

```

package com.example.covid_19alertapp.roomdatabase;

import android.content.Context;

import androidx.room.Database;
import androidx.room.Room;
import androidx.room.RoomDatabase;

import java.util.concurrent.ExecutorService;
import java.util.concurrent.Executors;

@Database(entities = {VisitedLocations.class}, version = 1, exportSchema = false)
public abstract class VisitedLocationsDatabase extends RoomDatabase {

    public abstract VisitedLocationsDao visitedLocationsDao();

    private static volatile VisitedLocationsDatabase INSTANCE;

    // room queries need to be done on separate threads
    private static final int NUMBER_OF_THREADS = 3;
    public static final ExecutorService databaseWriteExecutor =
        Executors.newFixedThreadPool(NUMBER_OF_THREADS);

    public static VisitedLocationsDatabase getDatabase(final Context context) {
        /*
        thread safe singleton design to get database INSTANCE
        */

```

```
if (INSTANCE == null) {  
    synchronized (VisitedLocations.class) {  
        if (INSTANCE == null) {  
            INSTANCE = Room.databaseBuilder(context.getApplicationContext(),  
                VisitedLocationsDatabase.class, "word_database")  
                .build();  
        }  
    }  
}  
return INSTANCE;  
}  
  
}
```