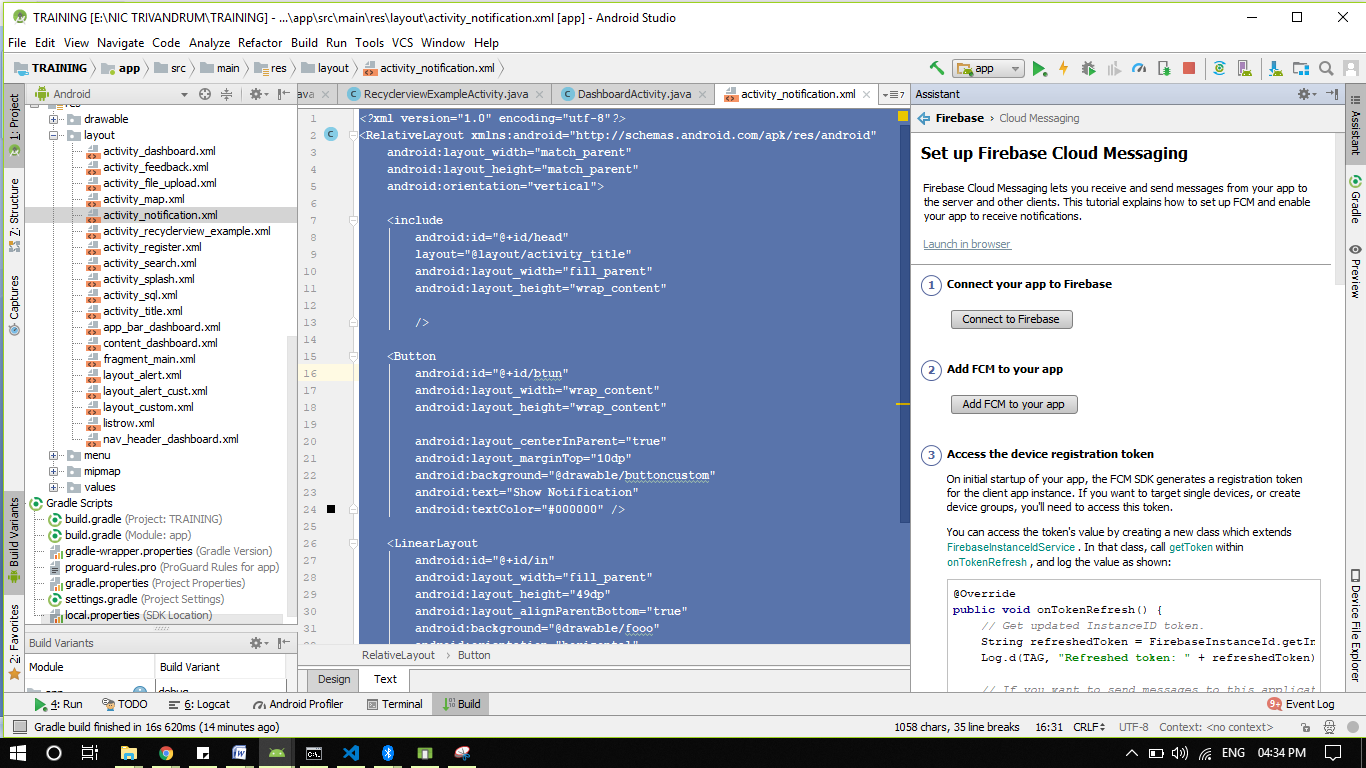
**Go to tools/firebase**



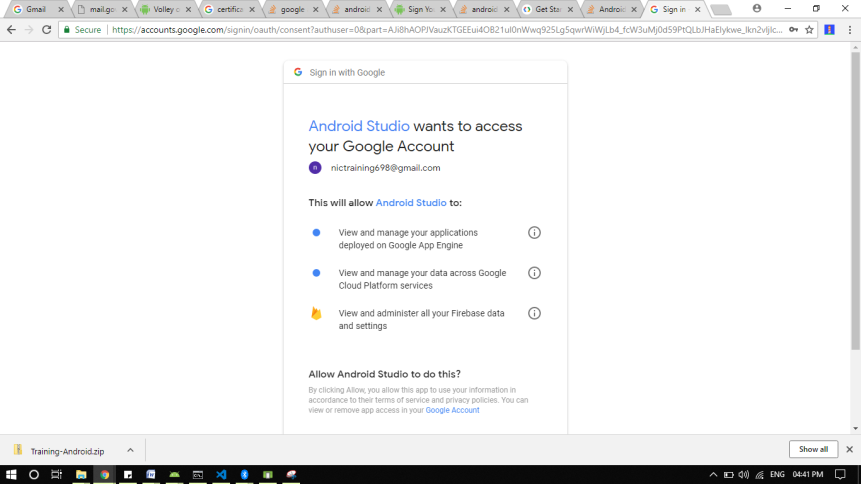
On selecting firebase, you will be able to see a panel in the right side



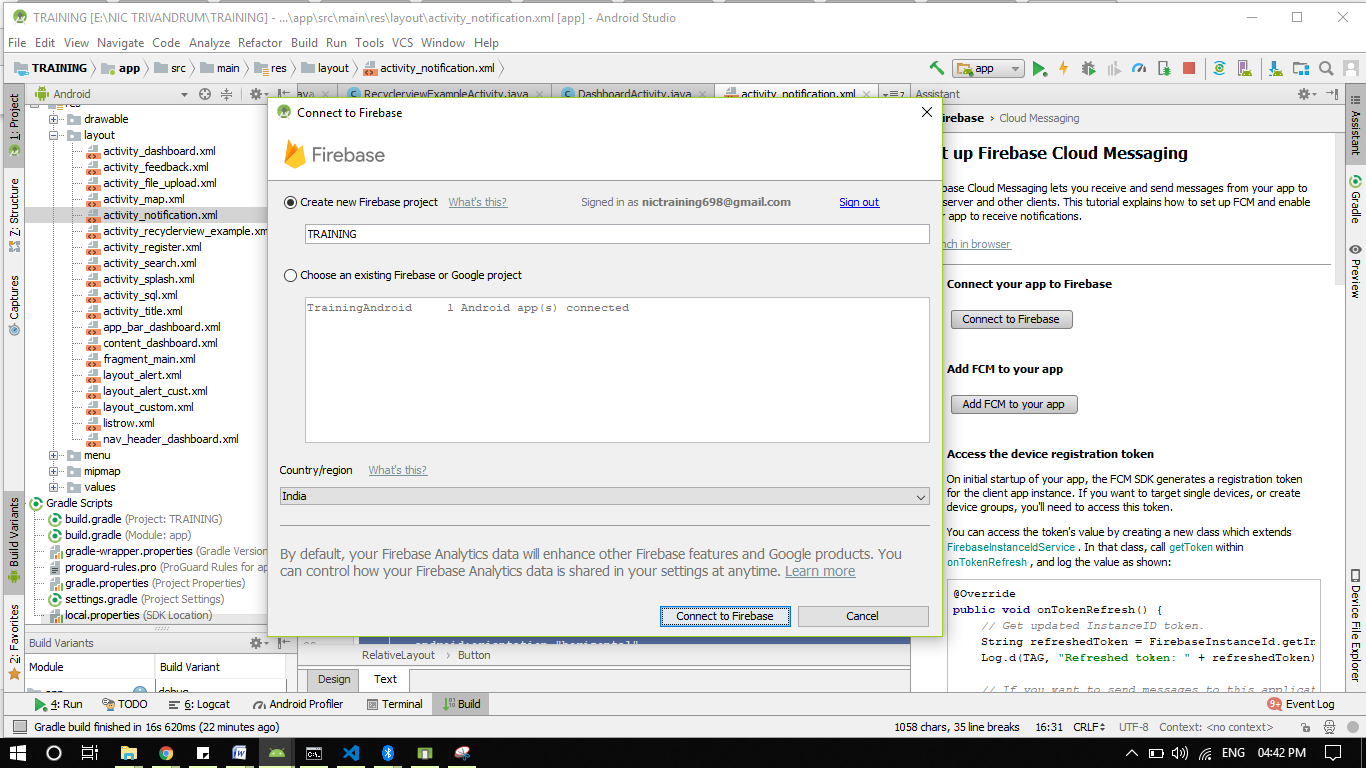
Select Cloud Messaging>Setup Firebase Cloud Messeging



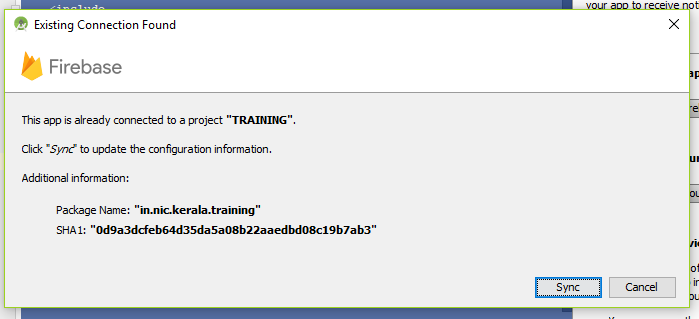
Select **Connect to Firebase.**This will open browser, login with gmail and give permission to android studio



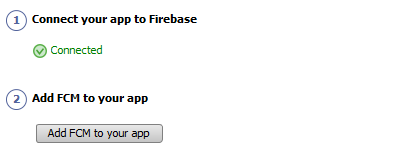
After selecting “allow” button ,open android studio.You can now create a firebase project from within android studio



Select Connect to firebase and wait for few seconds. If there is any error, you can retry with sync option, as shown in the figure below

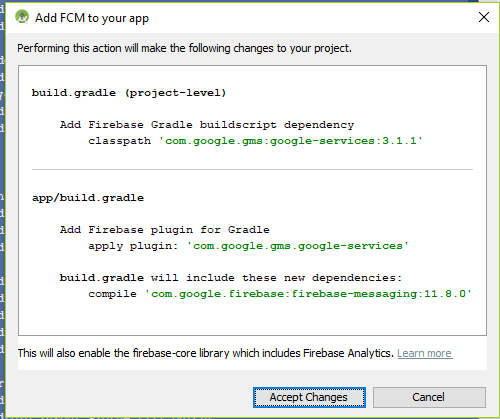


Once connected you can see the status in the same panel



Now click on **Add FCM to your app** Button

A popup may appear as shown below. Select **accept changes**



Now create a class named **MessageReceiver.java** and add below lines

**package** in.nic.kerala.training;  
  
  
**import** android.app.Notification;  
**import** android.app.NotificationChannel;  
**import** android.app.NotificationManager;  
**import** android.app.PendingIntent;  
**import** android.content.Intent;  
**import** android.support.v4.app.NotificationCompat;  
**import** android.support.v4.content.ContextCompat;  
  
**import** com.google.firebase.messaging.FirebaseMessagingService;  
**import** com.google.firebase.messaging.RemoteMessage;  
  
  
**public class** MessageReceiver **extends** FirebaseMessagingService {  
 **private static final int *REQUEST\_CODE*** = 1;  
 **private static final int *NOTIFICATION\_ID*** = 6579;  
  
 **public** MessageReceiver() {  
 **super**();  
 }  
  
 @Override  
 **public void** onMessageReceived(RemoteMessage remoteMessage) {  
 **super**.onMessageReceived(remoteMessage);  
  
 **final** String title = remoteMessage.getData().get(**"title"**);  
 **final** String message = remoteMessage.getData().get(**"body"**);  
  
 showNotifications(title, message);  
 }  
  
 **private void** showNotifications(String title, String msg) {  
 Intent i = **new** Intent(**this**, DashboardActivity.**class**);  
  
 PendingIntent pendingIntent = PendingIntent.*getActivity*(**this**, ***REQUEST\_CODE***,  
 i, PendingIntent.***FLAG\_UPDATE\_CURRENT***);  
  
  
  
  
 NotificationManager notificationManager = (NotificationManager) **this**.getSystemService(getApplicationContext().***NOTIFICATION\_SERVICE***);  
 NotificationCompat.Builder builder = **null**;  
 **if** (android.os.Build.VERSION.***SDK\_INT*** >= android.os.Build.VERSION\_CODES.***O***) {  
 **int** importance = NotificationManager.***IMPORTANCE\_DEFAULT***;  
 NotificationChannel notificationChannel = **new** NotificationChannel(**"ID"**, **"Sagara -Mobile App"**, importance);  
 notificationManager.createNotificationChannel(notificationChannel);  
 builder = **new** NotificationCompat.Builder(getApplicationContext(), notificationChannel.getId());  
 builder.setContentIntent(pendingIntent);  
 } **else** {  
  
 builder = **new** NotificationCompat.Builder(getApplicationContext());  
 builder.setContentIntent(pendingIntent);  
 }  
  
 builder = builder  
 .setSmallIcon(R.mipmap.***ic\_launcher***)  
 .setColor(ContextCompat.*getColor*(getApplicationContext(), R.color.***colorAccent***))  
 .setContentTitle(title)  
 .setTicker(**"Sagara -Mobile App"**)  
 .setContentText(msg)  
 .setDefaults(Notification.***DEFAULT\_ALL***)  
 .setAutoCancel(**true**);  
 notificationManager.notify(***NOTIFICATION\_ID***, builder.build());  
  
  
  
 }  
}

Create another class **InstanceIdService.java** and paste below code

**package** in.nic.kerala.training;  
  
  
**import** android.util.Log;  
  
**import** com.google.firebase.iid.FirebaseInstanceId;  
**import** com.google.firebase.iid.FirebaseInstanceIdService;  
  
 **public class** InstanceIdService **extends** FirebaseInstanceIdService {  
 **public** InstanceIdService() {  
 **super**();  
 }  
  
 @Override  
 **public void** onTokenRefresh() {  
 **super**.onTokenRefresh();  
  
 String token = FirebaseInstanceId.*getInstance*().getToken();  
  
 Log.*d*(**"Firebase"**, **"token "**+ token);  
  
 **if**(token!=**null**) {  
 **if**(!token.trim().isEmpty()) {  
 *//store locally if u want* }  
 }  
  
  
 *//sends this token to the server* sendToServer(token);  
  
  
 }  
  
 **private void** sendToServer(String token) {  
  
 *//here you can write  
 //logic to send this token  
 //to your server* }  
}

Now add below lines in androidmanifest.xml

<**service android:name=".MessageReceiver"**>  
 <**intent-filter**>  
 <**action android:name="com.google.firebase.MESSAGING\_EVENT"** />  
 </**intent-filter**>  
</**service**>  
<**service android:name=".InstanceIdService"**>  
 <**intent-filter**>  
 <**action android:name="com.google.firebase.INSTANCE\_ID\_EVENT"** />  
 </**intent-filter**>  
</**service**>

At this point of time, your build.gradle(App) file will look like

apply **plugin**: **'com.android.application'**android {  
 compileSdkVersion 27  
 defaultConfig {  
 applicationId **"in.nic.kerala.training"** minSdkVersion 17  
 targetSdkVersion 27  
 versionCode 1  
 versionName **"1.0"** testInstrumentationRunner **"android.support.test.runner.AndroidJUnitRunner"** }  
  
 buildTypes {  
 release {  
 minifyEnabled **true** proguardFiles getDefaultProguardFile(**'proguard-android.txt'**), **'proguard-rules.pro'** }  
 }  
}  
  
dependencies {  
 implementation fileTree(**include**: [**'\*.jar'**], **dir**: **'libs'**)  
 implementation **'com.android.support:appcompat-v7:27.1.1'** implementation **'com.android.support.constraint:constraint-layout:1.1.0'** implementation **'com.android.support:support-v4:27.1.1'** implementation **'com.android.support:design:27.1.1'** implementation **'com.google.android.gms:play-services-maps:15.0.0'** implementation **'com.google.firebase:firebase-messaging:15.0.0'** testImplementation **'junit:junit:4.12'** androidTestImplementation **'com.android.support.test:runner:1.0.2'** androidTestImplementation **'com.android.support.test.espresso:espresso-core:3.0.2'** implementation **'com.android.volley:volley:1.1.0'**}  
  
apply **plugin**: **'com.google.gms.google-services'**

Here we are implelementing notification using firebase cloud messaging (FCM). Now go to firebase console

https://console.firebase.google.com/?pli=1

Step1:

Create a project with this package name. Add google-services.json that we will get during creating a project in firebase console to the project.

Step2:

Select Notification in side menu and select new message. Enter firebase id generated while running the app and type the message in the space provided. Then send the message. On completion of sending message, we will recieve a Notification in phone. This is an example of implementing push notification through FCM.

Sending message from firebase console is not always an optimal solution for lot of apps. Huge number of apps requires an automatic notification to user whenever there is an event, which is not possible using firebase console. To overcome this problem you need to integrate the firebase API from your backend server.

