Build.gradle(app)

apply plugin: **'com.android.application'**android {  
 compileSdkVersion 23  
 buildToolsVersion **'23.0.2'** dexOptions {  
 incremental **true** javaMaxHeapSize **"4g"** }  
  
  
  
 defaultConfig {  
 applicationId **"in.djtrinity.www.trinity"** minSdkVersion 17  
 targetSdkVersion 22  
 versionCode 3  
 versionName **"1.2"** multiDexEnabled **true** }  
 buildTypes {  
 release {  
 minifyEnabled **false** proguardFiles getDefaultProguardFile(**'proguard-android.txt'**), **'proguard-rules.pro'** }  
 }  
}  
repositories {  
 maven { url **"https://jitpack.io"** }  
}  
  
  
dependencies {  
 compile fileTree(dir: **'libs'**, include: [**'\*.jar'**])  
 testCompile **'junit:junit:4.12'** compile files(**'libs/commons-net-3.4.jar'**)  
 compile **'com.android.support:support-v4:23.1.1'** compile(**'com.mcxiaoke.viewpagerindicator:library:2.4.1'**) {  
 exclude module: **'support-v4'**;  
 }  
 *//compile 'com.google.android.gms:play-services-gcm:9.2.0'* compile **'com.google.zxing:core:3.3.0'** compile **'com.android.support:design:23.1.0'** compile **'com.android.support:recyclerview-v7:23.1.0'** compile **'com.android.support:cardview-v7:23.1.0'** compile **'com.google.android.gms:play-services:10.0.1'** compile **'com.mcxiaoke.volley:library-aar:1.0.0'** compile **'org.apache.httpcomponents:httpcore:4.4.1'** compile **'org.apache.httpcomponents:httpclient:4.5'** compile **'com.android.support:support-v4:23.1.0'** compile **'com.google.firebase:firebase-core:10.0.1'** compile **'com.google.firebase:firebase-messaging:10.0.1'** compile **'com.github.PhilJay:MPAndroidChart:v2.0.9'** compile **'com.google.android.gms:play-services-ads:10.0.1'** compile **'com.google.android.gms:play-services-identity:10.0.1'** compile **'com.google.android.gms:play-services-gcm:10.0.1'**}  
  
apply plugin: **'com.google.gms.google-services'**

**//add this code to IDPTFragment.class**

**package** in.djtrinity.www.trinity;  
  
**import** android.app.Fragment;  
**import** android.app.ProgressDialog;  
**import** android.content.Context;  
**import** android.net.ConnectivityManager;  
**import** android.net.NetworkInfo;  
**import** android.os.Bundle;  
**import** android.view.LayoutInflater;  
**import** android.view.View;  
**import** android.view.ViewGroup;  
**import** android.webkit.WebSettings;  
**import** android.webkit.WebView;  
**import** android.webkit.WebViewClient;  
**import** android.widget.Toast;  
  
**import** android.graphics.Color;  
**import** android.os.Bundle;  
**import** android.support.v7.app.ActionBarActivity;  
  
**import** com.github.mikephil.charting.charts.BarChart;  
**import** com.github.mikephil.charting.data.BarData;  
**import** com.github.mikephil.charting.data.BarDataSet;  
**import** com.github.mikephil.charting.data.BarEntry;  
**import** com.github.mikephil.charting.utils.ColorTemplate;  
  
**import** java.util.ArrayList;  
  
**public class** IDPTFragment **extends** Fragment {  
  
  
 */\*private WebView webView;  
 private ProgressDialog progressDialog;  
  
 public IDPTFragment() {  
 }\*/* **public** Float **points**[]={4f,10f,25f,345f,67f,78f,346f};  
  
  
 @Override  
 **public** View onCreateView(LayoutInflater inflater, ViewGroup container, Bundle savedInstanceState) {  
 View view = inflater.inflate(R.layout.***fragment\_idpt***, container, **false**);  
 BarChart chart = (BarChart) view.findViewById(R.id.***chart***);  
  
 BarData data = **new** BarData(getXAxisValues(), getDataSet());  
 chart.setData(data);  
 chart.setDescription(**"My Chart"**);  
 chart.animateXY(2000, 2000);  
 chart.invalidate();  
  
  
 **return** view;  
  
  
 }  
 **private** ArrayList<BarDataSet> getDataSet() {  
 ArrayList<BarDataSet> dataSets = **null**;  
  
 ArrayList<BarEntry> valueSet1 = **new** ArrayList<>();  
 BarEntry v1e1 = **new** BarEntry(**points**[0], 0); *// Jan* valueSet1.add(v1e1);  
 BarEntry v1e2 = **new** BarEntry(**points**[1], 1); *// Feb* valueSet1.add(v1e2);  
 BarEntry v1e3 = **new** BarEntry(**points**[2], 2); *// Mar* valueSet1.add(v1e3);  
 BarEntry v1e4 = **new** BarEntry(**points**[3], 3); *// Apr* valueSet1.add(v1e4);  
 BarEntry v1e5 = **new** BarEntry(**points**[4], 4); *// May* valueSet1.add(v1e5);  
 BarEntry v1e6 = **new** BarEntry(**points**[5], 5); *// Jun* valueSet1.add(v1e6);  
 BarEntry v1e7 = **new** BarEntry(**points**[6], 5); *// Jun* valueSet1.add(v1e7);  
  
  
  
 BarDataSet barDataSet1 = **new** BarDataSet(valueSet1, **"Department"**);  
 barDataSet1.setColors(ColorTemplate.***COLORFUL\_COLORS***);  
  
 dataSets = **new** ArrayList<>();  
 dataSets.add(barDataSet1);  
 *// dataSets.add(barDataSet2);* **return** dataSets;  
 }  
  
 **private** ArrayList<String> getXAxisValues() {  
 ArrayList<String> xAxis = **new** ArrayList<>();  
 xAxis.add(**"I.T"**);  
 xAxis.add(**"COMPS"**);  
 xAxis.add(**"BIO-PROD"**);  
 xAxis.add(**"EXTC"**);  
 xAxis.add(**"ELEX"**);  
 xAxis.add(**"MECH"**);  
 xAxis.add(**"CHEM"**);  
 **return** xAxis;  
 }  
}  
 */\* webView = (WebView) view.findViewById(R.id.idpt\_webview);  
 webView.setWebViewClient(new MyBrowser());  
  
 String url = "http://www.djtrinity.in/table1/";  
  
 progressDialog = new ProgressDialog(getActivity());  
 progressDialog.setMessage("Loading");  
 progressDialog.setCancelable(true);  
 progressDialog.setCanceledOnTouchOutside(true);  
 progressDialog.show();  
  
 webView.getSettings().setLoadsImagesAutomatically(true);  
 webView.getSettings().setAppCacheMaxSize(5 \* 1024 \* 1024);  
 webView.getSettings().setAppCachePath(getActivity().getCacheDir().getAbsolutePath());  
 webView.getSettings().setAllowFileAccess(true);  
 webView.getSettings().setAppCacheEnabled(true);  
 webView.getSettings().setCacheMode(WebSettings.LOAD\_DEFAULT);  
 webView.getSettings().setJavaScriptEnabled(true);  
 webView.setScrollBarStyle(View.SCROLLBARS\_INSIDE\_OVERLAY);  
 if (!isNetworkAvailable()) {  
 Toast.makeText(getActivity(), "Internet Required to update the scores", Toast.LENGTH\_SHORT).show();  
 webView.getSettings().setCacheMode(WebSettings.LOAD\_CACHE\_ELSE\_NETWORK);  
 }  
 webView.loadUrl(url);  
 return view;  
 }  
  
 private boolean isNetworkAvailable() {  
 ConnectivityManager connectivityManager = (ConnectivityManager) getActivity().getSystemService(Context.CONNECTIVITY\_SERVICE);  
 NetworkInfo activeNetworkInfo = connectivityManager.getActiveNetworkInfo();  
 return activeNetworkInfo != null && activeNetworkInfo.isConnected();  
 }  
  
 private class MyBrowser extends WebViewClient {  
  
 @Override  
 public boolean shouldOverrideUrlLoading(WebView view, String url) {  
 view.loadUrl(url);  
 return true;  
 }  
  
 @Override  
 public void onPageFinished(WebView view, String url) {  
 super.onPageFinished(view, url);  
 if (progressDialog.isShowing())  
 progressDialog.hide();  
 }  
 }  
}\*/*