```
package com.example.sairamkrishna.myapplication;

import android.Manifest;
import android.app.Activity;
import android.bluetooth.BluetoothAdapter;
import android.bluetooth.BluetoothDevice;

import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.pm.PackageManager;
import android.location.Criteria;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.os.Environment;
import android.support.v4.app.ActivityCompat;
import android.view.View;

import android.widget.ArrayAdapter;
import android.widget.Button;

import android.widget.ListView;
import android.widget.Toast;

import java.io.OutputStreamWriter;
import java.sql.Time;
import java.util.ArrayList;
import java.util.HashSet;
import java.util.Set;
import java.util.Calendar;


public class MainActivity extends Activity {
    Button bScan, bList, bOn;
    ListView lview;
    private BluetoothAdapter btAdapter;
    private HashSet<String> storedDeviceNames;
    double latitude, longitude;

    private class DataPoint {
        int time;
        String name;
        double latitude, longitude;

        public DataPoint(int t, double lat, double lon, String n) {
            time = t;
            name = n;
            latitude = lat;
            longitude = lon;
        }

        public String getOutput() {
            return String.valueOf(time) + ", " + String.format("%.2f", latitude) +
", "
                    + String.format("%.2f", longitude) + ", " + name;
        }
    }

    private ArrayList<DataPoint> datapoints;
    private LocationManager locationManager;
    private LocationListener listener;


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        bScan = (Button) findViewById(R.id.bscan);
        bList = (Button) findViewById(R.id.blist);
        bOn = (Button) findViewById(R.id.bon);

        btAdapter = BluetoothAdapter.getDefaultAdapter();
        lview = (ListView) findViewById(R.id.lv);
        datapoints = new ArrayList<DataPoint>();
        addNames();
        latitude = 0;
        longitude = 0;
        listener = new LocationListener() {
            @Override
            public void onLocationChanged(Location location) {
                latitude = location.getLatitude();
                longitude = location.getLongitude();
            }

            @Override
            public void onStatusChanged(String provider, int status, Bundle extras)
{
            }

            @Override
            public void onProviderEnabled(String provider) {
            }
            @Override
            public void onProviderDisabled(String provider) {
            }
        };

        locationManager = (LocationManager)
this.getSystemService(Context.LOCATION_SERVICE);
        if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED &&
ActivityCompat.checkSelfPermission(this, Manifest.permission.ACCESS_COARSE_LOCATION)
!= PackageManager.PERMISSION_GRANTED) {
            // TODO: Consider calling
            //    ActivityCompat#requestPermissions
            // here to request the missing permissions, and then overriding
            //   public void onRequestPermissionsResult(int requestCode, String[]
permissions,
            //                                          int[] grantResults)
            // to handle the case where the user grants the permission. See the
documentation
            // for ActivityCompat#requestPermissions for more details.
            return;
        }
        locationManager.requestLocationUpdates(LocationManager.NETWORK_PROVIDER, 0,
0, listener);
        //saveData();


    }

    public void saveData() {
        String state = Environment.getExternalStorageState();
        if( Environment.MEDIA_MOUNTED.equals(state) ) {
                  }
    }

    private final BroadcastReceiver mReceiver = new BroadcastReceiver() {
        public void onReceive(Context context, Intent intent) {
            String action = intent.getAction();
            if (BluetoothAdapter.ACTION_DISCOVERY_STARTED.equals(action)) {
            } else if (BluetoothAdapter.ACTION_DISCOVERY_FINISHED.equals(action)) {
                btAdapter.startDiscovery();
            } else if (BluetoothDevice.ACTION_FOUND.equals(action)) {
                BluetoothDevice device = (BluetoothDevice)
intent.getParcelableExtra(BluetoothDevice.EXTRA_DEVICE);
                if( storedDeviceNames.contains( device.getName() ) ) {
                    int seconds = Calendar.getInstance().get( Calendar.SECOND );
                    datapoints.add( new DataPoint(  seconds, latitude, longitude,
device.getName() ) );
                    //if( datapoints.size() > 100 ) {
                    //    saveData();
                    //}
                }
            }
        }
    };

    public void startScan( View v ) {
        if( !btAdapter.isEnabled() ) {
            Toast.makeText(getApplicationContext(), "Bluetooth is off...",
Toast.LENGTH_LONG).show();
        } else {
            Toast.makeText(getApplicationContext(),
"Scanning...",Toast.LENGTH_SHORT).show();
            IntentFilter filter = new IntentFilter();
            filter.addAction(BluetoothDevice.ACTION_FOUND);
            filter.addAction(BluetoothAdapter.ACTION_DISCOVERY_STARTED);
            filter.addAction(BluetoothAdapter.ACTION_DISCOVERY_FINISHED);
            registerReceiver(mReceiver, filter);
            int MY_PERMISSIONS_REQUEST_ACCESS_COARSE_LOCATION = 1;
            ActivityCompat.requestPermissions(this,
                    new String[]{Manifest.permission.ACCESS_COARSE_LOCATION},
                    MY_PERMISSIONS_REQUEST_ACCESS_COARSE_LOCATION);
            btAdapter.startDiscovery();
        }
    }
    public void stopScan( View v ) {
        if (!btAdapter.isEnabled()) {
            Toast.makeText(getApplicationContext(), "Bluetooth is off...",
Toast.LENGTH_LONG).show();
        } else {
            btAdapter.cancelDiscovery();
        }
    }

    public void addNames() {
        storedDeviceNames = new HashSet<String>();
        storedDeviceNames.add( "HC-05" );
        storedDeviceNames.add( "H-C-2010-06-01" );
        storedDeviceNames.add( "QCAMP13" );
        storedDeviceNames.add( "QCAMP5" );
        storedDeviceNames.add( "Qualcomm3" );
    }

    public void turnOnBT( View v ){
        if( btAdapter != null ) {
            if( !btAdapter.isEnabled() ) {
                Intent turnOn = new Intent(BluetoothAdapter.ACTION_REQUEST_ENABLE);
                startActivityForResult( turnOn, 0 );
                Toast.makeText(getApplicationContext(), "Turned on",
Toast.LENGTH_LONG).show();
            } else {
                Toast.makeText(getApplicationContext(), "Already on",
Toast.LENGTH_LONG).show();
            }
        }
    }

    public void turnOffBT( View v ) {
        btAdapter.disable();
        Toast.makeText( getApplicationContext(), "Turned off" ,Toast.LENGTH_LONG
).show();
    }


    public void list( View v ){

 //       Toast.makeText(getApplicationContext(), "Showing Paired
Devices",Toast.LENGTH_SHORT).show();

        ArrayList< String > list = new ArrayList< String >();
        for( int i = Math.max( 0, datapoints.size() - 10 ); i < datapoints.size();
i++ ) {

           // list.add( String.valueOf( dp.time ) + " " +  dp.name );
            list.add(  datapoints.get(i).getOutput() );
        }

        final ArrayAdapter adapter = new ArrayAdapter<String>( this,
android.R.layout.simple_list_item_1, list );


        lview.setAdapter( adapter );
    }

    public void onDestroy() {
        unregisterReceiver( mReceiver );
        //saveData();
        super.onDestroy();
    }
}
```