

```
package com.example.covid_19alertapp.extras;
```

```
import android.app.IntentService;
```

```
import android.content.Intent;
```

```
import android.location.Address;
```

```
import android.location.Geocoder;
```

```
import android.location.Location;
```

```
import android.os.Bundle;
```

```
import android.os.ResultReceiver;
```

```
import android.text.TextUtils;
```

```
import android.util.Log;
```

```
import androidx.annotation.Nullable;
```

```
import java.io.IOException;
```

```
import java.util.ArrayList;
```

```
import java.util.List;
```

```
import java.util.Locale;
```

```
public class FetchAddress extends IntentService {
```

```
    private static final String GEO_LOCATION = "geo_location";
```

```
    private static final String GEO_ADDRESS = "geo_address";
```

```
    private static final String GEO_RECEIVER = "geo_receiver";
```

```
    private static final String LIST_POSITION = "position@list";
```

```
private static final int GEO_FAILURE = 103;
```

```
private static final int GEO_SUCCESS = 104;
```

```
protected ResultReceiver receiver;
```

```
/**
```

```
 * fetch address from co-ordinates
```

```
 */
```

```
public FetchAddress() {
```

```
    super(FetchAddress.class.getName());
```

```
}
```

```
@Override
```

```
protected void onHandleIntent(@Nullable Intent intent) {
```

```
    /*
```

```
    receive location inside 'intent'
```

```
    decode the address
```

```
    */
```

```
Log.d(LogTags.Address_TAG, "onHandleIntent: inside FetchAddress class");
```

```
Geocoder geocoder = new Geocoder(this, Locale.getDefault());
```

```
if (intent == null) {  
    return;  
}  
  
String errorMessage = "";  
  
// get the location and receiver passed to this service through an extra.  
Location location = intent.getParcelableExtra(GEO_LOCATION);  
  
// get the receiver from calling activity  
receiver = intent.getParcelableExtra(GEO_RECEIVER);  
  
// get the position of list  
int listPosition = intent.getIntExtra(LIST_POSITION, -1);  
  
List<Address> addresses = null;  
  
try {  
  
    Log.d(LogTags.Address_TAG,  
        "onHandleIntent: latlong - "+location.getLatitude()+" "+location.getLongitude()  
    );  
  
    addresses = geocoder.getFromLocation(  
        location.getLatitude(),  
        location.getLongitude(),
```

```

        // get just a single address.

        1);

    } catch (IOException ioException) {

        // Catch network or other I/O problems.

        errorMessage = "service not available";

        Log.d(LogTags.Address_TAG, errorMessage, ioException);

    } catch (IllegalArgumentException illegalArgumentException) {

        // Catch invalid latitude or longitude values.

        errorMessage = "invalid lat_long used";

        Log.d(LogTags.Address_TAG, errorMessage + ". " +

            "Latitude = " + location.getLatitude() +

            ", Longitude = " +

            location.getLongitude(), illegalArgumentException);

    }

    // Handle case where no address was found.

    if (addresses == null || addresses.size() == 0) {

        if (!errorMessage.isEmpty()) {

            errorMessage = "address not found";

            Log.d(LogTags.Address_TAG, errorMessage);

        }

        deliverResultToReceiver(GEO_FAILURE, errorMessage, listPosition);

```

```
}
```

```
else {
```

```
    Log.d(LogTags.Address_TAG, "address found");
```

```
    Address address = addresses.get(0);
```

```
    List<String> addressFragments = new ArrayList<String>();
```

```
    // Fetch the address lines using getAddressLine,
```

```
    // join them, and send them to the UI thread.
```

```
    for(int i = 0; i <= address.getMaxAddressLineIndex(); i++) {
```

```
        addressFragments.add(address.getAddressLine(i));
```

```
    }
```

```
    // fix too long addresses
```

```
    String senAddress = "";
```

```
    if(addressFragments.size() >= 5)
```

```
        senAddress = addressFragments.get(addressFragments.size()-3) + ", "
```

```
        + addressFragments.get(addressFragments.size()-1) + ", "
```

```
        + addressFragments.get(addressFragments.size()-1);
```

```
    else
```

```
        senAddress = TextUtils.join(", ", addressFragments);
```

```

        deliverResultToReceiver(GEO_SUCCESS,
                                senAddress,
                                listPosition);
    }

}

private void deliverResultToReceiver(int resultCode, String address, int position) {

    Bundle bundle = new Bundle();
    bundle.putString(GEO_ADDRESS, address);
    bundle.putInt(LIST_POSITION, position);

    receiver.send(resultCode, bundle);

}

public static int getGeoFailure() {
    return GEO_FAILURE;
}

public static int getGeoSuccess() {
    return GEO_SUCCESS;
}

```

}