```
/*********************************************************************
 This is an example for our nRF52 based Bluefruit LE modules

 Pick one up today in the adafruit shop!

 Adafruit invests time and resources providing this open source code,
 please support Adafruit and open-source hardware by purchasing
 products from Adafruit!

 MIT license, check LICENSE for more information
 All text above, and the splash screen below must be included in
 any redistribution
*********************************************************************/
#include <bluefruit.h>

BLEDis bledis;
BLEHidAdafruit blehid;

// the number of the pushbutton pin
const int buttonPin = 11;
const int blueLED = 15;
const int BUZZ = 16;

// variable for reading the pushbutton status
bool buttonPressed = false;
bool hasKeyPressed = false;

void setup()
{

  // initialize the pushbutton pin as an input:
  pinMode(buttonPin, INPUT);
  pinMode(BUZZ, OUTPUT);
  digitalWrite(BUZZ, HIGH);
  noTone(BUZZ);

  Serial.begin(115200);
  while ( !Serial ) delay(10);   // for nrf52840 with native usb

  Serial.println("Bluefruit52 HID Keyboard Example");
  Serial.println("---------------------------------\n");

  Serial.println();
  Serial.println("Go to your phone's Bluetooth settings to pair your device");
  Serial.println("then open an application that accepts keyboard input");

  Serial.println();
  Serial.println("Enter the character(s) to send:");
  Serial.println();

  Bluefruit.begin();
  Bluefruit.setTxPower(4);    // Check bluefruit.h for supported values
  Bluefruit.setName("Della Bluefruit");

  // Configure and Start Device Information Service
  bledis.setManufacturer("Adafruit Industries");
```

```
  bledis.setModel("Bluefruit Feather 52");
  bledis.begin();

  /* Start BLE HID
  * Note: Apple requires BLE device must have min connection interval >= 20m
  * ( The smaller the connection interval the faster we could send data).
  * However for HID and MIDI device, Apple could accept min connection interval
  * up to 11.25 ms. Therefore BLEHidAdafruit::begin() will try to set the min and max
  * connection interval to 11.25  ms and 15 ms respectively for best performance.
  */
  blehid.begin();


  // Set up and start advertising
  startAdv();
}

void startAdv(void)
{
  // Advertising packet
  Bluefruit.Advertising.addFlags(BLE_GAP_ADV_FLAGS_LE_ONLY_GENERAL_DISC_MODE);
  Bluefruit.Advertising.addTxPower();
  Bluefruit.Advertising.addAppearance(BLE_APPEARANCE_HID_KEYBOARD);

  // Include BLE HID service
  Bluefruit.Advertising.addService(blehid);

  // There is enough room for the dev name in the advertising packet
  Bluefruit.Advertising.addName();

  /* Start Advertising
  * - Enable auto advertising if disconnected
  * - Interval:  fast mode = 20 ms, slow mode = 152.5 ms
  * - Timeout for fast mode is 30 seconds
  * - Start(timeout) with timeout = 0 will advertise forever (until connected)
  *
  * For recommended advertising interval
  * https://developer.apple.com/library/content/qa/qa1931/_index.html
  */
  Bluefruit.Advertising.restartOnDisconnect(true);
  Bluefruit.Advertising.setInterval(32, 244);    // in unit of 0.625 ms
  Bluefruit.Advertising.setFastTimeout(30);      // number of seconds in fast mode
  Bluefruit.Advertising.start(0);                // 0 = Don't stop advertising after n seconds
}

void loop()
{
  Serial.setTimeout(50);
  // Only send KeyRelease if previously pressed to avoid sending
  // multiple keyRelease reports (that consume memory and bandwidth)
  buttonPressed = digitalRead(buttonPin);


//If the button is pressed, press "a" and hasKeyPressed activated. Turns on LED and buzzer

  if ( buttonPressed )
```

```
113   {
114     Serial.write("Button Pressed: Sending Message");
115     buttonPressed = false;
116 
117     blehid.keyPress('H');
118     blehid.keyPress('e');
119     blehid.keyPress('y');
120     blehid.keyPress(' ');
121     blehid.keyPress(':');
122     blehid.keyPress(')');
123     pinMode(blueLED, OUTPUT);
124     digitalWrite(blueLED, HIGH);
125     tone(BUZZ, 450, 500);
126 
127     hasKeyPressed = true;
128 
129     // Delay a bit after a report
130     //delay(5);
131   }
132 
133 
134   // If hasKeyPressed, button deactivated and keyReleased. Turns off LED and buzzer
135 
136   if ( hasKeyPressed )
137   {
138     buttonPressed = true;
139     blehid.keyRelease();
140     digitalWrite(blueLED, LOW);
141     noTone(BUZZ);
142 
143     // Delay a bit after a report to stop multiple responses for one button press
144     delay(100);
145   }
146 
147 }
148 
149 
150 
151 
152 
153 
154 
155 
156 
157 
158 
159 
160 
161 
162 
163 
164 
165 
166 
167 
168 
```

169
170
171