**< MainActivity.java >**

**public class MainActivity extends AppCompatActivity { // 원격 장치에 대한 Bluetooth 연결을 생성하는 코드**

**@Override**

**protected void onCreate(Bundle savedInstanceState) // AppCompatActivity에 있는onCreate() 호출**

**{**

**handler = new Handler(Looper.getMainLooper()) { // GUI 핸들러**

**@Override**

**public void handleMessage(Message msg) // 핸들러를 통해 실행결과를 Message로 전달**

**{**

**}**

**buttonConnect.setOnClickListener(new View.OnClickListener() // 블루투스 장치 선택**

**{**

**}**

**buttonToggle.setOnClickListener(new View.OnClickListener() // ON/OFF 버튼에 따른 아두이노 cmd 메시지 전송**

**{**

**}**

**}**

**public static class CreateConnectThread extends Thread // 블루투스 연결 생성 스레드**

**{**

**public CreateConnectThread(BluetoothAdapter bluetoothAdapter, String address) {**

**}**

**public void run() // 소켓을 통해 원격 장치에 연결한다. 이 호출은 성공or예외 발생할 때까지 차단됨.**

**{**

**}**

**public void cancel() // 클라이언트 소켓을 닫고 스레드를 완료한다.**

**{**

**}**

**}**

**public static class ConnectedThread extends Thread // 데이터 전송을 위한 스레드**

**{**

**public ConnectedThread(BluetoothSocket socket) // 블루투스 통신은 Socket을 이용해서 이루어진다.**

**소켓 통신은 InputStream, OutputStream 객체를 사용하여 데이터를 송수신한다.**

**{**

**}**

**public void run() // 종료 문자에 도달 할 때까지 Arduino의 InputStream에서 읽습니다.**

**그런 다음 전체 문자열 메시지를 GUI 처리기에 보냅니다.**

**{**

**}**

**public void write(String input) // 데이터를 읽어서 원격장치로 보낸다**

**{**

**}**

**public void cancel() // 클라이언트 소켓을 닫고 스레드 완료**

**{**

**}**

**}**

**@Override**

**public void onBackPressed() // 뒤로가기 버튼을 누르면 연결을 종료하고 앱을 닫는다**

**{**

**}**

**}**

**< DeviceListAdapter.java > // 연결할 수 있도록 페어린 된 Bluetooth 장치 목록을 표시하는 클래스이다.**

**목록인 SelectDeviceActivity에 표시된다.**

**public class DeviceListAdapter extends RecyclerView.Adapter<RecyclerView.ViewHolder>**

**{**

**public static class ViewHolder extends RecyclerView.ViewHolder**

**{**

**public ViewHolder(View v)**

**{**

**}**

**}**

**public DeviceListAdapter(Context context, List<Object> deviceList)**

**{**

**}**

**public RecyclerView.ViewHolder onCreateViewHolder(ViewGroup parent, int viewType)**

**{**

**}**

**@Override**

**public void onBindViewHolder(final RecyclerView.ViewHolder holder, final int position)**

**{**

**}**

**@Override**

**public int getItemCount()**

**{**

**}**

**}**

**< SelectDeviceActivity.java >**

**public class SelectDeviceActivity extends AppCompatActivity {**

**@Override**

**protected void onCreate(Bundle savedInstanceState)**

**{**

**}**

**}**

**< DeviceInfoModel.java > // 원격 장치 정보에 대한 자리 표시자 역할을 하는 클래스**

**public class DeviceInfoModel**

**{**

**public DeviceInfoModel() // 메소드가 비어있음**

**{**

**}**

**public DeviceInfoModel(String deviceName, String deviceHardwareAddress) // 장치 이름, MAC주소 표시**

**{**

**}**

**public String getDeviceName() // 장치이름 얻기**

**{**

**}**

**public String getDeviceHardwareAddress() // MAC 주소 얻기**

**{**

**}**

**}**