Some variables init

```
initialize global UUID_Service to " 49535343-fe7d-4ae5-8fa9-9fafd205e455 "
initialize global UUID_TX to " 49535343-8841-43f4-a8d4-ecbe34729bb3 "
initialize global UUID_Notification to " 49535343-1e4d-4bd9-ba61-23c647249616 "
initialize global DeviceAddress to 0
initialize global timerInterval to 0
initialize global IsConnected to false
initialize global EnableRx to false
initialize global FirstReadDone to false
```

```
initialize global redState to 0
initialize global greenState to 0
initialize global blueState to 0
```
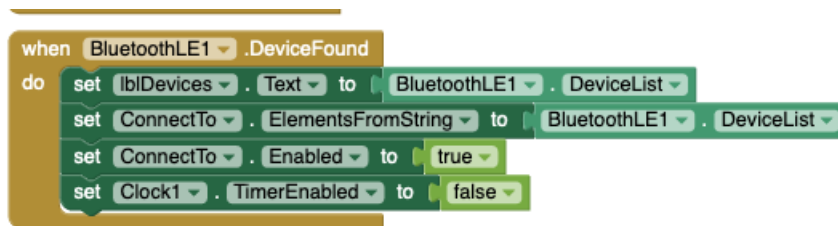
Start scanning

```
when btnBLEStartScan .Click
do  set lblDevices . Text to " Searching... "
    call BluetoothLE1 .StartScanning
    set Clock1 . TimerInterval to 1000
    set Clock1 . TimerEnabled to true
    set global timerInterval to 0
```

Stop scanning

```
when btnCancelScan .Click
do  set lblDevices . Text to " Idle... "
    set ConnectTo . Enabled to false
    call BluetoothLE1 .StopScanning
    set Clock1 . TimerEnabled to false
```
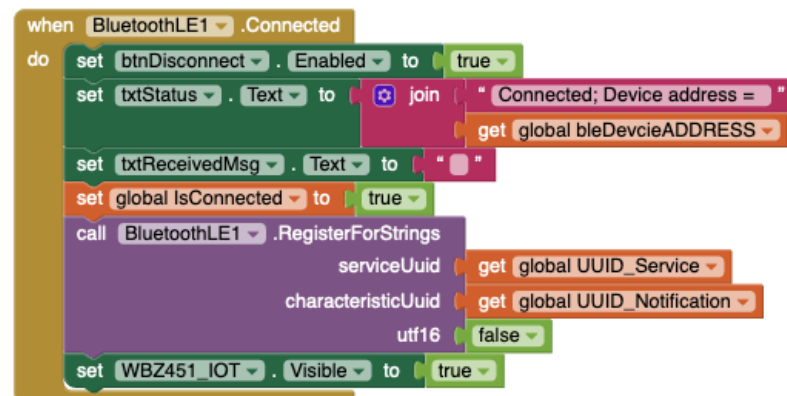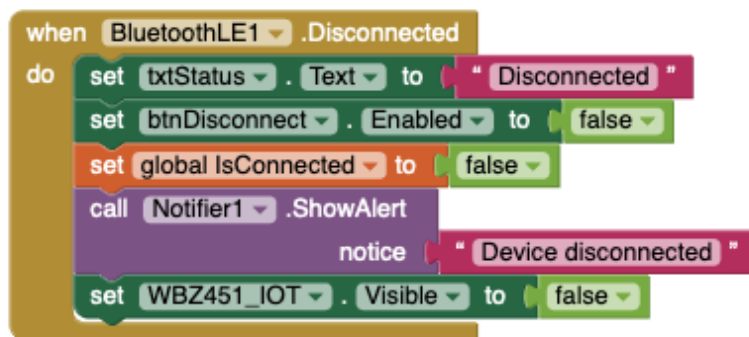
Device found

```
when  BluetoothLE1 ▼ .DeviceFound
do    set  lblDevices ▼ . Text ▼ to    BluetoothLE1 ▼ . DeviceList ▼
      set  ConnectTo ▼ . ElementsFromString ▼ to    BluetoothLE1 ▼ . DeviceList ▼
      set  ConnectTo ▼ . Enabled ▼ to    true ▼
      set  Clock1 ▼ . TimerEnabled ▼ to    false ▼
```

Button "Connect to…" is a list picker

```
when  ConnectTo ▼ .AfterPicking
do    set  global bleDevcieADDRESS ▼ to    select list item  list    split at spaces    ConnectTo ▼ . Selection ▼
                                                             index   1
      set  global bleDevcieName ▼ to    select list item  list    split at spaces    ConnectTo ▼ . Selection ▼
                                                          index   2
      set  lblDevices ▼ . Text ▼ to    ⚙ join    " Connecting to device "
                                                  get  global bleDevcieADDRESS ▼
      call  BluetoothLE1 ▼ .ConnectWithAddress
                                address    get  global bleDevcieADDRESS ▼
      set  ConnectTo ▼ . Enabled ▼ to    false ▼
      call  BluetoothLE1 ▼ .StopScanning
      set  lblDevices ▼ . Text ▼ to    " Idle... "
```

Event Connected, Register for Notification

```
when  BluetoothLE1 ▼ .Connected
do    set  btnDisconnect ▼ . Enabled ▼ to    true ▼
      set  txtStatus ▼ . Text ▼ to    ⚙ join    " Connected; Device address = "
                                                 get  global bleDevcieADDRESS ▼
      set  txtReceivedMsg ▼ . Text ▼ to    " ● "
      set  global IsConnected ▼ to    true ▼
      call  BluetoothLE1 ▼ .RegisterForStrings
                               serviceUuid    get  global UUID_Service ▼
                        characteristicUuid    get  global UUID_Notification ▼
                                     utf16    false ▼
      set  WBZ451_IOT ▼ . Visible ▼ to    true ▼
```

Event disconnected

```
when  BluetoothLE1 ▾  .Disconnected
do    set  txtStatus ▾ . Text ▾  to  "  Disconnected  "
      set  btnDisconnect ▾ . Enabled ▾  to    false ▾
      set  global IsConnected ▾  to    false ▾
      call  Notifier1 ▾ .ShowAlert
                            notice    "  Device disconnected  "
      set  WBZ451_IOT ▾ . Visible ▾  to    false ▾
```

Receiving from WBZ451

Packet received are encapsulated in (message) need to removed them
Filter on rcv message "+RGB: R g B". Upper case is ON, lower case is OFF.
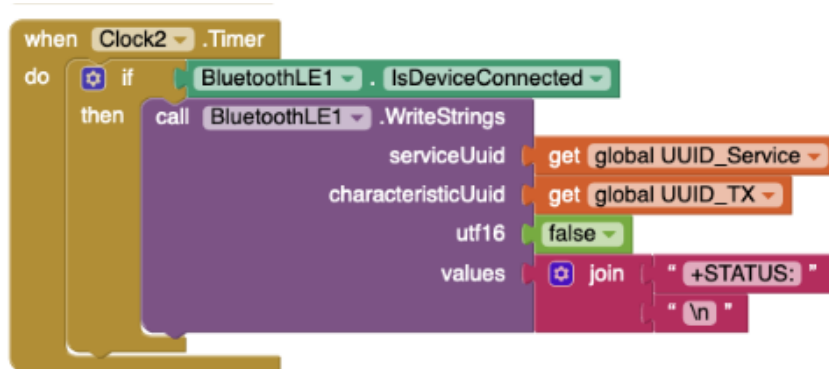Set appropriate variable , and change ledButton background.
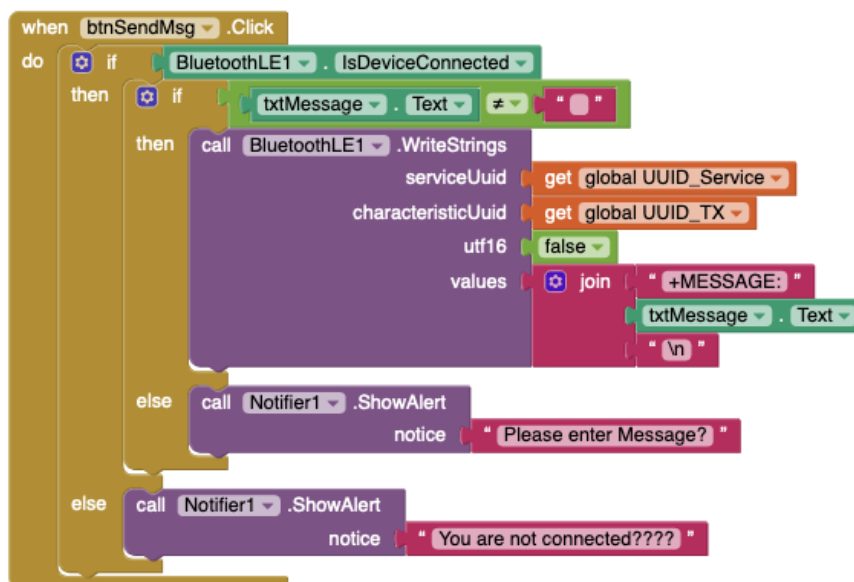Filter on rcv message "+TEMP:"
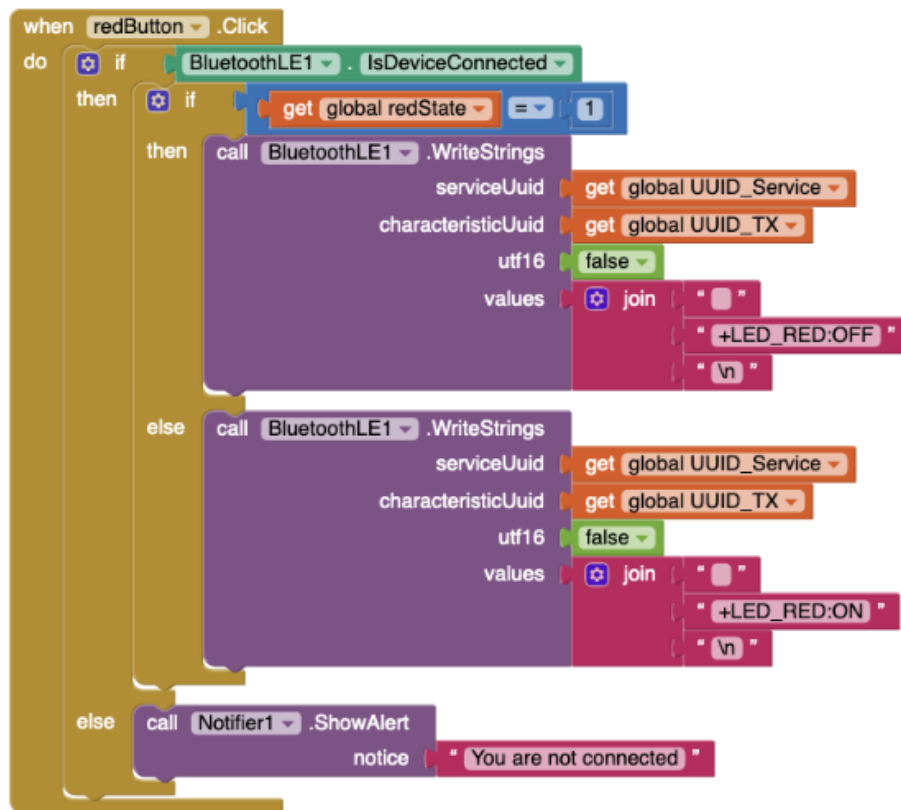If NONE above, just display in txtReceivedMsg txtBox
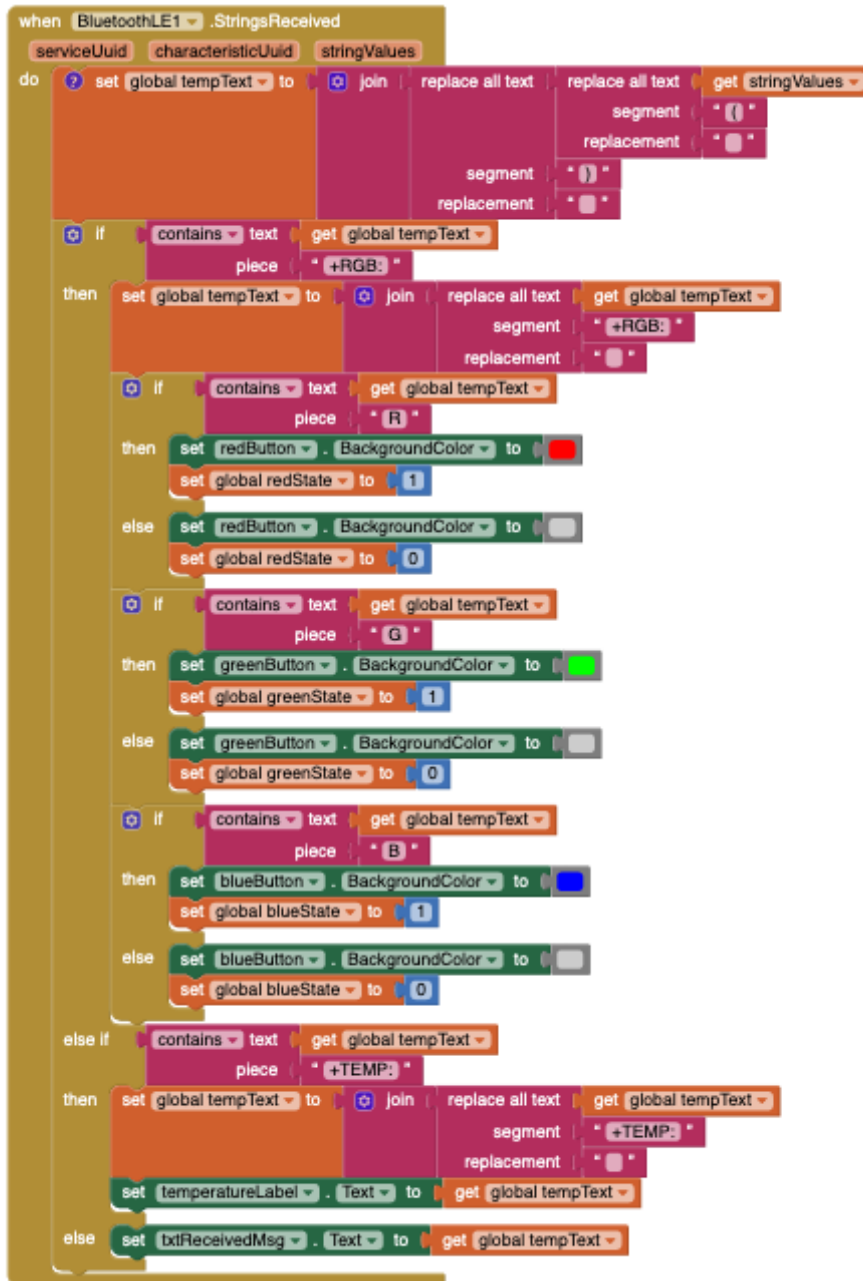
Sending +STATUS to WBZ451 using Clock2(100mSec) event. When WBZ451 received this message, it sends back the le "+RGB: r G B", upper case for ON, lower case for OFF
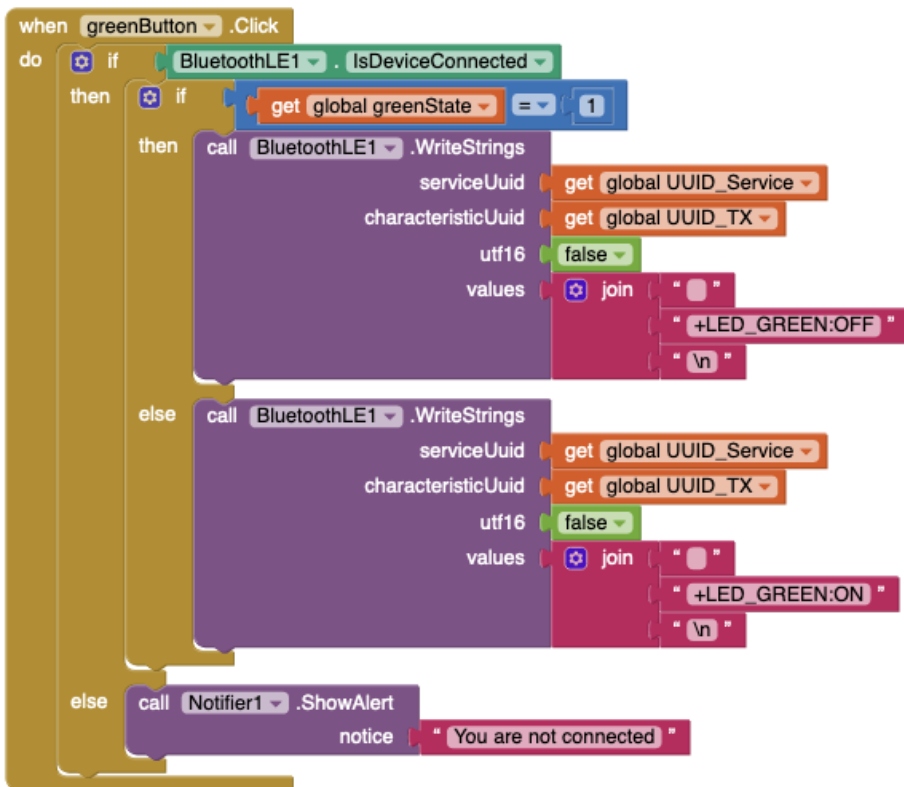

Sending Message button to WBZ451

Sending RED_LED button

Sending GREEN_LED button

Sending BLUE_LED button


```
when blueButton .Click
do    if    BluetoothLE1 . IsDeviceConnected
      then    if    get global blueState  =  1
              then   call BluetoothLE1 .WriteStrings
                           serviceUuid       get global UUID_Service
                           characteristicUuid  get global UUID_TX
                           utf16             false
                           values            join  " ■ "
                                                   " +LED_BLUE:OFF "
                                                   " \n "
              else   call BluetoothLE1 .WriteStrings
                           serviceUuid       get global UUID_Service
                           characteristicUuid  get global UUID_TX
                           utf16             false
                           values            join  " ■ "
                                                   " +LED_BLUE:ON "
                                                   " \n "
      else   call Notifier1 .ShowAlert
                    notice  " You are not connected "
```