




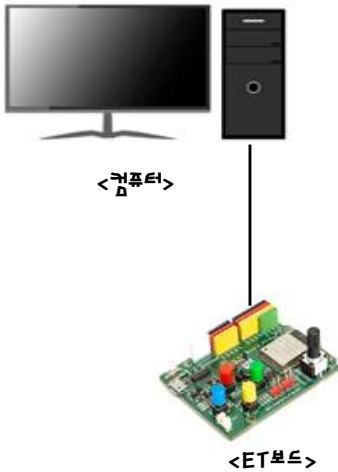


# ET보드 Level2 콘텐츠

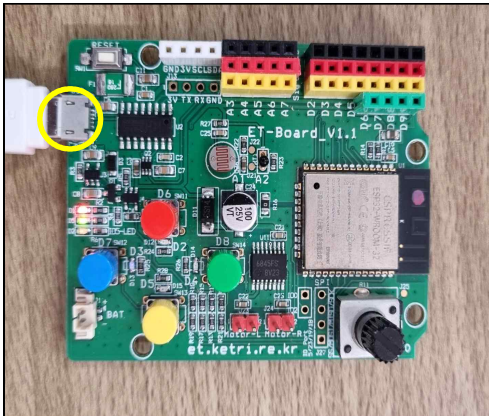
211124 최수범

학습내용	버튼 한 개를 눌렀다 뗐다 해보기(시리얼 모니터로 출력)
소 스	01._button_one_status.ino
개념	  <p>버튼은 실생활에서 자주 쓰이는 모듈이다. 예) 엘리베이터의 버튼, 스마트폰의 음량 조절 버튼 등</p>
준비물	   <p>&lt;컴퓨터&gt;                      &lt;USB Micro 5Pin 케이블&gt;                      &lt;ET보드&gt;</p>
회로 구성	 <p>&lt;컴퓨터&gt;                      ← &lt;USB Micro 5Pin 케이블&gt;                      &lt;ET보드&gt;</p>

회로  
구성



① 컴퓨터 본체의 USB포트에 케이블을 연결합니다.



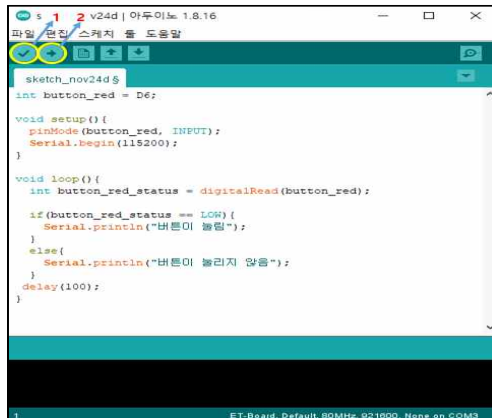
② 다른 한쪽 케이블을 ET보드에 연결합니다.

소스  
코드

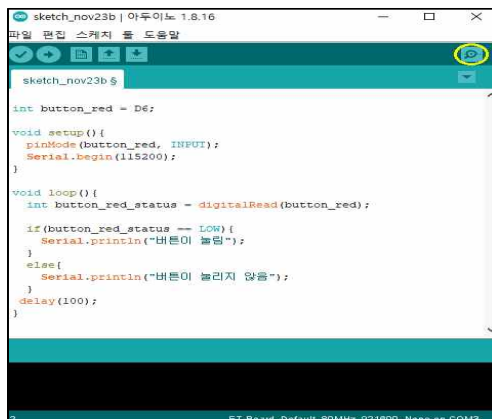
```
int button_red = D6;                                     // 빨강 버튼

void setup(){
    pinMode(button_red, INPUT);                           // 핀 모드 설정
    Serial.begin(115200);
}

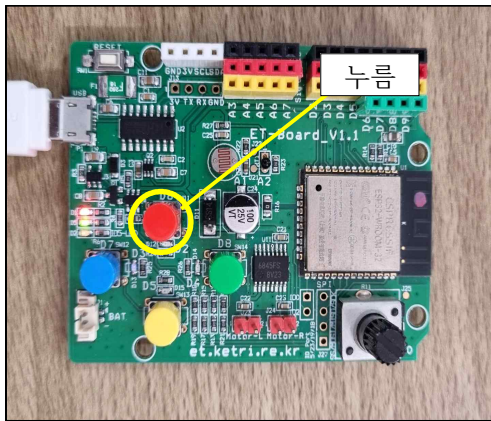
void loop(){
    // 빨강 버튼의 값을 저장
    int button_red_status = digitalRead(button_red);
    // 버튼이 눌렸는지 체크 버튼이 눌리면 LOW 눌리지 않으면 HIGH
    if(button_red_status == LOW){
        Serial.println("버튼이 눌림");
    }
    else{
        Serial.println("버튼이 눌리지 않음");
    }
    delay(100);                                           // 0.1초 대기
}
```



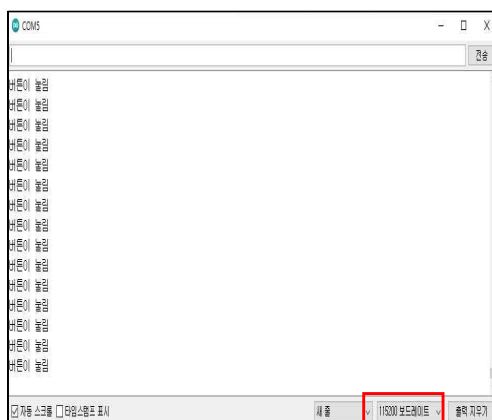
① 아두이노 코드입력 후 상단의 **1 [확인]** 아이콘을 클릭하고 완료되면 **2 [업로드]** 아이콘도 클릭해 펌웨어를 ET보드에 업로드합니다.



② 업로드까지 마치면 우측 상단 돋보기 모양의 “시리얼모니터”를 클릭해줍니다.

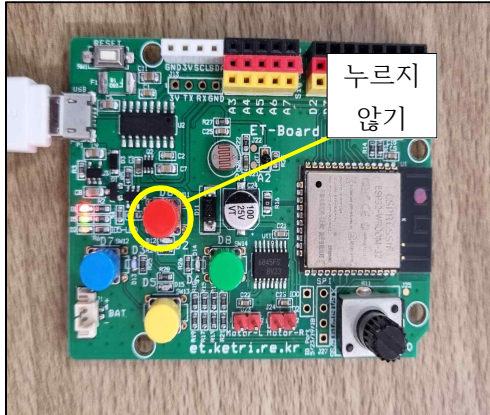


③ 빨간색 버튼을 눌러줍니다.

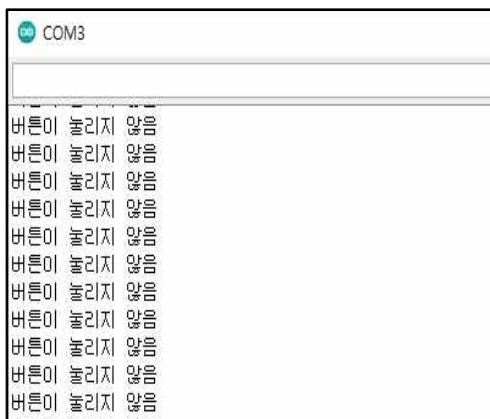


④ 시리얼모니터에 “버튼이 눌림” 이라 출력됩니다.  
(하단의 보드레이트 값을 **115200** 로 조정합니다.)

중  
작  
과  
정



⑤ 빨간색 버튼을 누르지 않습니다.



⑥ 시리얼 모니터에 “버튼이 눌리지 않음” 이라 출력됩니다.

참고  
사항

① 회로

○ button에 대한 참고 링크 :

<https://ko.wikipedia.org/wiki/%EB%B2%84%ED%8A%BC>

<https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=gu04005&logNo=221235520661>

② 소스코드

○ pinMode에 대한 참고 링크 :

<https://www.arduino.cc/reference/ko/language/functions/digital-io/pinmode/>

<http://www.iamamaker.kr/ko/tutorials/%EC%95%84%EB%91%90%EC%9D%B4%EB%85%B8-%EB%A0%88%ED%8D%BC%EB%9F%B0%EC%8A%A4-pinmode-%ED%95%A8%EC%88%98/>

○ Serial.begin에 대한 참고 링크 :

<http://www.iamamaker.kr/ko/tutorials/arduino/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-serial-begin-%ed%95%a8%ec%88%98/>

<http://www.iamamaker.kr/ko/tutorials/arduino/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-serial-println-%ed%95%a8%ec%88%98/>

○ digitalRead에 대한 참고 링크 :

<http://www.iamamaker.kr/ko/tutorials/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-digitalread-%ed%95%a8%ec%88%98/>

○ delay에 대한 참고 링크 :

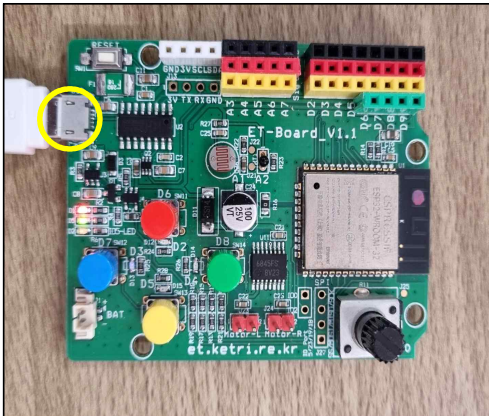
<https://www.arduino.cc/reference/ko/language/functions/time/delay/>

학습내용	버튼 두 개를 눌렀다 떴다 해보기
소스	02._button_two_status.ino
개념	<div data-bbox="300 427 758 768" data-label="Image"> </div> <div data-bbox="815 427 1273 768" data-label="Image"> </div> <p>버튼은 실생활에서 자주 쓰이는 모듈이다. 예) 엘리베이터의 버튼, 스마트폰의 음량 조절 버튼 등</p>
준비물	<div data-bbox="309 1021 598 1160" data-label="Image"> </div> <div data-bbox="402 1218 501 1254" data-label="Caption"> <p>&lt;컴퓨터&gt;</p> </div> <div data-bbox="668 983 876 1189" data-label="Image"> </div> <div data-bbox="628 1218 920 1254" data-label="Caption"> <p>&lt;USB Micro 5Pin 케이블&gt;</p> </div> <div data-bbox="973 1005 1197 1178" data-label="Image"> </div> <div data-bbox="1038 1218 1145 1254" data-label="Caption"> <p>&lt;ET보드&gt;</p> </div>
회로 구성	<div data-bbox="316 1420 603 1559" data-label="Image"> </div> <div data-bbox="354 1583 445 1617" data-label="Caption"> <p>&lt;컴퓨터&gt;</p> </div> <div data-bbox="711 1632 1031 1671" data-label="Caption"> <p>← &lt;USB Micro 5Pin 케이블&gt;</p> </div> <div data-bbox="470 1715 655 1859" data-label="Image"> </div> <div data-bbox="549 1859 651 1892" data-label="Caption"> <p>&lt;ET보드&gt;</p> </div>

회로 구성



① 컴퓨터 본체의 USB포트에 케이블을 연결합니다.



② 다른 한쪽 케이블을 ET보드에 연결합니다.

소스 코드

```
int button_red    = D6;           // 빨강 버튼
int button_blue   = D7;           // 파랑 버튼

void setup(){
    pinMode(button_red, INPUT);    // 핀 모드 설정
    pinMode(button_blue, INPUT);   // 핀 모드 설정
    Serial.begin(115200);
}

void loop(){
    // 빨강 버튼의 값을 저장
    int button_red_status = digitalRead(button_red);
    // 파랑 버튼의 값을 저장
    int button_blue_status = digitalRead(button_blue);

    // 버튼이 눌렸는지 체크 버튼이 눌리면 LOW 눌리지 않으면 HIGH
    if(button_red_status == LOW){
        Serial.println("빨강 버튼이 눌림");
    }
}
```

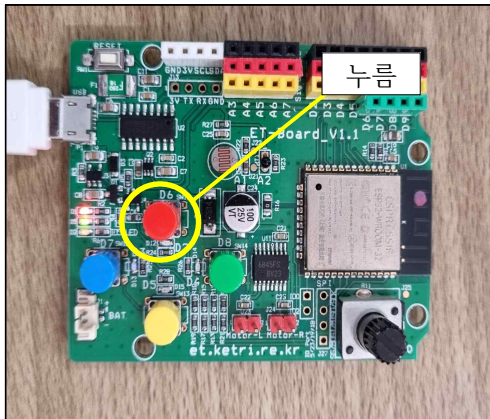


```

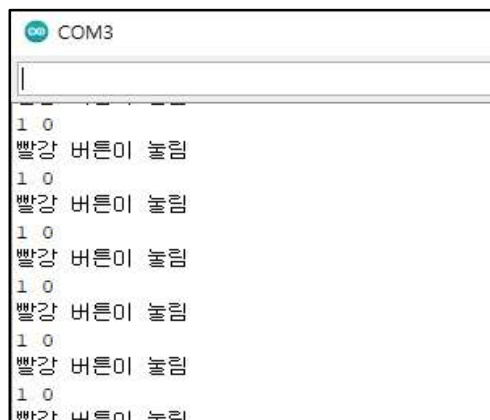
if(button_blue_status == LOW){
    Serial.println("파랑 버튼이 눌림");
}
Serial.print(button_red_status);
Serial.print(" ");
Serial.print(button_blue_status);
delay(100);           // 0.1초 대기
}

```

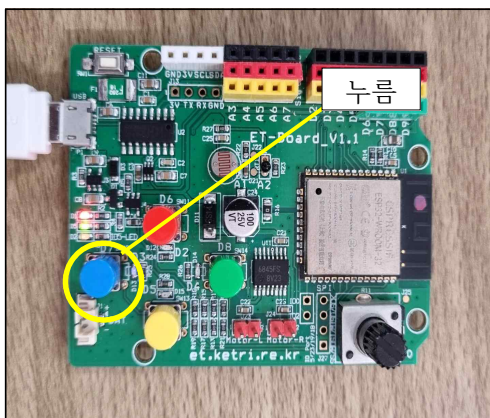
동 작  
과 정



① 빨간색 버튼을 누릅니다.

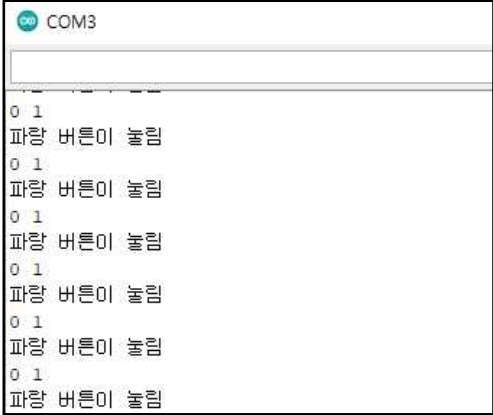
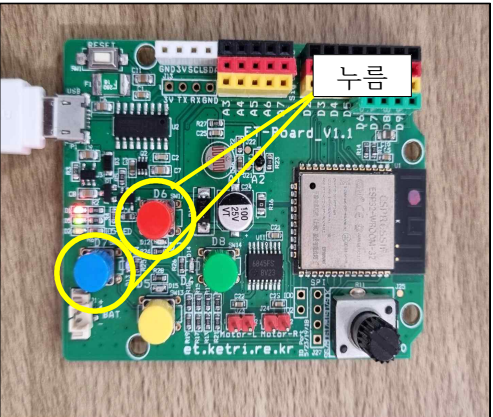
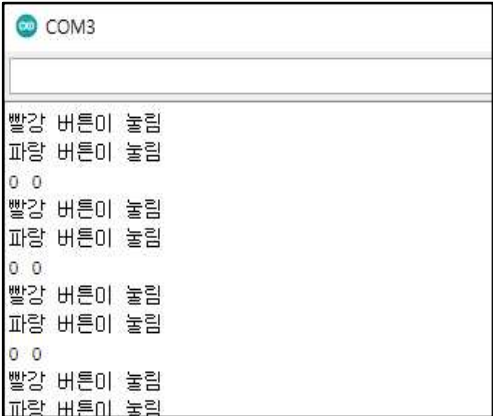


② 시리얼 모니터에 “빨강 버튼이 눌림”이라 출력됩니다.



③ 파랑 버튼을 누릅니다.



	<div data-bbox="304 210 799 624">  </div> <div data-bbox="874 389 1481 470"> <p>④ 시리얼 모니터에 “파랑 버튼이 눌림”이라 출력됩니다.</p> </div> <div data-bbox="304 696 799 1111">  </div> <div data-bbox="874 864 1372 902"> <p>⑤ 빨간 / 파란 버튼을 동시에 누릅니다.</p> </div> <div data-bbox="304 1182 799 1597">  </div> <div data-bbox="874 1350 1372 1431"> <p>⑥ 시리얼 모니터에 “빨강 버튼이 눌림”, “파랑 버튼이 눌림” 이라 출력됩니다.</p> </div>
<p>참고 사항</p>	<p>① 회로 ○ button에 대한 참고 링크 :  <a href="https://ko.wikipedia.org/wiki/%EB%B2%84%ED%8A%BC">https://ko.wikipedia.org/wiki/%EB%B2%84%ED%8A%BC</a>  <a href="https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&amp;blogId=gu04005&amp;logNo=221235520661">https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&amp;blogId=gu04005&amp;logNo=221235520661</a></p> <p>② 소스코드 ○ pinMode에 대한 참고 링크 :</p>

<https://www.arduino.cc/reference/ko/language/functions/digital-io/pinmode/>

<http://www.iamamaker.kr/ko/tutorials/%EC%95%84%EB%91%90%EC%9D%B4%EB%85%B8-%EB%A0%88%ED%8D%BC%EB%9F%B0%EC%8A%A4-pinmode-%ED%95%A8%EC%88%98/>

○ Serial.begin에 대한 참고 링크 :

<http://www.iamamaker.kr/ko/tutorials/arduino/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-serial-begin-%ed%95%a8%ec%88%98/>

<http://www.iamamaker.kr/ko/tutorials/arduino/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-serial-println-%ed%95%a8%ec%88%98/>

○ digitalRead에 대한 참고 링크 :

<http://www.iamamaker.kr/ko/tutorials/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-digitalread-%ed%95%a8%ec%88%98/>

○ delay에 대한 참고 링크 :

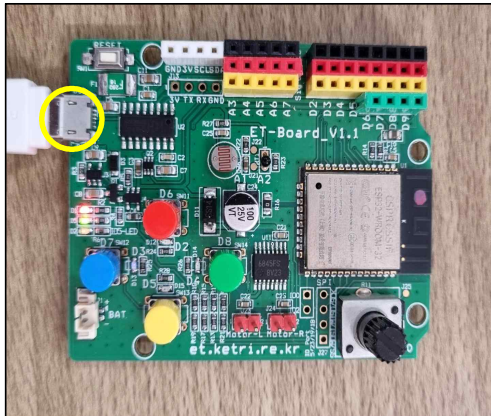
<https://www.arduino.cc/reference/ko/language/functions/time/delay/>

학습내용	모든 버튼을 눌렀다 뗐다 해보기(시리얼 모니터 출력)
소스	03._button_all_status.ino
개념	<div data-bbox="296 367 754 707" data-label="Image"> </div> <div data-bbox="815 367 1273 707" data-label="Image"> </div> <p>버튼은 실생활에서 자주 쓰이는 모듈이다. 예) 엘리베이터의 버튼, 스마트폰의 음량 조절 버튼 등</p>
준비물	<div data-bbox="309 952 598 1093" data-label="Image"> </div> <div data-bbox="405 1151 501 1184" data-label="Caption"> <p>&lt;컴퓨터&gt;</p> </div> <div data-bbox="670 916 876 1122" data-label="Image"> </div> <div data-bbox="630 1151 917 1184" data-label="Caption"> <p>&lt;USB Micro 5Pin 케이블&gt;</p> </div> <div data-bbox="975 938 1197 1111" data-label="Image"> </div> <div data-bbox="1042 1151 1144 1184" data-label="Caption"> <p>&lt;ET보드&gt;</p> </div>
회로 구성	<div data-bbox="316 1388 604 1529" data-label="Image"> </div> <div data-bbox="378 1552 469 1585" data-label="Caption"> <p>&lt;컴퓨터&gt;</p> </div> <div data-bbox="580 1603 895 1637" data-label="Caption"> <p>← &lt;USB Micro 5Pin 케이블&gt;</p> </div> <div data-bbox="472 1684 655 1825" data-label="Image"> </div> <div data-bbox="512 1830 612 1861" data-label="Caption"> <p>&lt;ET보드&gt;</p> </div>

## 회로 구성



① 컴퓨터 본체의 USB포트에 케이블을 연결합니다.



② 다른 한쪽 케이블을 ET보드에 연결합니다.

## 소스 코드

```
int button_red    = D6;           // 빨강 버튼
int button_blue   = D7;           // 파랑 버튼
int button_green   = D8;           // 초록 버튼
int button_yellow  = D9;           // 노랑 버튼

void setup(){
    pinMode(button_red, INPUT);     // 핀 모드 설정
    pinMode(button_blue, INPUT);    // 핀 모드 설정
    pinMode(button_green, INPUT);    // 핀 모드 설정
    pinMode(button_yellow, INPUT);   // 핀 모드 설정
    Serial.begin(115200);
}

void loop(){
    int button_red_status = digitalRead(button_red); // 빨강 버튼의 값을 저장
    int button_blue_status = digitalRead(button_blue); // 파랑 버튼의 값을 저장
    int button_green_status = digitalRead(button_green); // 초록 버튼의 값을 저장
    int button_yellow_status = digitalRead(button_yellow); // 노랑 버튼의 값을 저장
```

```

// 빨강 버튼이 눌렸는지 체크
if(button_red_status == LOW){
    Serial.println("빨강버튼이 눌림");
}

// 파랑 버튼이 눌렸는지 체크
if(button_blue_status == LOW){
    Serial.println("파란버튼이 눌림");
}

// 초록 버튼이 눌렸는지 체크
if(button_green_status == LOW){
    Serial.println("초록버튼이 눌림");
}

// 노랑 버튼이 눌렸는지 체크
if(button_yellow_status == LOW){
    Serial.println("노란버튼이 눌림");
}

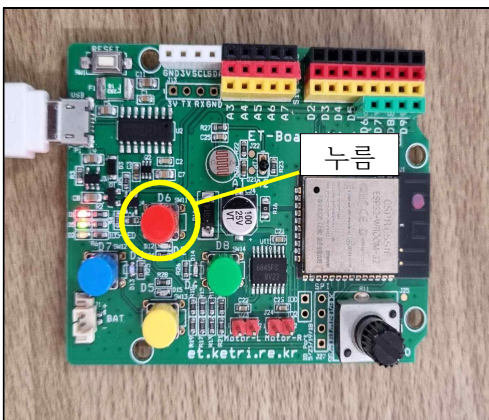
// 시리얼 플롯트를 보기 위한 코드입니다.
Serial.print(button_blue_status);
Serial.print(" ");
Serial.print(button_red_status);
Serial.print(" ");
Serial.print(button_green_status);
Serial.print(" ");
Serial.println(button_yellow_status);

delay(100);
}

```

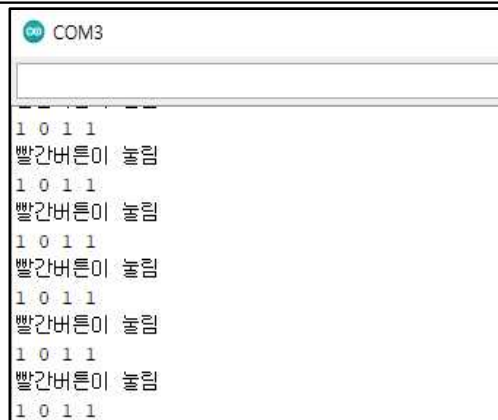
// 파랑 버튼 상태 체크  
// 빨강 버튼 상태 체크  
// 초록 버튼 상태 체크  
// 노랑 버튼 상태 체크

동 작  
과 정

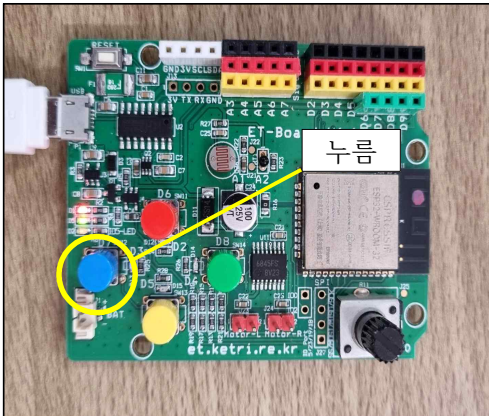


① 빨강 버튼을 누릅니다.

동 작  
과 정



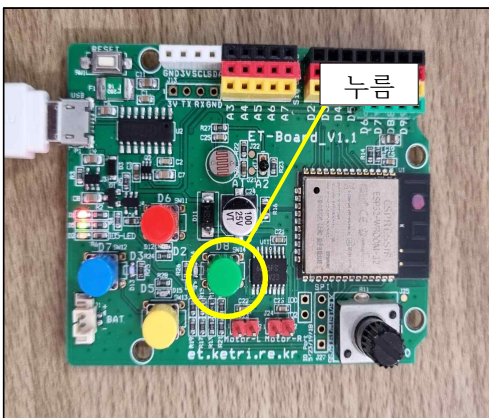
② 시리얼 모니터에 “빨간버튼이 눌림”이라 출력됩니다.



③ 파랑 버튼을 누릅니다.



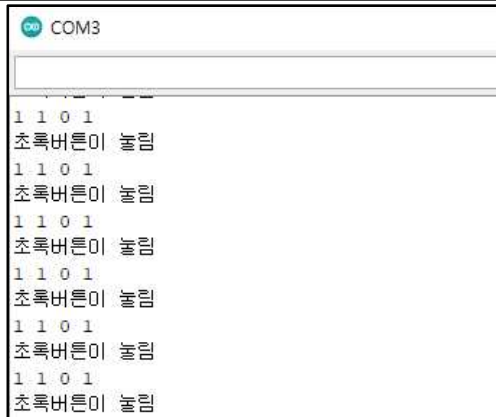
④ 시리얼 모니터에 “파란버튼이 눌림”이라 출력됩니다.



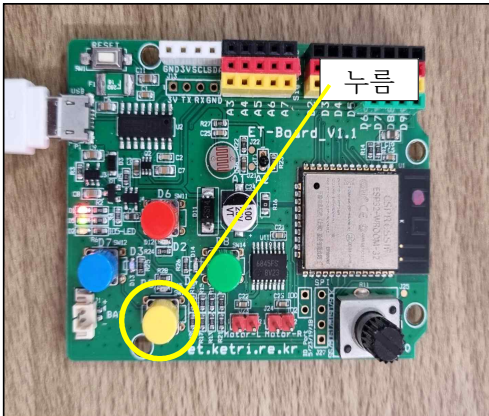
⑤ 초록 버튼을 누릅니다.



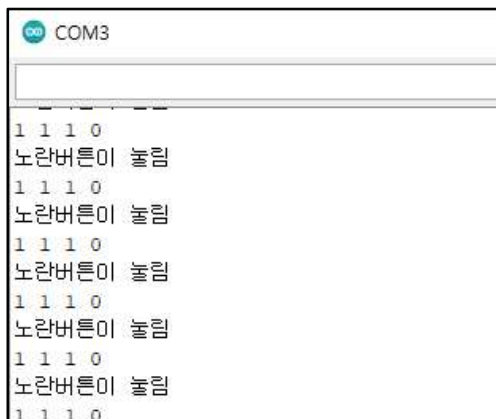
동 작  
과 정



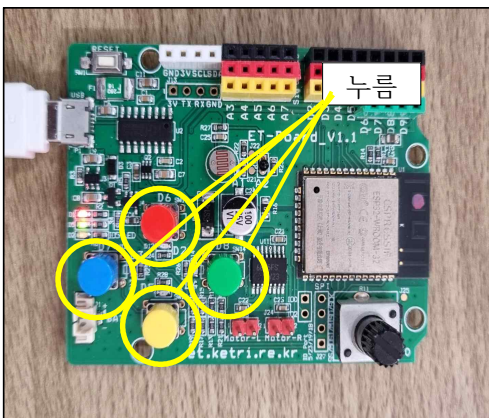
⑥ 시리얼 모니터에 “초록버튼이 눌림”이라 출력됩니다.



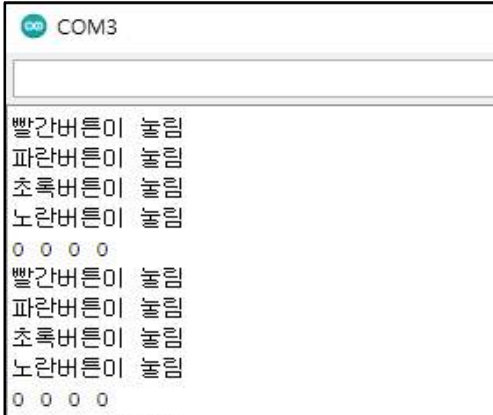
⑦ 노랑 버튼을 누릅니다.



⑧ 시리얼 모니터에 “노란버튼이 눌림”이라 출력됩니다.



⑨ 모든 버튼을 누릅니다.



- ⑩ 시리얼 모니터에 “빨간버튼이 눌림”,  
“파란버튼이 눌림”, “초록버튼이 눌림”,  
“노란버튼이 눌림”이라 출력 됩니다.

**참고  
사항**

① 회로

- button에 대한 참고 링크 :

<https://ko.wikipedia.org/wiki/%EB%B2%84%ED%8A%BC>

<https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=gu04005&logNo=221235520661>

② 소스코드

- pinMode에 대한 참고 링크 :

<https://www.arduino.cc/reference/ko/language/functions/digital-io/pinmode/>

<http://www.iamamaker.kr/ko/tutorials/%EC%95%84%EB%91%90%EC%9D%B4%EB%85%B8-%EB%A0%88%ED%8D%BC%EB%9F%B0%EC%8A%A4-pinmode-%ED%95%A8%EC%88%98/>

- Serial.begin에 대한 참고 링크 :

<http://www.iamamaker.kr/ko/tutorials/arduino/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-serial-begin-%ed%95%a8%ec%88%98/>

<http://www.iamamaker.kr/ko/tutorials/arduino/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-serial-println-%ed%95%a8%ec%88%98/>

- digitalWrite에 대한 참고 링크 :

<http://www.iamamaker.kr/ko/tutorials/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-digitalread-%ed%95%a8%ec%88%98/>

- delay에 대한 참고 링크 :

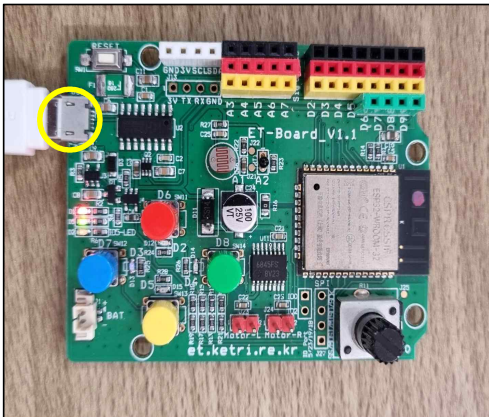
<https://www.arduino.cc/reference/ko/language/functions/time/delay/>

학습내용	버튼과 같은 색의 LED 켜다 꺼보기
소스	04._button_led_control.ino
개념	<div data-bbox="296 367 754 707" data-label="Image"> </div> <div data-bbox="815 367 1273 707" data-label="Image"> </div> <p>버튼은 실생활에서 자주 쓰이는 모듈이다. 예) 엘리베이터의 버튼, 스마트폰의 음량 조절 버튼 등</p>
준비물	<div data-bbox="309 1003 598 1144" data-label="Image"> </div> <div data-bbox="402 1200 499 1236" data-label="Caption"> <p>&lt;컴퓨터&gt;</p> </div> <div data-bbox="668 967 876 1173" data-label="Image"> </div> <div data-bbox="628 1200 919 1236" data-label="Caption"> <p>&lt;USB Micro 5Pin 케이블&gt;</p> </div> <div data-bbox="973 990 1197 1162" data-label="Image"> </div> <div data-bbox="1038 1200 1145 1234" data-label="Caption"> <p>&lt;ET보드&gt;</p> </div>
회로 구성	<div data-bbox="316 1424 603 1565" data-label="Image"> </div> <div data-bbox="375 1588 467 1624" data-label="Caption"> <p>&lt;컴퓨터&gt;</p> </div> <div data-bbox="722 1637 1042 1673" data-label="Text"> <p>← &lt;USB Micro 5Pin 케이블&gt;</p> </div> <div data-bbox="470 1718 657 1861" data-label="Image"> </div> <div data-bbox="558 1863 660 1897" data-label="Caption"> <p>&lt;ET보드&gt;</p> </div>

회로  
구성



① 컴퓨터 본체의 USB포트에 케이블을 연결합니다.



② 다른 한쪽 케이블을 ET보드에 연결합니다.

소스  
코드

```
int led_red      = D2;           // 빨강 LED
int led_blue     = D3;           // 파랑 LED
int led_green    = D4;           // 초록 LED
int led_yellow   = D5;           // 노랑 LED

int button_red   = D6;           // 빨강 버튼
int button_blue  = D7;           // 파랑 버튼
int button_green = D8;           // 초록 버튼
int button_yellow = D9;          // 노랑 버튼

int button_red_value    = 0;      // 빨강 버튼의 상태
int button_red_old_value = 1;     // 빨강 버튼의 이전 상태
int led_red_status      = 0;      // 빨강 LED 상태

int button_blue_value    = 0;     // 파랑 버튼의 상태
int button_blue_old_value = 1;    // 파랑 버튼의 이전 상태
int led_blue_status      = 0;     // 파랑 LED 상태
```

```

int button_green_value    = 0;      // 초록 버튼의 상태
int button_green_old_value = 1;      // 초록 버튼의 이전 상태
int led_green_status      = 0;      // 초록 LED 상태

int button_yellow_value   = 0;      // 노랑 버튼의 상태
int button_yellow_old_value = 1;      // 노랑 버튼의 이전 상태
int led_yellow_status     = 0;      // 노랑 LED 상태

void setup(){
    pinMode(led_red, OUTPUT);        // 핀 모드 설정
    pinMode(led_blue, OUTPUT);       // 핀 모드 설정
    pinMode(led_green, OUTPUT);      // 핀 모드 설정
    pinMode(led_yellow, OUTPUT);     // 핀 모드 설정

    pinMode(button_red, INPUT);       // 핀 모드 설정
    pinMode(button_blue, INPUT);      // 핀 모드 설정
    pinMode(button_green, INPUT);     // 핀 모드 설정
    pinMode(button_yellow, INPUT);    // 핀 모드 설정
}

void loop(){
    // 빨강 버튼 으로 빨간 LED 제어
    button_red_value = digitalRead(button_red);

    if(button_red_value == 0 && button_red_old_value == 1){
        led_red_status = 1- led_red_status;
    }
    button_red_old_value = button_red_value;
    if(led_red_status == 1){
        digitalWrite(led_red, HIGH);
    }else{
        digitalWrite(led_red, LOW);
    }

    // 파랑 버튼 으로 파랑 LED 제어
    button_blue_value = digitalRead(button_blue);

    if(button blue value == 0 && button blue old value == 1){

```

```

    led_blue_status = 1- led_blue_status;
}
button_blue_old_value = button_blue_value;
if(led_blue_status == 1){
    digitalWrite(led_blue, HIGH);
}else{
    digitalWrite(led_blue, LOW);
}

// 초록 버튼 으로 초록 LED 제어
button_green_value = digitalRead(button_green);

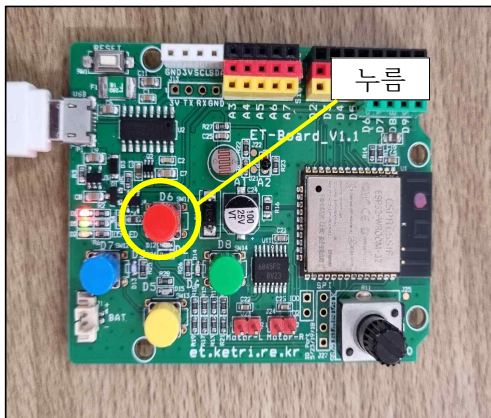
if(button_green_value == 0 && button_green_old_value == 1){
    led_green_status = 1- led_green_status;
}
button_green_old_value = button_green_value;
if(led_green_status == 1){
    digitalWrite(led_green, HIGH);
}else{
    digitalWrite(led_green, LOW);
}

// 노랑 버튼 으로 노랑 LED 제어
button_yellow_value = digitalRead(button_yellow);

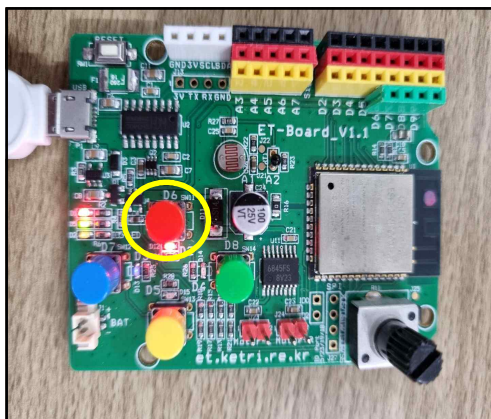
if(button_yellow_value == 0 && button_yellow_old_value == 1){
    led_yellow_status = 1- led_yellow_status;
}
button_yellow_old_value = button_yellow_value;
if(led_yellow_status == 1){
    digitalWrite(led_yellow, HIGH);
}else{
    digitalWrite(led_yellow, LOW);
}
}

```

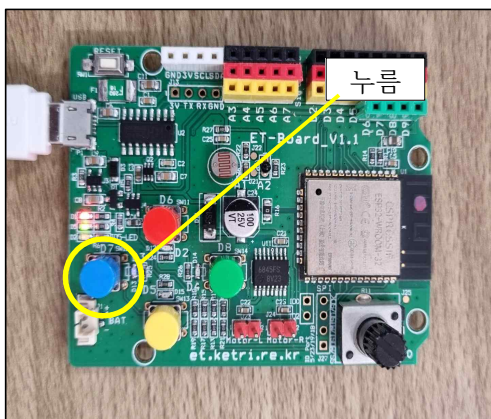




① 빨강 버튼을 누릅니다.



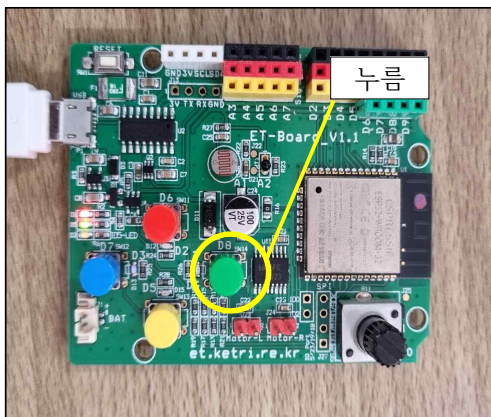
② 빨강 LED가 켜집니다.



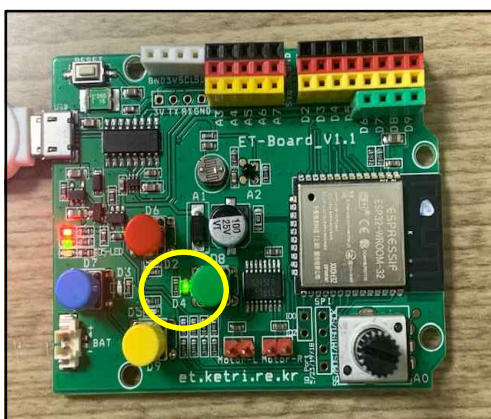
① 파랑 버튼을 누릅니다.



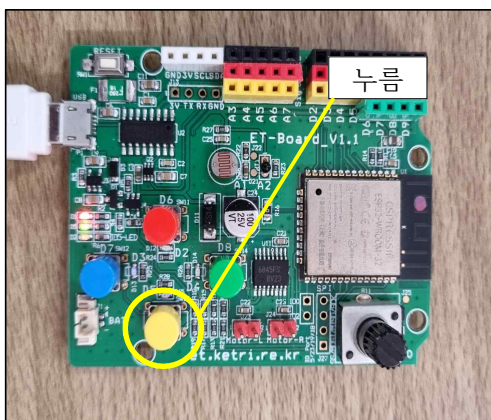
② 파랑 LED가 켜집니다.



① 초록 버튼을 누릅니다.



② 초록 LED가 켜집니다.



① 노랑 버튼을 누릅니다.



② 노랑 LED가 켜집니다.

<p><b>참고 사항</b></p>	<p>① 회로</p> <p>○ button에 대한 참고 링크 :  <a href="https://ko.wikipedia.org/wiki/%EB%B2%84%ED%8A%BC">https://ko.wikipedia.org/wiki/%EB%B2%84%ED%8A%BC</a>  <a href="https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&amp;blogId=gu04005&amp;logNo=221235520661">https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&amp;blogId=gu04005&amp;logNo=221235520661</a></p> <p>② 소스코드</p> <p>○ pinMode에 대한 참고 링크 :  <a href="https://www.arduino.cc/reference/ko/language/functions/digital-io/pinmode/">https://www.arduino.cc/reference/ko/language/functions/digital-io/pinmode/</a>  <a href="http://www.iamamaker.kr/ko/tutorials/%EC%95%84%EB%91%90%EC%9D%B4%EB%85%B8-%EB%A0%88%ED%8D%BC%EB%9F%B0%EC%8A%A4-pinmode-%ED%95%A8%EC%88%98/">http://www.iamamaker.kr/ko/tutorials/%EC%95%84%EB%91%90%EC%9D%B4%EB%85%B8-%EB%A0%88%ED%8D%BC%EB%9F%B0%EC%8A%A4-pinmode-%ED%95%A8%EC%88%98/</a></p> <p>○ Serial.begin에 대한 참고 링크 :  <a href="http://www.iamamaker.kr/ko/tutorials/arduino/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-serial-begin-%ed%95%a8%ec%88%98/">http://www.iamamaker.kr/ko/tutorials/arduino/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-serial-begin-%ed%95%a8%ec%88%98/</a>  <a href="http://www.iamamaker.kr/ko/tutorials/arduino/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-serial-println-%ed%95%a8%ec%88%98/">http://www.iamamaker.kr/ko/tutorials/arduino/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-serial-println-%ed%95%a8%ec%88%98/</a></p> <p>○ digitalWrite에 대한 참고 링크 :  <a href="http://www.iamamaker.kr/ko/tutorials/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-digitalread-%ed%95%a8%ec%88%98/">http://www.iamamaker.kr/ko/tutorials/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-digitalread-%ed%95%a8%ec%88%98/</a></p> <p>○ delay에 대한 참고 링크 :  <a href="https://www.arduino.cc/reference/ko/language/functions/time/delay/">https://www.arduino.cc/reference/ko/language/functions/time/delay/</a></p>
---------------------	---

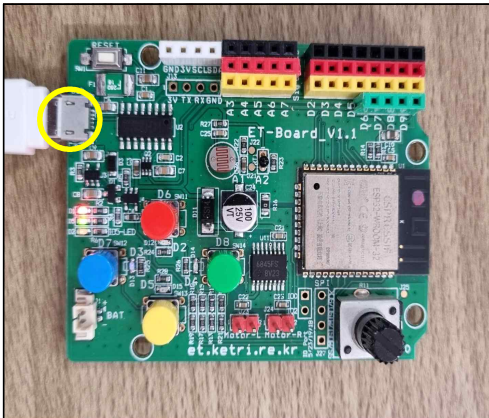
학습내용	빨강 버튼을 눌러 빨강 LED를 켜보기
소스	05._button_led_control_2.ino
개념	<div data-bbox="296 400 754 741" data-label="Image"> </div> <div data-bbox="815 400 1273 741" data-label="Image"> </div> <p>버튼은 실생활에서 자주 쓰이는 모듈이다. 예) 엘리베이터의 버튼, 스마트폰의 음량 조절 버튼 등</p>
준비물	<div data-bbox="309 1061 598 1202" data-label="Image"> </div> <div data-bbox="405 1258 501 1294" data-label="Caption"> <p>&lt;컴퓨터&gt;</p> </div> <div data-bbox="670 1025 877 1234" data-label="Image"> </div> <div data-bbox="630 1258 920 1294" data-label="Caption"> <p>&lt;USB Micro 5Pin 케이블&gt;</p> </div> <div data-bbox="975 1048 1197 1220" data-label="Image"> </div> <div data-bbox="1042 1258 1147 1292" data-label="Caption"> <p>&lt;ET보드&gt;</p> </div>
회로 구성	<div data-bbox="316 1458 603 1599" data-label="Image"> </div> <div data-bbox="400 1621 491 1655" data-label="Caption"> <p>&lt;컴퓨터&gt;</p> </div> <div data-bbox="711 1671 1029 1706" data-label="Text"> <p>← &lt;USB Micro 5Pin 케이블&gt;</p> </div> <div data-bbox="470 1749 655 1899" data-label="Image"> </div> <div data-bbox="558 1897 659 1928" data-label="Caption"> <p>&lt;ET보드&gt;</p> </div>



회로  
구성



① 컴퓨터 본체의 USB포트에 케이블을 연결합니다.



② 다른 한쪽 케이블을 ET보드에 연결합니다.

소스  
코드

```
int led_red      = D2;           // 빨강 LED
int button_red   = D6;           // 빨강 버튼

int button_red_value   = 0;      // 빨강 버튼의 상태
int button_red_old_value = 1;    // 빨강 버튼의 이전 상태
int led_red_status     = 0;      // 빨강 LED 상태

void setup(){
    pinMode(led_red, OUTPUT);     // 핀 모드 설정
    pinMode(button_red, INPUT);   // 핀 모드 설정
}

void loop(){
    // 빨강 버튼 으로 빨강 LED 제어
    button_red_value = digitalRead(button_red);

    if(button_red_value == LOW && button_red_old_value == HIGH){
        led_red_status = !led_red_status;
    }
}
```

	<pre> button_red_old_value = button_red_value; if(led_red_status == HIGH){     digitalWrite(led_red, HIGH); }else{     digitalWrite(led_red, LOW); } } </pre>
동작 과정	<div>  </div> <p>① 빨강 버튼을 누릅니다.</p> <div>  </div> <p>② 빨강 LED가 켜집니다.</p>
참고 사항	<p>① 회로</p> <ul style="list-style-type: none"> <li>○ button에 대한 참고 링크 :  <a href="https://ko.wikipedia.org/wiki/%EB%B2%84%ED%8A%BC">https://ko.wikipedia.org/wiki/%EB%B2%84%ED%8A%BC</a>  <a href="https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&amp;blogId=gu04005&amp;logNo=221235520661">https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&amp;blogId=gu04005&amp;logNo=221235520661</a> </li> </ul> <p>② 소스코드</p> <ul style="list-style-type: none"> <li>○ pinMode에 대한 참고 링크 :  <a href="https://www.arduino.cc/reference/ko/language/functions/digital-io/pinmode/">https://www.arduino.cc/reference/ko/language/functions/digital-io/pinmode/</a>  <a href="http://www.iamamaker.kr/ko/tutorials/%EC%95%84%EB%91%90%EC%9D%B4%EB%85%B8-%EB%A0%88%ED%8D%BC%EB%9F%B0%EC%8A%A4-pinmode-%ED%95%A8%EC%88%98/">http://www.iamamaker.kr/ko/tutorials/%EC%95%84%EB%91%90%EC%9D%B4%EB%85%B8-%EB%A0%88%ED%8D%BC%EB%9F%B0%EC%8A%A4-pinmode-%ED%95%A8%EC%88%98/</a> </li> </ul>



○ Serial.begin에 대한 참고 링크 :

<http://www.iamamaker.kr/ko/tutorials/arduino/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-serial-begin-%ed%95%a8%ec%88%98/>

<http://www.iamamaker.kr/ko/tutorials/arduino/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-serial-println-%ed%95%a8%ec%88%98/>

○ digitalRead에 대한 참고 링크 :

<http://www.iamamaker.kr/ko/tutorials/%ec%95%84%eb%91%90%ec%9d%b4%eb%85%b8-%eb%a0%88%ed%8d%bc%eb%9f%b0%ec%8a%a4-digitalread-%ed%95%a8%ec%88%98/>

○ delay에 대한 참고 링크 :

<https://www.arduino.cc/reference/ko/language/functions/time/delay/>