



드론으로 배우는
프로그래밍 교실

Ad1. 업 다운 게임 만들기



01 Serial 통신	01
시리얼 통신이란?	02
하드웨어 시리얼	04
Serial통신 실습하기	05
02 조건문	07
if문	08
if문 작성 해보기	11
03 상수에 대해서	13
상수란?	12
상수 작성 해보기	13



드론으로 배우는
프로그래밍 교실

초판발행 2016년 9월 23일
지은이 최정애 | 펴낸이 최정애
펴낸곳 WHIT | 주소 안산시 한양대로55 창업보육센터 B01
전화 010-5125-2139

Published by WHIT. Printed in Korea
Copyright © 2016 최정애 & WHIT

이 책의 저작권은 최정애와 WHIT에 있습니다.
저작권법에 의해 보호를 받는 저작물이므로 무단 복제 및 무단 전재를 금합니다.

01 업 다운 게임 만들기1



업 다운 게임은 사회자가 특정 숫자를 정하면 그 숫자를 맞추는 게임입니다. 시계방향으로 돌아가면서 숫자를 하나씩 부릅니다. 사회자는 불린 숫자가 정한 숫자보다 높으면 다운을, 낮으면 업을 말해줍니다. 게임 시작 전 횟수를 정해 놓고 그 횟수 내에 맞추면 사회자가 지는 것이고, 횟수를 넘어가면 숫자를 맞춘 사람이 진 것으로 됩니다.

업 다운 게임이란?

업 다운 게임

업 앤 다운

업 다운 게임 만들기1

스위치,
가변저항
사용

- 1 다음과 같이 코드를 작성하여 아두이노에 업로드합니다.

up_down_1

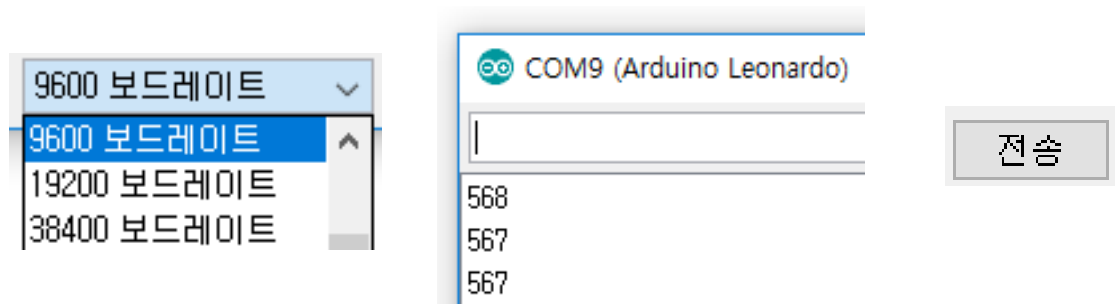
```

1  int cnt = 0;
2
3  void setup() {
4      Serial.begin(9600);
5      pinMode(3, OUTPUT);
6      pinMode(2, INPUT);
7      randomSeed(analogRead(5));
8      delay(3000);
9  }
10
11 void loop() {
12     if (digitalRead(2) == HIGH) {
13         while (digitalRead(2) == HIGH);
14         cnt++;
15         int temp = analogRead(0);
16         Serial.println(temp);
17         delay(50);
18     }
19 }
```

<그림1-6> 시리얼 통신 코드

- 2  버튼을 눌러 시리얼 모니터를 켭니다.

- 3 보드레이트를 맞춘 후 가변저항을 돌려가며 스위치를 눌렀을 때 값이 나오는지 확인합니다.



<그림1-7> 시리얼 통신 확인

스위치, 가변저항 사용 해석

```
int cnt = 0; // 누른 횟수 저장 변수

void setup() {
  Serial.begin(9600); // 시리얼 통신 시작
  pinMode(3, OUTPUT); // 3번 핀 출력으로 설정
  pinMode(2, INPUT); // 2번 핀 입력으로 설정
  randomSeed(analogRead(5)); // 랜덤 시드 설정
  delay(3000); // 3초 대기
}

void loop() {
  if (digitalRead(2) == HIGH) { // 만약 스위치가 눌리면
    while (digitalRead(2) == HIGH); // 땔 때 까지 대기
    cnt++; // 누른 횟수 증가
    int temp = analogRead(0); // 가변저항 값 저장
    Serial.println(temp); // 값 출력
    delay(50); // 채터링 방지
  }
}
```

사용자 입력 받기 함수화

사용자 입력 받기 함수화

1 다음과 같이 코드를 함수화합니다.

```

12 if (digitalRead(2) == HIGH) {
13   while (digitalRead(2) == HIGH);
14   cnt++;
15   int temp = analogRead(0);
16   Serial.println(temp);
17   delay(50);
18 }

17 int guessNumber() {
18   if (digitalRead(2) == HIGH) {
19     while (digitalRead(2) == HIGH);
20     cnt++;
21     int temp = analogRead(0);
22     Serial.println(temp);
23     delay(50);
24     return temp;
25   }
26 }

```


2 loop()의 코드를 다음과 같이 변형합니다.

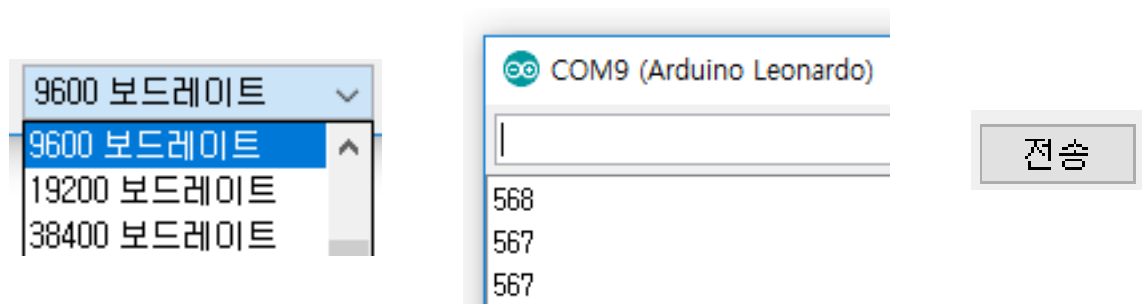
바뀐 부분

```

up_down_2
1 int cnt = 0;
2
3 void setup() {
4   Serial.begin(9600);
5   pinMode(3, OUTPUT);
6   pinMode(2, INPUT);
7   randomSeed(analogRead(5));
8   delay(3000);
9 }
10
11 void loop() {
12   int val = random(0, 1024);
13   while (digitalRead(2) == LOW);
14   int num = guessNumber();
15 }
16
17 int guessNumber() {
18   if (digitalRead(2) == HIGH) {
19     while (digitalRead(2) == HIGH);
20     cnt++;
21     int temp = analogRead(0);
22     Serial.println(temp);
23     delay(50);
24     return temp;
25   }
26 }

```

- 3  버튼을 눌러 시리얼 모니터를 켭니다.
- 4 보드레이트를 맞춘 후 가변저항을 돌려가며 스위치를 눌렀을 때 값이 나오는지 확인합니다.



사용자 입력 받기 해석

```
int cnt = 0; // 누른 횟수 저장 변수

void setup() {
  Serial.begin(9600); // 시리얼 통신 시작
  pinMode(3, OUTPUT); // 3번 핀 출력으로 설정
  pinMode(2, INPUT); // 2번 핀 입력으로 설정
  randomSeed(analogRead(5)); // 랜덤 시드 설정
  delay(3000); // 3초 대기
}

void loop() {
  int val = random(0, 1024); // 0부터 1023 사이의 랜덤 값 저장
  while (digitalRead(2) == LOW); // 스위치 입력 대기
  int num = guessNumber(); // 함수 호출 및 데이터 저장
}

int guessNumber() { // guessNumber() 함수 정의
  if (digitalRead(2) == HIGH) { // 만약 스위치가 눌리면
    while (digitalRead(2) == HIGH); // 땔 때 까지 대기
    cnt++; // 누른 횟수 증가
    int temp = analogRead(0); // 가변저항 값 저장
    Serial.println(temp); // 값 출력
    delay(50); // 채터링 방지
    return temp; // 가변저항 값 반환
  }
}
```


02 업 다운 게임 만들기2



업 다운 게임은 사회자가 특정 숫자를 정하면 그 숫자를 맞추는 게임입니다. 시계방향으로 돌아가면서 숫자를 하나씩 부릅니다. 사회자는 불린 숫자가 정한 숫자보다 높으면 다운을, 낮으면 업을 말해줍니다. 게임 시작 전 횟수를 정해 놓고 그 횟수 내에 맞추면 사회자가 지는 것이고, 횟수를 넘어가면 숫자를 맞춘 사람이 진 것으로 됩니다.


업 다운 게임 만들기2

- 1 다음과 같이 코드를 변형하여 아두이노에 업로드합니다.

바뀐 부분

```
up_down_3
1 int cnt = 0;
2
3 void setup() {
4   Serial.begin(9600);
5   pinMode(3, OUTPUT);
6   pinMode(2, INPUT);
7   randomSeed(analogRead(5));
8   delay(3000);
9 }
10
11 void loop() {
12   int val = random(0, 1024);
13   cnt = 0;
14
15   Serial.println("Guess what number I think(0~1023)");
16   Serial.println();
17
18   while (1) {
19     while (digitalRead(2) == LOW);
20     int num = guessNumber();
21
22   }
23 }
24
25 int guessNumber() {
26   if (digitalRead(2) == HIGH) {
27     while (digitalRead(2) == HIGH);
28     cnt++;
29     int temp = analogRead(0);
30     Serial.println(temp);
31     delay(50);
32     return temp;
33   }
34 }
```

<그림1-6> 시리얼 통신 코드

- 3  버튼을 눌러 시리얼 모니터를 켭니다.
- 4 보드레이트를 맞춘 후 가변저항을 돌려가며 스위치를 눌렀을 때 값이 나오는지 확인합니다.

```
int cnt = 0; // 누른 횟수 저장 변수
```

```
void setup() {  
  Serial.begin(9600); // 시리얼 통신 시작  
  pinMode(3, OUTPUT); // 3번 핀 출력으로 설정  
  pinMode(2, INPUT); // 2번 핀 입력으로 설정  
  randomSeed(analogRead(5)); // 랜덤 시드 설정  
  delay(3000); // 3초 대기  
}
```

```
void loop() {  
  int val = random(0, 1024); // 0부터 1023 사이의 랜덤 값 저장  
  cnt = 0; // 변수 0으로 초기화
```

```
  Serial.println("Guess what number I think(0~1023)");  
  Serial.println(); // 문구 출력
```

```
  while (1) { // 무한 반복  
    while (digitalRead(2) == LOW); // 스위치 입력 대기  
    int num = guessNumber(); // 가변저항 값 저장
```

```
  }  
}
```

```
int guessNumber() { // guessNumber() 함수 정의  
  if (digitalRead(2) == HIGH) { // 만약 스위치가 눌리면  
    while (digitalRead(2) == HIGH); // 떴을 때 까지 대기  
    cnt++; // 누른 횟수 증가  
    int temp = analogRead(0); // 가변저항 값 저장  
    Serial.println(temp); // 값 출력  
    delay(50); // 채터링 방지  
    return temp; // 가변저항 값 반환  
  }  
}
```

업 다운 게임 빈칸 채우기

5 다음 부분의 코드를 변형하여 코드를 완성해 봅시다.

up_down_3

```

1 int cnt = 0;
2
3 void setup() {
4   Serial.begin(9600);
5   pinMode(3, OUTPUT);
6   pinMode(2, INPUT);
7   randomSeed(analogRead(5));
8   delay(3000);
9 }
10
11 void loop() {
12   int val = random(0, 1024);
13   cnt = 0;
14
15   Serial.println("Guess what number I think(0~1023)");
16   Serial.println();
17
18   while (1) {
19     while (digitalRead(2) == LOW);
20     int num = guessNumber();
21
22   }
23 }
24
25 int guessNumber() {
26   if (digitalRead(2) == HIGH) {
27     while (digitalRead(2) == HIGH);
28     cnt++;
29     int temp = analogRead(0);
30     Serial.println(temp);
31     delay(50);
32     return temp;
33   }
34 }

```

변형할 부분

꿀TIP

조건문 사용하기

조건문을 사용하여
num값이 val값보다
큰지 작은지 비교해야
합니다.

03 업 다운 게임 만들기3



업 다운 게임은 사회자가 특정 숫자를 정하면 그 숫자를 맞추는 게임입니다. 시계방향으로 돌아가면서 숫자를 하나씩 부릅니다. 사회자는 불린 숫자가 정한 숫자보다 높으면 다운을, 낮으면 업을 말해줍니다. 게임 시작 전 횟수를 정해 놓고 그 횟수 내에 맞추면 사회자가 지는 것이고, 횟수를 넘어가면 숫자를 맞춘 사람이 진 것으로 됩니다.

업 다운 게임 만들기3

업 다운 게임 최종

```
18 while (1) {
19     while (digitalRead(2) == LOW);
20     int num = guessNumber();
21
22 }
```



```
18 while (1) {
19     while (digitalRead(2) == LOW);
20     int num = guessNumber();
21
22     if (num > val) {
23         Serial.println("down");
24     } else if (num < val) {
25         Serial.println("up");
26     } else {
27         Serial.println("You got it!!");
28         Serial.print("Your count : ");
29         Serial.println(cnt);
30         Serial.println();
31         delay(2000);
32         break;
33     }
34
35 }
```

2 업 다운 게임 최종 코드입니다.

up_down_4

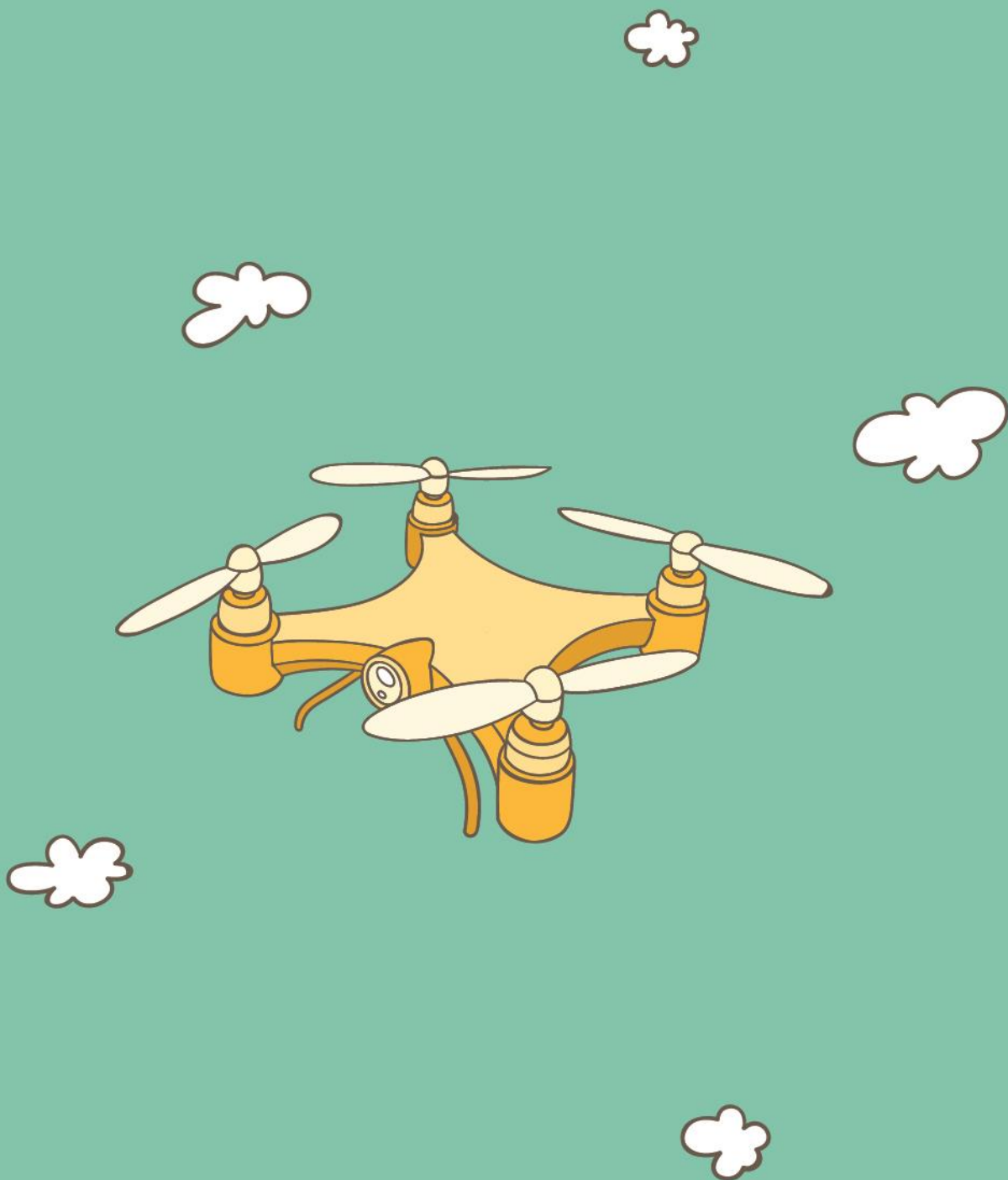
```
1 int cnt = 0;
2
3 void setup() {
4     Serial.begin(9600);
5     pinMode(3, OUTPUT);
6     pinMode(2, INPUT);
7     randomSeed(analogRead(5));
8     delay(3000);
9 }
```

```
10
11 void loop() {
12   int val = random(0, 1024);
13   cnt = 0;
14
15   Serial.println("Guess what number I think(0~1023)");
16   Serial.println();
17
18   while (1) {
19     while (digitalRead(2) == LOW);
20     int num = guessNumber();
21
22     if (num > val) {
23       Serial.println("down");
24     } else if (num < val) {
25       Serial.println("up");
26     } else {
27       Serial.println("You got it!!");
28       Serial.print("Your count : ");
29       Serial.println(cnt);
30       Serial.println();
31       delay(2000);
32       break;
33     }
34
35   }
36 }
37
38 int guessNumber() {
39   if (digitalRead(2) == HIGH) {
40     while (digitalRead(2) == HIGH);
41     cnt++;
42     int temp = analogRead(0);
43     Serial.println(temp);
44     delay(50);
45     return temp;
46   }
47 }
```

업 다운 게임 최종 해석

편의를 위하여 loop()문만 해석합니다.

```
void loop() {  
  int val = random(0, 1024); //0부터 1023 사이의 랜덤 값 저장  
  cnt = 0; // 변수 0으로 초기화  
  
  Serial.println("Guess what number I think(0~1023)");  
  Serial.println(); // 문구 출력  
  
  while (1) { // 무한 반복  
    while (digitalRead(2) == LOW); // 스위치 입력 대기  
    int num = guessNumber(); // 가변저항 값 저장  
  
    if (num > val) { // 만약 num이 val보다 크면  
      Serial.println("down"); // 문구 출력  
    } else if (num < val) { // 만약 num이 val보다 작으면  
      Serial.println("up"); // 문구 출력  
    } else { // 그 외의 경우 num이 val과 같으면  
      Serial.println("You got it!!"); // 문구 출력  
      Serial.print("Your count : "); // 문구 출력  
      Serial.println(cnt); // 데이터 출력  
      Serial.println(); // 줄 바꿈  
      delay(2000); // 2초 대기  
      break; // while문 빠져나가기  
    }  
  }  
}
```

WHIT