```java
package com.example.weatherapp;

import android.content.Context;
import android.content.pm.ApplicationInfo;
import android.content.pm.PackageManager;
import android.location.Location;
import android.os.AsyncTask;
import android.util.Log;

import androidx.appcompat.app.AppCompatActivity;
import androidx.fragment.app.FragmentTransaction;

import com.google.gson.Gson;

import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;
import java.lang.ref.WeakReference;
import java.net.HttpRetryException;
import java.net.HttpURLConnection;
import java.net.URL;
import java.util.ArrayList;
import java.util.Calendar;

public class WeatherFetch extends AsyncTask<Void,Void, ArrayList<WeatherInformation>>
{
    private WeakReference<Context> mainContext;
    private Location lastLocation;

    WeatherFetch(Context mainContext, Location lastLocation){
        this.mainContext = new WeakReference<>(mainContext);
        this.lastLocation = lastLocation;
    }

    @Override
    protected void onPostExecute(ArrayList<WeatherInformation> weatherInformations) {
        super.onPostExecute(weatherInformations);
        if(mainContext.get() != null){
            AppCompatActivity activity = (AppCompatActivity)mainContext.get();

            FragmentTransaction fragmentTransaction = activity.
            getSupportFragmentManager().beginTransaction();
            fragmentTransaction.replace(R.id.frameLayoutMain, new StartingFragment(
            weatherInformations,mainContext));
            fragmentTransaction.commit();
        }

    }

    @Override
    protected ArrayList<WeatherInformation> doInBackground(Void... voids) {
        try {
            String fetchUrl;
            ApplicationInfo applicationInfo = mainContext.get().getApplicationContext
            ().getPackageManager().getApplicationInfo(mainContext.get().getPackageName
            (),PackageManager.GET_META_DATA);
            String openWeatherAPIKEY = String.valueOf(applicationInfo.metaData.get(
            "WEATHER_API_KEY"));
            if(lastLocation == null){
                fetchUrl = String.format(
                "https://api.openweathermap.org/data/2.5/forecast?q=Germany&appid=%s&u
                nits=metric",openWeatherAPIKEY);
            }else{
                fetchUrl = String.format(
                "https://api.openweathermap.org/data/2.5/forecast?lat=%s&lon=%s&appid=
                %s&units=metric",lastLocation.getLatitude(),lastLocation.getLongitude
                (),openWeatherAPIKEY);
            }

            Log.d("fetchURL",fetchUrl);
            URL url = new URL(fetchUrl);
            HttpURLConnection httpURLConnection = (HttpURLConnection) url.
            openConnection();
```

```
62              httpURLConnection.setRequestMethod("GET");
63              httpURLConnection.setRequestProperty("Content-Type","application/json");
64              httpURLConnection.connect();
65              int responseCode = httpURLConnection.getResponseCode();
66              if(responseCode == HttpURLConnection.HTTP_OK){
67                  BufferedReader bufferedReader = new BufferedReader(new
                    InputStreamReader(httpURLConnection.getInputStream()));
68                  String inputLine;
69                  StringBuilder response = new StringBuilder();
70                  while ((inputLine = bufferedReader.readLine()) != null){
71                      response.append(inputLine);
72                  }
73                  bufferedReader.close();
74                  httpURLConnection.disconnect();
75
76                  Gson gson = new Gson();
77                  CustomResponse customResponse = gson.fromJson(response.toString(),
                    CustomResponse.class);
78
79                  ArrayList<WeatherInformation> weatherInformations= new ArrayList<>();
80
81                  for(WeatherResponse listItem : customResponse.list){
82                      Calendar calendar = Calendar.getInstance();
83                      calendar.setTimeInMillis(listItem.dt * 1000L);
84
85                      WeatherInformation weatherInformation = new WeatherInformation(
86                              listItem.clouds.all,
87                              listItem.wind.speed,
88                              listItem.main.humidity,
89                              listItem.main.temp_max,
90                              listItem.main.temp_min,
91                              listItem.weather[0].icon,
92                              calendar,
93                              listItem.weather[0].main);
94
95                      weatherInformations.add(weatherInformation);
96                  }
97
98                  return weatherInformations;
99              }
100             else if (responseCode >= 400 && responseCode < 500){
101                 throw new HttpRetryException("Client error",responseCode);
102             }else if(responseCode >= 500 && responseCode < 600){
103                 throw new HttpRetryException("Server error",responseCode);
104             }
105
106         } catch (IOException e) {
107             throw new RuntimeException(e);
108         } catch (PackageManager.NameNotFoundException e) {
109             throw new RuntimeException(e);
110         }
111         return null;
112     }
113 }
114
```