

```
package com.example.covid_19alertapp.adapters;
```

```
import android.annotation.SuppressLint;
```

```
import android.app.AlertDialog;
```

```
import android.content.Context;
```

```
import android.content.DialogInterface;
```

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

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

```
import android.view.LayoutInflater;
```

```
import android.view.View;
```

```
import android.view.ViewGroup;
```

```
import android.widget.LinearLayout;
```

```
import android.widget.TextView;
```

```
import android.widget.Toast;
```

```
import androidx.annotation.NonNull;
```

```
import androidx.recyclerview.widget.RecyclerView;
```

```
import com.example.covid_19alertapp.R;
```

```
import com.example.covid_19alertapp.activities.MatchedLocationsMapsActivity;
```

```
import com.example.covid_19alertapp.extras.LogTags;
```

```
import com.example.covid_19alertapp.models.MatchedLocation;
```

```
import com.example.covid_19alertapp.roomdatabase.VisitedLocationsDao;
```

```
import com.example.covid_19alertapp.roomdatabase.VisitedLocationsDatabase;
```

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

```
public class LocationListAdapter extends RecyclerView.Adapter<LocationListAdapter.MyViewHolder> {
```

```

Context context;

ArrayList<MatchedLocation> locationsList;

public LocationListAdapter(Context context,ArrayList<MatchedLocation> locationsList)
{
    this.context = context;
    this.locationsList = locationsList;
}

@NonNull
@Override
public LocationListAdapter.MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int
viewType) {
    return new MyViewHolder(LayoutInflater.from(context).inflate(R.layout.locationlist_view, parent,
false));
}

@SuppressLint("SetTextI18n")
@Override
public void onBindViewHolder(@NonNull LocationListAdapter.MyViewHolder holder, final int position)
{
    holder.location.setText(locationsList.get(position).getAddress());
    holder.dateTime.setText(locationsList.get(position).getMeaningfulDateTime());
    holder.verifiedCount.setText("Infected (Verified):
"+locationsList.get(position).getVerifiedCount()+"");
    holder.unverifiedCount.setText("Infected (Unverified) :
"+locationsList.get(position).getUnverifiedCount()+"");

    holder.location.setOnClickListener(new View.OnClickListener() {
        @Override

```

```

public void onClick(View v) {

    //SHOW IN MAP FUNCTION

    Intent intent = new Intent(context, MatchedLocationsMapsActivity.class);

    intent.putExtra(MatchedLocationsMapsActivity.getBILatitudeKey(),
locationsList.get(position).getBILatitude());

    intent.putExtra(MatchedLocationsMapsActivity.getBILongitudeKey(),
locationsList.get(position).getBILongitude());

    context.startActivity(intent);

}

});

```

```

holder.linearLayout.setOnLongClickListener(new View.OnLongClickListener() {

    @Override

    public boolean onLongClick(View v) {

        AlertDialog Dialog = new AlertDialog.Builder(context)

            .setTitle("Delete")

            .setMessage("Do you want to remove this record?")

            .setPositiveButton("Ok", new DialogInterface.OnClickListener() {

                @Override

                public void onClick(DialogInterface dialog, int which) {

                    // delete location from local db

                    if(locationsList.get(position).getRawDateTime().equals(""))

                        || locationsList.get(position).getLatLon().equals("")) {

                        Toast.makeText(context, "Can't delete nearby home entries", Toast.LENGTH_LONG)

                            .show();

                        return;

```

```
}
```

```
final String deletionPrimaryKey =
```

```
locationsList.get(position).getLatLon()+"_"+locationsList.get(position).getRawDateTime();
```

```
VisitedLocationsDatabase roomDatabase =
```

```
VisitedLocationsDatabase.getDatabase(context);
```

```
final VisitedLocationsDao visitedLocationsDao = roomDatabase.visitedLocationsDao();
```

```
roomDatabase.databaseWriteExecutor.execute(new Runnable() {
```

```
    @Override
```

```
    public void run() {
```

```
        visitedLocationsDao.deleteByPrimaryKey(deletionPrimaryKey);
```

```
        Log.d(LogTags.MatchFound_TAG, "run: deleted entry = "+deletionPrimaryKey);
```

```
    }
```

```
});
```

```
locationsList.remove(position);
```

```
LocationListAdapter.this.notifyItemRemoved(position);
```

```
dialog.dismiss();
```

```
Toast.makeText(context, "Record Deleted!", Toast.LENGTH_SHORT).show();
```

```
}
```

```
}}
```

```
.setNegativeButton("Cancel", new DialogInterface.OnClickListener() {
```

```
    @Override
```

```

        public void onClick(DialogInterface dialog, int which) {
            dialog.dismiss();
        }
    })
    .show();

    return false; //DO NOT REMOVE THIS
}
});
}

@Override
public int getItemCount() {
    return locationsList.size();
}

public class MyViewHolder extends RecyclerView.ViewHolder {
    TextView unverifiedCount,verifiedCount,dateTime,location;
    LinearLayout linearLayout;
    public MyViewHolder(@NonNull View itemView) {
        super(itemView);
        unverifiedCount = itemView.findViewById(R.id.locationList_unverifiedCounter);
        verifiedCount = itemView.findViewById(R.id.locationList_verifiedCounter);
        dateTime = itemView.findViewById(R.id.locationList_dateText);
        location = itemView.findViewById(R.id.locationList_locationText);
        linearLayout = itemView.findViewById(R.id.locationList_linearLayout);
    }
}

```

