**Deprem MainActivity- AsyncTask ile Network işlemleri**

public class MainActivity extends AppCompatActivity {  
  
 private ListView listView;  
 private ArrayList<Deprem> depremList;  
 private ArrayAdapter<Deprem> adaptor;  
  
 private String jsonCevabi;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
 listView = findViewById(R.id.listView);  
  
 // jsonCevabi = dosyadanOku("deprem.json");  
 String urlString = "https://earthquake.usgs.gov/fdsnws/event/1/query?format=geojson&starttime=2020-01-01&endtime=2020-03-12&minmagnitude=4&minlatitude=36&maxlatitude=42&minlongitude=26&maxlongitude=45";  
 Indirme indirme = new Indirme();  
 indirme.execute(urlString);  
  
 }  
  
 class Indirme extends AsyncTask<String, Void, String> {  
  
 @Override  
 protected String doInBackground(String... urlString) {  
 jsonCevabi = jsonCevabiAl(urlString[0]);//NETWORK ON MAIN THREAD EXCEPTION--> BACKGROUND THREAD'DE YAPILACAK  
 return jsonCevabi;  
 }  
  
  
 @Override  
 protected void onPostExecute(String s) { //MAIN THREAD'DE ARAYÜZE MÜDAHELE EDEBİLİRİZ.  
 depremList = jsonAyikla(jsonCevabi);  
 adaptor = new ArrayAdapter<>(getApplicationContext(), android.R.layout.simple\_list\_item\_1, depremList);  
 listView.setAdapter(adaptor);  
 }  
  
 private String jsonCevabiAl(String urlString) {  
 StringBuilder sb=new StringBuilder();  
 InputStream is = null;  
  
 HttpURLConnection urlConnection=null;  
  
  
 try {  
 URL url=new URL(urlString);  
 urlConnection= (HttpURLConnection) url.openConnection();  
 urlConnection.setRequestMethod("GET");  
 if(urlConnection.getResponseCode()==200){  
 is=urlConnection.getInputStream();  
 }  
  
 //is = getAssets().open(dosyaAdi);//DOSYADAN VERİ AKIŞI (STREAM) BAŞLATMA  
  
 InputStreamReader isr = new InputStreamReader(is);  
 BufferedReader br = new BufferedReader(isr);  
 String satir = br.readLine();  
 while (satir != null) {  
 sb.append(satir);  
 satir = br.readLine();  
 }  
  
  
 } catch (IOException e) {  
 e.printStackTrace();  
 } finally {  
 if (is != null) {  
 try {  
 is.close();  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 if(urlConnection!=null)  
 urlConnection.disconnect();  
  
 }  
 return sb.toString();  
 }  
  
  
  
 }  
  
  
 private ArrayList<Deprem> jsonAyikla(String jsonCevabi) {  
 ArrayList<Deprem> liste = new ArrayList<Deprem>();  
  
 try {  
 JSONObject kokJO = new JSONObject(jsonCevabi);  
 JSONArray featuresJA = kokJO.getJSONArray("features");  
 for (int i = 0; i < featuresJA.length(); i++) {  
 JSONObject featureJO = featuresJA.getJSONObject(i);  
 JSONObject propertiesJO = featureJO.getJSONObject("properties");  
 double siddet = propertiesJO.getDouble("mag");  
 String konum = propertiesJO.getString("place");  
 long tarih = propertiesJO.getLong("time");  
 Deprem deprem = new Deprem(tarih, siddet, konum);  
 liste.add(deprem);  
 }  
 } catch (JSONException e) {  
 e.printStackTrace();  
 }  
  
  
 return liste;  
 }  
  
 private String dosyadanOku(String dosyaAdi) {  
  
 StringBuilder sb = new StringBuilder();  
 InputStream is = null;  
 try {  
 is = getAssets().open(dosyaAdi);  
 InputStreamReader isr = new InputStreamReader(is);  
 BufferedReader br = new BufferedReader(isr);  
 String satir = br.readLine();  
  
 while (satir != null) {  
 sb.append(satir);  
 satir = br.readLine();  
 }  
 } catch (IOException e) {  
 e.printStackTrace();  
 } finally {  
 if (is != null) {  
 try {  
 is.close();  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
 }  
 }  
  
 return sb.toString();  
 }  
  
  
}