

```

package com.example.elegestionfamillesmedandroid.ouils;

import android.os.AsyncTask;
import android.util.Log;

import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.HttpClient;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.message.BasicNameValuePair;
import org.apache.http.util.EntityUtils;

import java.io.IOException;
import java.io.UnsupportedEncodingException;
import java.util.ArrayList;

public class AccesHTTP extends AsyncTask<String, Integer, Long> {

    private ArrayList<NameValuePair> parametres;
    private String ret = null; // variable chaine pour le retour
    public AsyncResponse delegate = null;

    ArrayList<String> list;
    protected void onPreExecute() {
        super.onPreExecute();
        list = new ArrayList<>();
    }

    /**
     * Constructeur
     */
    public AccesHTTP() {
        parametres = new ArrayList<NameValuePair>();
    }

    /**
     * Ajout d'un paramètre post
     * @param nom
     * @param valeur
     */
    public void addParam(String nom, String valeur) {
        parametres.add(new BasicNameValuePair(nom, valeur));
    }

    /**
     * connexion eb tache de fond dans un thread séparé
     * @param strings
     * @return
     */
    @Override
    protected Long doInBackground(String... strings) {
        HttpClient cnxHttp = new DefaultHttpClient();
        HttpPost parmaCnx = new HttpPost(strings[0]);

        try {
            //encodage des paramètres
            parmaCnx.setEntity(new UrlEncodedFormEntity(parametres));
            // connexion et envoie de paramètres, attente de réponse

```

```

        HttpResponse response = cnxHttp.execute(parmaCnx);
        // transformation de la réponse
        ret = EntityUtils.toString(response.getEntity());
    } catch (UnsupportedEncodingException e) {
        Log.d("Erreur encodage ", "*****" + e.toString());
    } catch (ClientProtocolException e) {
        Log.d("Erreur protocole ", "*****" +
e.toString());
    } catch (IOException e) {
        Log.d("Erreur I/O ", "*****" + e.toString());
    }
    return null;
}

@Override
protected void onPostExecute(Long result) {
    delegate.processFinish(ret.toString());
}
}

```