```
//author(s): Chris Kovaleski, Luis Duque
package com.cyf.challengeyourfriends;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.UnsupportedEncodingException;
import java.util.List;

import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.client.methods.HttpGet;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.client.utils.URLEncodedUtils;
import org.apache.http.impl.client.DefaultHttpClient;
import org.json.JSONException;
import org.json.JSONObject;

import android.util.Log;

public class JSONParser {

    static InputStream is = null;
    static JSONObject jObj = null;
    static String json = "";

    // constructor
    public JSONParser() {

    }

    public static JSONObject getJSONFromUrl(String url) {
        // Making HTTP request
        try {
          // defaultHttpClient
          DefaultHttpClient httpClient = new DefaultHttpClient();
          HttpPost httpPost = new HttpPost(url);
          HttpResponse httpResponse = httpClient.execute(httpPost);
          HttpEntity httpEntity = httpResponse.getEntity();
          is = httpEntity.getContent();
        } catch (UnsupportedEncodingException e) {
          e.printStackTrace();
        } catch (ClientProtocolException e) {
          e.printStackTrace();
        } catch (IOException e) {
          e.printStackTrace();
        }
        try {
          BufferedReader reader = new BufferedReader(new InputStreamReader(
              is, "iso-8859-1"), 8);
          StringBuilder sb = new StringBuilder();
          String line = null;
```

```
      while ((line = reader.readLine()) != null) {
        sb.append(line + "n");
      }
      is.close();
      json = sb.toString();
    } catch (Exception e) {
      Log.e("Buffer Error", "Error converting result " + e.toString());
    }
    // try parse the string to a JSON object
    try {
      jObj = new JSONObject(json);
    } catch (JSONException e) {
      Log.e("JSON Parser", "Error parsing data " + json + e.toString());
    }
    // return JSON String
    return jObj;
  }



// function get json from url
// by making HTTP POST or GET mehtod
public static JSONObject makeHttpRequest(String url, String method,
    List<NameValuePair> params) {

    // Making HTTP request
    try {

        // check for request method
        if(method == "POST"){
            // request method is POST
            // defaultHttpClient
            DefaultHttpClient httpClient = new DefaultHttpClient();
            HttpPost httpPost = new HttpPost(url);
            httpPost.setEntity(new UrlEncodedFormEntity(params));

            HttpResponse httpResponse = httpClient.execute(httpPost);
            HttpEntity httpEntity = httpResponse.getEntity();
            is = httpEntity.getContent();

        }else if(method == "GET"){
            // request method is GET
            DefaultHttpClient httpClient = new DefaultHttpClient();
            String paramString = URLEncodedUtils.format(params, "utf-8");
            url += "?" + paramString;
            HttpGet httpGet = new HttpGet(url);

            HttpResponse httpResponse = httpClient.execute(httpGet);
            HttpEntity httpEntity = httpResponse.getEntity();
            is = httpEntity.getContent();
        }

    } catch (UnsupportedEncodingException e) {
        e.printStackTrace();
    } catch (ClientProtocolException e) {
        e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        }

        try {
            BufferedReader reader = new BufferedReader(new InputStreamReader(
                    is, "iso-8859-1"), 8);
            StringBuilder sb = new StringBuilder();
            String line = null;
            while ((line = reader.readLine()) != null) {
                sb.append(line + "\n");
            }
            is.close();
            json = sb.toString();
        } catch (Exception e) {
            Log.e("Buffer Error", "Error converting result " + e.toString());
        }

        // try parse the string to a JSON object
        try {
            jObj = new JSONObject(json);
        } catch (JSONException e) {
            Log.e("JSON Parser", "Error parsing data " + e.toString());
        }

        // return JSON String
        return jObj;

    }
}
```