

비트가 한개가 높아나면 배수로 높아남(x2)

← x2		← x2		← x2	
bit	bit	bit	bit	bit	bit
			0	0	
			0	1	
			1	0	
			1	1	

0~3 (4개)

1100101

1bit

2bit

3bit

4bit

8 16 나눈 나머지가

& 비트연산자의 활용

$$\begin{array}{r} .0110\ 1001 \\ \& .0000\ 1111 \\ \hline .0000\ 1001 \end{array}$$

값을 통과시키는 효과

값을 막는 효과

1개의 점(pixel) = R G B
8bit/8bit/8bit

0~127까지 빛의 세를 조절 가능 (총 128개)
0000 0000 ~ 1111 1111

*비트 이클 연산자

<< 1

0~3 bit 까지만 이동

1010

.00010100

0 .00101000

빈자리는 + 0

overflow 제거

>>

53

52.5

44

43.5

52

43

```
public static void main(String[] args) {
```

```
int i = 4;
```

```
System.out.println(i++);
```

i를 증가시키기 전에 i 값을 출력해라.

```
    printf i;
```

```
int temp = i;
```

```
i = i +1
```

```
System.out.println(temp)
```

```
int i = 7;
```

```
i = i++ ;
```

```
    printf i;
```

```
int temp = i;
```

```
i = i+1;    i = 8
```

```
i = temp    i = 7
```

```
i = 7;
```

i를 증가시키기 전의 i 값을 놓는다.

i++ 증가된 i 값을 놓는다.

*keyboard 입력과 읽기

keyboard	

1. 입력 (1 2 3)

<https://www.tutorial.com/developers/keycode/for/a-keycode> 보기



