

OpenCV를 활용한 이미지 처리2

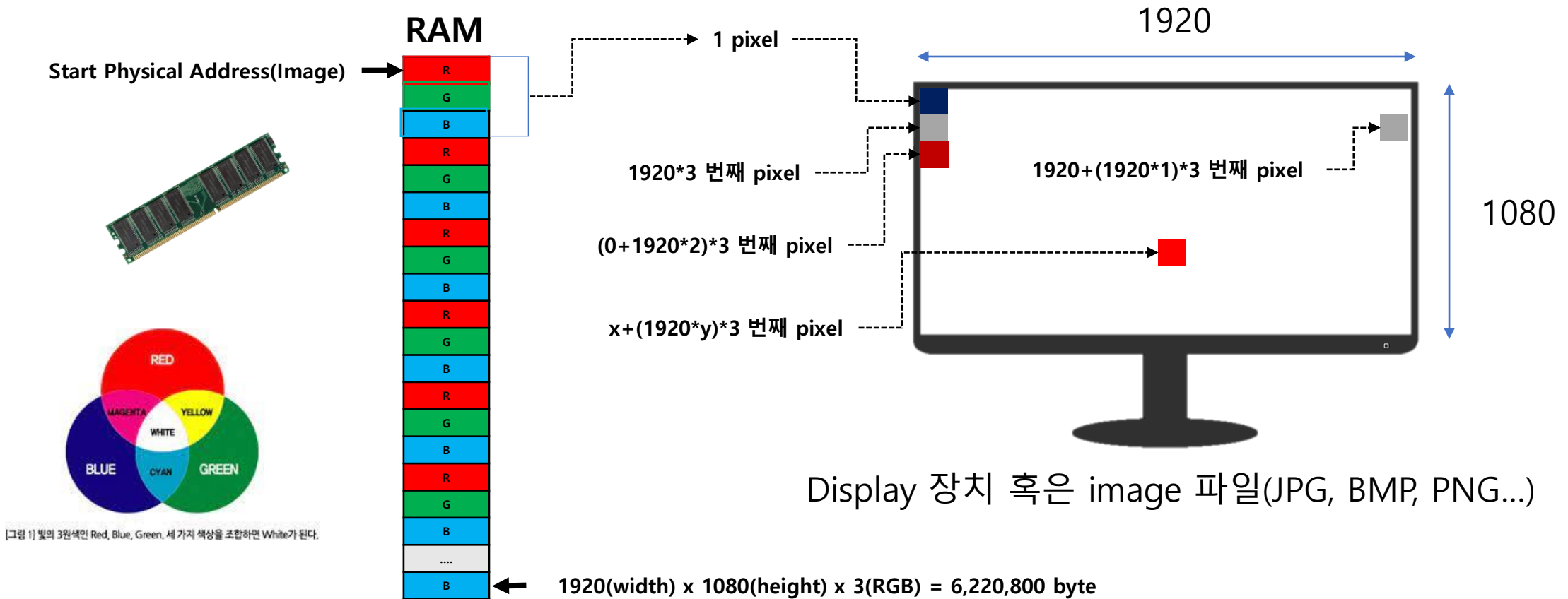


무엇을 해볼까요?

- pixel과 image
- 특정 색상을 찾아라
- 이미지에서 특징 추출(corner, line)
- 이미지 변환(이동, 회전, Affine, Perspective Transform)
- 물체를 찾기 위한 ROI와 영상 2진화
- Template Matching으로 물체인식
- 숨은 그림 찾기

무엇을 해볼까요?

- pixel과 image 그리고 OpenCV cv::Mat

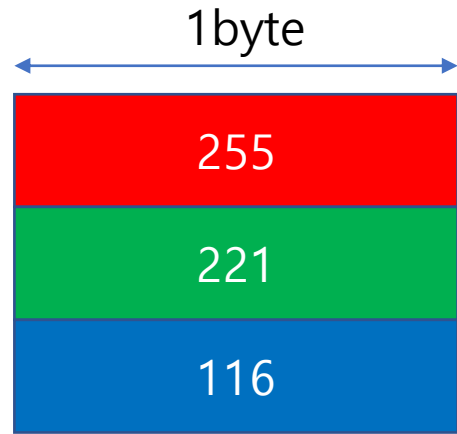


무엇을 해볼까요?

- pixel과 image 그리고 OpenCV cv::Mat



[그림 1] 빛의 3원색인 Red, Blue, Green. 세 가지 색상을 조합하면 White가 된다.



색 편집

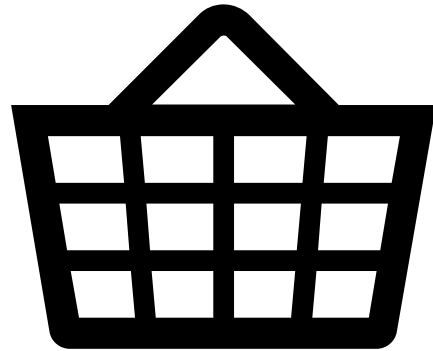


#EBDD74
RGB
235 빨강
221 녹색
116 파랑



무엇을 해볼까요?

- pixel과 image 그리고 OpenCV cv::Mat



cv::Mat

이미지 정보는 Matrix



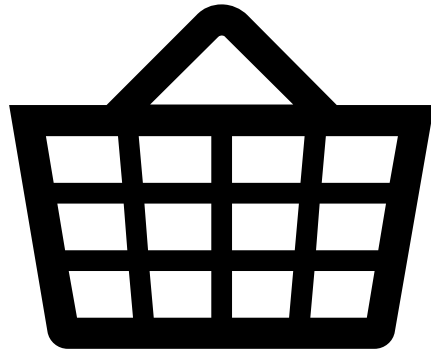
무엇을 해볼까요?

- pixel과 image 그리고 OpenCV cv::Mat

파일에서 이미지를 읽어서 cv::Mat이라는 컨테이너에 저장해라

```
cv::Mat image = cv::imread("/home/dsu/work/01_opencv_imshow_example/test.jpg");
```

읽어올 파일 경로



cv::Mat

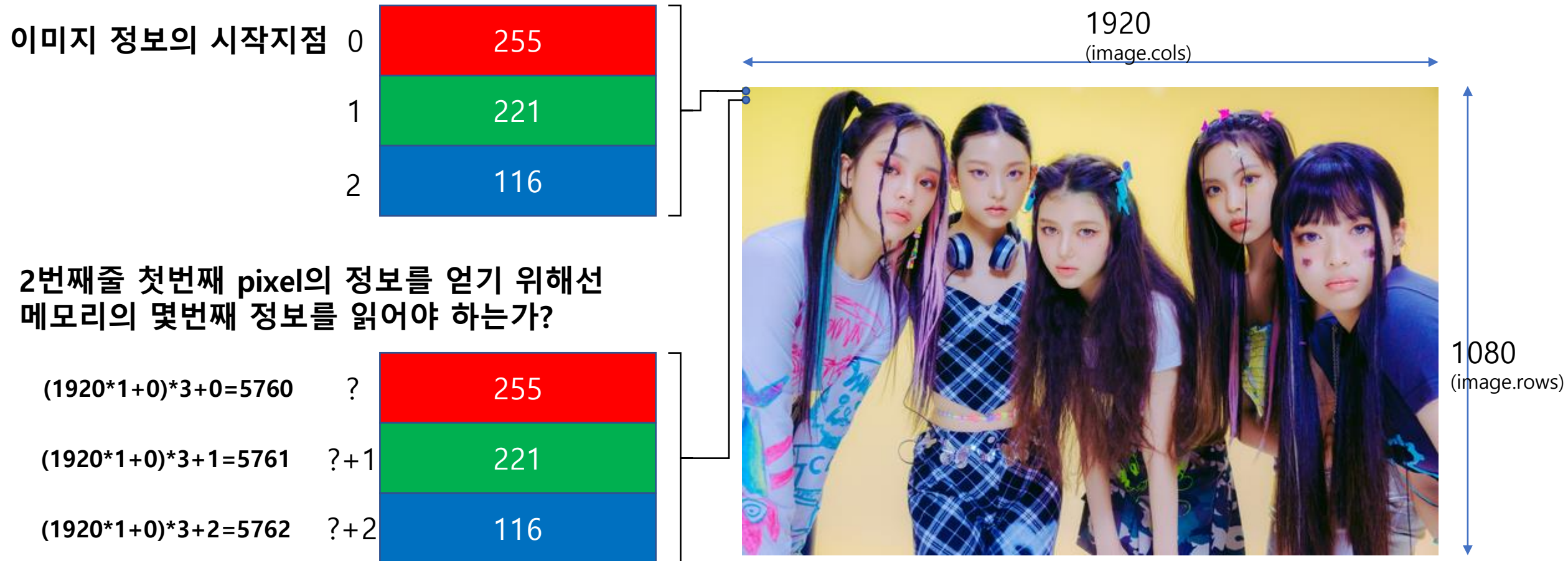
이미지 정보는 Matrix



1080

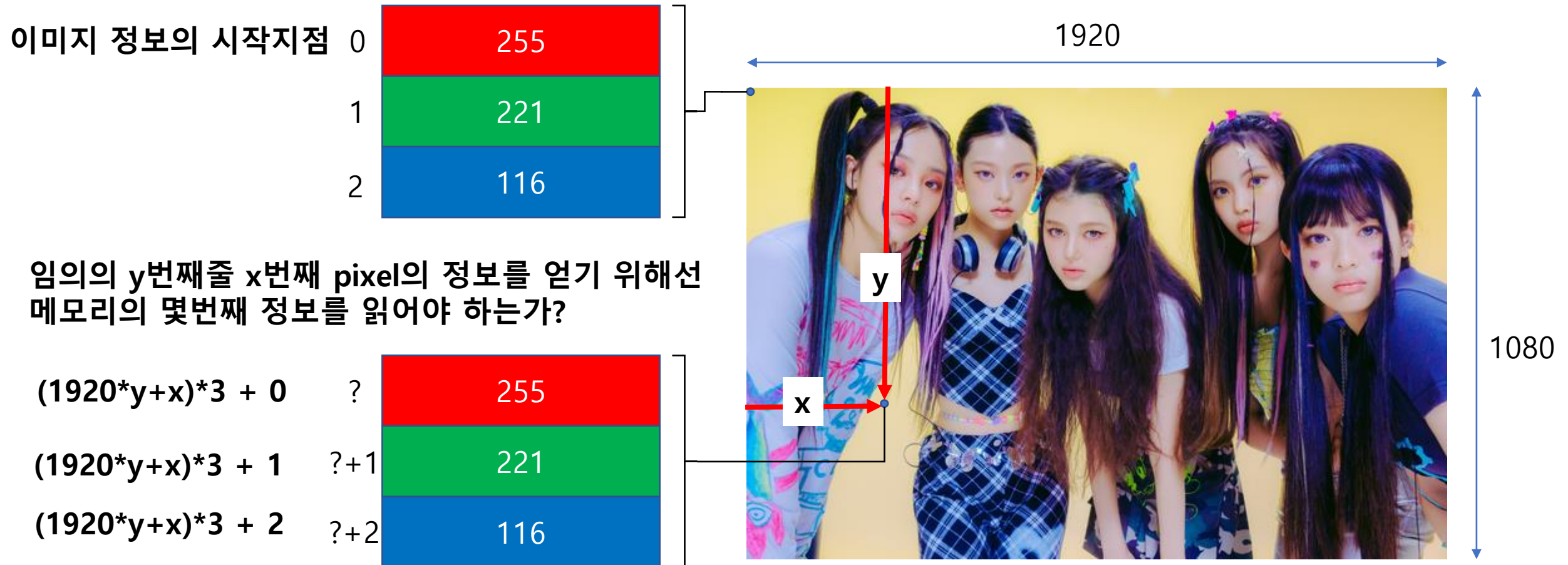
무엇을 해볼까요?

- pixel과 image 그리고 OpenCV cv::Mat



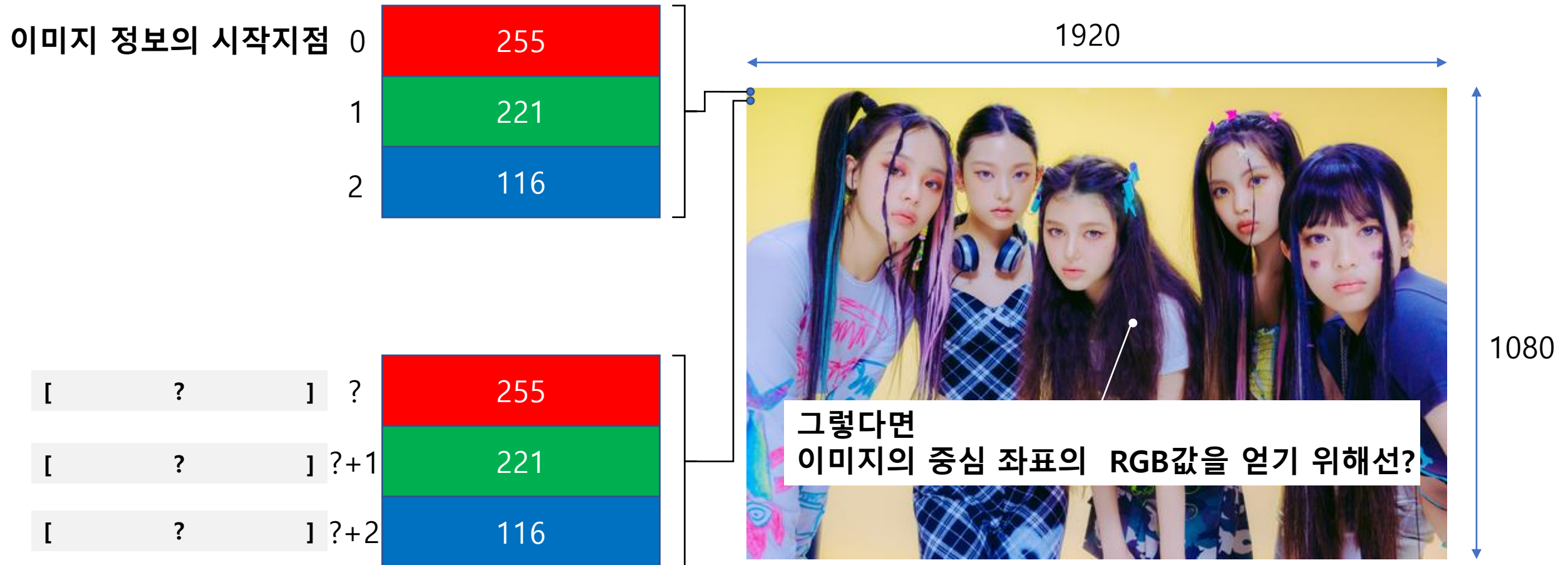
무엇을 해볼까요?

- pixel과 image 그리고 OpenCV cv::Mat



무엇을 해볼까요?

- pixel과 image 그리고 OpenCV cv::Mat



무엇을 해볼까요?

- pixel과 image 그리고 OpenCV cv::Mat
- cv::Mat에 저장되어 있는 이미지 중심좌표의 RGB값을 읽어 봅시다.
 - 이미지 사이즈 확인
 - 이미지 중심점 확인



```
#include "opencv2/opencv.hpp"
#include <stdio.h>

int main(void)
{
    printf("Hello World\n");
    cv::Mat image = cv::imread("/home/dsu/work/01_opencv_imshow_example/test.jpg");
    int image_width = image.cols;
    int image_height = image.rows;
    printf("image width = %d\n", image_width);
    printf("image height = %d\n", image_height);
    cv::imshow("test", image);
    cv::waitKey(0);

    return 0;
}
```

예제) 09_opencv/01_image_center.cpp

무엇을 해볼까요?



- pixel과 image 그리고 OpenCV cv::Mat
- 중심좌표의 RGB값을 읽어 봅시다.
 - 이미지 사이즈 확인 : image_width, image_height
 - 이미지 중심점 좌표 : image_width/2, image_height/2

`gcc 01_image_center.cpp -o test `pkg-config --cflags --libs opencv4` -lstdc++`



```
dsu@dsu-virtual-machine:~/work/01_opencv_imshow_example$ ./test
Hello World
image width = 559
image height = 376
█
```

예제) 09_opencv/01_image_center.cpp

무엇을 해볼까요?

- 아래의 이미지에서 중심점 위치는?
 - 중심 좌표의 pixel정보(color 값)를 얻어보자.
 - 중심좌표값
 - $C(x, y) = C(559/2, 376/2) = \mathbf{C(279, 188)}$
 - 중심좌표값의 Red : $\mathbf{(559*188+279)*3 + 0}$
 - 중심좌표값의 Green : $\mathbf{(559*188+279)*3 + 1}$
 - 중심좌표값의 Blue : $\mathbf{(559*188+279)*3 + 2}$
- **cv::Mat의 기본 color 순서는 BGR순서로 되어있음.**
 - 중심좌표값의 Red : $\mathbf{(559*188+279)*3 + 2}$
 - 중심좌표값의 Green : $\mathbf{(559*188+279)*3 + 1}$
 - 중심좌표값의 Blue : $\mathbf{(559*188+279)*3 + 0}$

```
$ gcc test.cpp -o test `pkg-config --cflags --libs opencv4` -lstdc++
```

무엇을 해볼까?

- 아래의 이미지에서 중심좌표값의 RGB 색

중심좌표값의 Red 위치 : $(559*188+279)*3 + 2$
중심좌표값의 Green 위치 : $(559*188+279)*3 + 1$
중심좌표값의 Blue 위치 : $(559*188+279)*3 + 0$



```
#include "opencv2/opencv.hpp"
#include <stdio.h>

int main(void)
{
    printf("Hello World\n");
    cv::Mat image = cv::imread("/home/dsu/work/01_opencv_imshow_example/test.jpg");

    int image_width = image.cols;
    int image_height = image.rows;
    int image_center_x = image_width/2;
    int image_center_y = image_height/2;

    int red_index = (image_width * image_center_y + image_center_x)*3 + 2;
    int green_index = (image_width * image_center_y + image_center_x)*3 + 1;
    int blue_index = (image_width * image_center_y + image_center_x)*3 + 0;

    int red_value = image.data[red_index];
    int green_value = image.data[green_index];
    int blue_value = image.data[blue_index];

    printf("red = %d\n", red_value);
    printf("green = %d\n", green_value);
    printf("blue = %d\n", blue_value);

    cv::imshow("test", image);
    cv::waitKey(0);

    return 0;
}
```

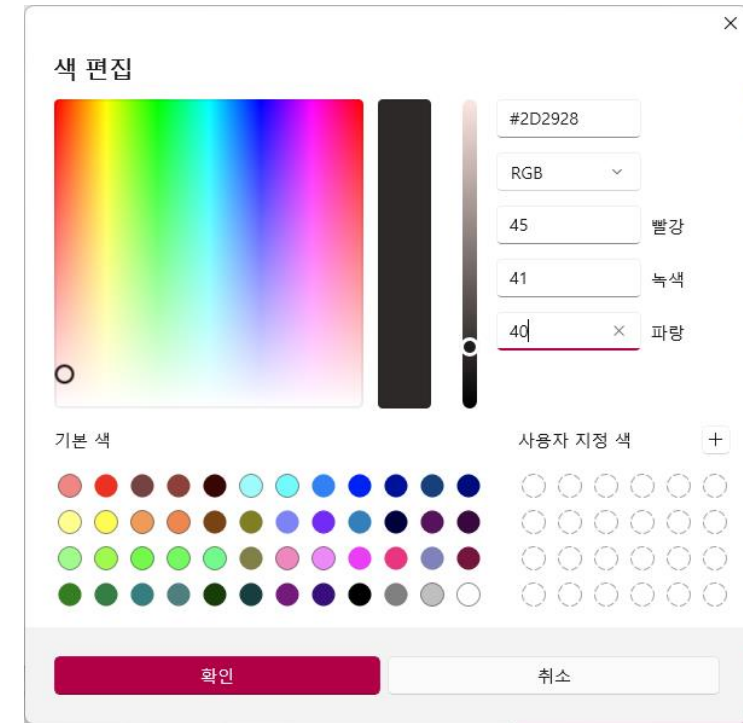
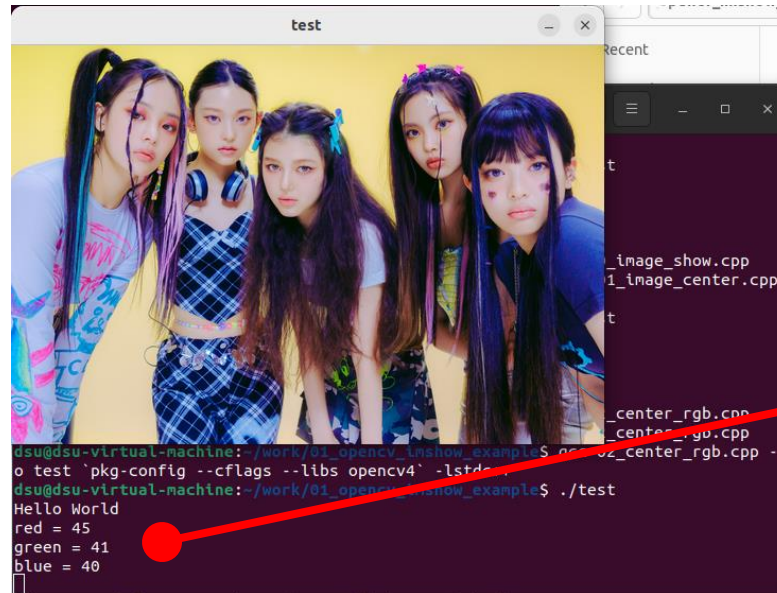
```
int red_value = image.data[red_index];
int green_value = image.data[green_index];
int blue_value = image.data[blue_index];
```

예제) 09_opencv/02_center_rgb.cpp

무엇을 해볼까요?

- 아래의 이미지에서 중심점 위치는?
 - 중심좌표값의 RGB 색상 값은?

중심좌표값의 Red 위치 : $(559*188+279)*3 + 2$
중심좌표값의 Green 위치 : $(559*188+279)*3 +$
중심좌표값의 Blue 위치 : $(559*188+279)*3 + 0$



```
dsu@dsu-virtual-m
Hello World
red = 45
green = 41
blue = 40
```

예제) 09_opencv/02_center_rgb.cpp

무엇을 해볼까요?

- C언어의 for 반복문을 이용하여
- 이미지의 모든 pixel정보를 얻어보자.



```
for( int y=0 ; y<image.rows ; y++ )  
{  
    for( int x=0 ; x<image.cols ; x++ )  
    {  
        //indexing  
    }  
}
```

예제) 09_opencv/03_for_indexing.cpp

무엇을 해볼까요?

- C언어의 for 반복문을 이용하여
- 이미지의 모든 pixel정보를 얻어보자.



중요 코드만 있습니다.
전문은 QR코드의 링크에서 확인

```
for( int y=0 ; y<image_height ; y++ )
{
    for( int x=0 ; x<image_width ; x++ )
    {
        //indexing
        int red_index = (image_width * y + x)*3 + 2 ;
        int green_index = (image_width * y + x)*3 + 1 ;
        int blue_index = (image_width * y + x)*3 + 0 ;

        int red_value = image.data[red_index] ;
        int green_value = image.data[green_index] ;
        int blue_value = image.data[blue_index] ;
        printf("(%d, %d) color = %d, %d, %d\n", x, y, red_value, green_value, blue_value) ;
    }
}
```

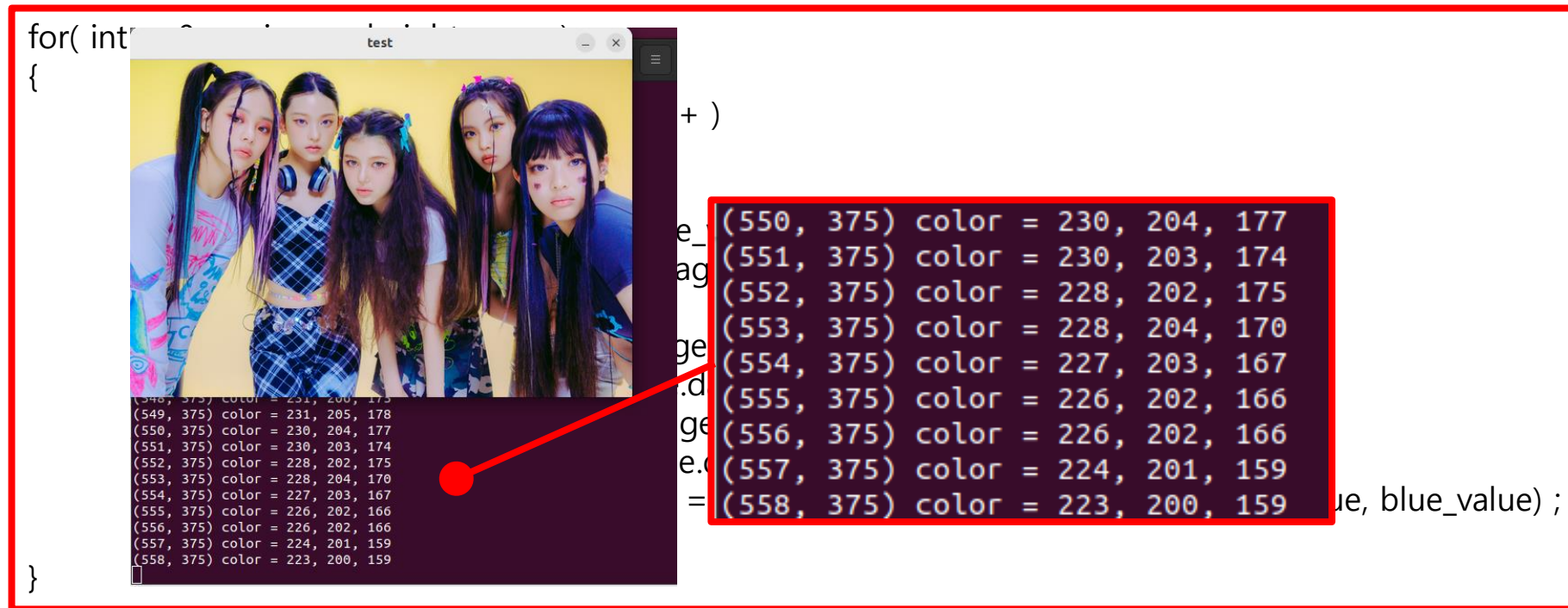
예제) 09_opencv/04_for_indexing_rgb.cpp

무엇을 해볼까요?

- C언어의 for 반복문을 이용하여
- 이미지의 모든 pixel정보를 얻어보자.



중요 코드만 있습니다.
전문은 QR코드의 링크에서 확인



예제) 09_opencv/04_for_indexing_rgb.cpp

무엇을 해볼까요?

- 반대로 C언어의 for 반복문을 이용하여
- 이미지의 모든 pixel정보를 넣어 보자(Write)

Color값을 Read

```
int red_value = image.data[red_index] ;  
int green_value = image.data[green_index] ;  
int blue_value = image.data[blue_index] ;
```

Color값을 Write

```
image.data[red_index] = red_value ;  
image.data[green_index] = green_value ;  
image.data[blue_index]
```


무엇을 해볼까요?

- 반대로 C언어의 for 반복문을 이용하여
- 이미지의 모든 pixel정보를 넣어 보자(Write)



중요 코드만 있습니다.
전문은 QR코드의 링크에서 확인

```
for( int y=0 ; y<image_height ; y++ )
{
    for( int x=0 ; x<image_width ; x++ )
    {
        //indexing
        int red_index = (image_width * y + x)*3 + 2 ;
        int green_index = (image_width * y + x)*3 + 1 ;
        int blue_index = (image_width * y + x)*3 + 0 ;

        image.data[red_index] = 255 ;    //red
        image.data[green_index] = 0;    //green
        image.data[blue_index] = 0;    //blue
    }
}
```

예제) 09_opencv/05_for_indexing_rgb_write.cpp

무엇을 해볼까요?

- 반대로 C언어의 for 반복문을 이용하여
- 이미지의 모든 pixel정보를 넣어 보자(Write)



중요 코드만 있습니다.
전문은 QR코드의 링크에서 확인

```
for
{
    dsu@dsu-virtual-machine: ~/work/01_opencv_imshow_example
    test
    3 + 2 ;
    x)*3 + 1 ;
    *3 + 0 ;
    d
    een
    ie
    dsu@dsu-virtual-machine:~/work/01_opencv_imshow_example$ gcc 05_for_indexing_rgb
    _write.cpp -o test `pkg-config --cflags --libs opencv4` -lstdc++
    dsu@dsu-virtual-machine:~/work/01_opencv_imshow_example$ ./test
    Hello World
}
```

예제) 09_opencv/05_for_indexing_rgb_write.cpp

무엇을 해볼까요?

- 반대로 C언어의 for 반복문을 이용하여
- 이미지의 모든 pixel정보를 넣어 보자(Write)



중요 코드만 있습니다.
전문은 QR코드의 링크에서 확인

```
for( int y=0 ; y<image_height ; y++ )
{
    for( int x=0 ; x<image_width ; x++ )
    {
        //indexing
        int red_index = (image_width * y + x)*3 + 2 ;
        int green_index = (image_width * y + x)*3 + 1 ;
        int blue_index = (image_width * y + x)*3 + 0 ;


        image.data[red_index] = 255 ;    //red
        image.data[green_index] = 0;    //green
        image.data[blue_index] = 0;    //blue

        cv::imshow("test", image) ;
        cv::waitKey(1) ;
    }
}
```

예제) 09_opencv/05_for_indexing_rgb_write_display.cpp

무엇을 해볼까요?

- 특정 색상(RGB(240,221,144))을 초록색(RGB(0,255,0))으로 바꿔 보자



색 편집

연한 노랑

#F0DD90

RGB

240 빨강

221 녹색

144 파랑

무엇을 해볼까요?

- 특정 색상(**RGB(240,221,144)**)을 초록색(**RGB(0,255,0)**)으로 바꿔 보자



중요 코드만 있습니다.
전문은 QR코드의 링크에서 확인

```
for( int y=0 ; y<image_height ; y++ )
{
    for( int x=0 ; x<image_width ; x++ )
    {
        //indexing
        int red_index = (image_width * y + x)*3 + 2 ;
        int green_index = (image_width * y + x)*3 + 1 ;
        int blue_index = (image_width * y + x)*3 + 0 ;

        int red_value = image.data[red_index] ;
        int green_value = image.data[green_index] ;
        int blue_value = image.data[blue_index] ;

        if( red_value == 240 && green_value == 221 && blue_value == 144 )
        {
            image.data[red_index] = 0 ;           //red
            image.data[green_index] = 255 ;       //green
            image.data[blue_index] = 0 ;          //blue
        }
    }
}
```

예제) 09_opencv/05_for_indexing_rgb_write_display.cpp

무엇을 해볼까요?

- 특정 색상영역(**RGB(240,221,144)**)을 초록색(**RGB(0,255,0)**)으로 바꿔보자



```
if( red_value > 200 && red_value < 260 &&
    green_value > 200 && green_value < 240 &&
    blue_value > 100 && blue_value < 170 )
{
    image.data[red_index] = 0 ;      //red
    image.data[green_index] = 255 ; //green
    image.data[blue_index] = 0 ;     //blue
}
```

중요 코드만 있습니다.
전문은 QR코드의 링크에서 확인

예제) 09_opencv/06_for_indexing_find_and_change_color.cpp

무엇을 해볼까요?

- 특정 색상영역(**RGB(240,221,144)**)을 초록색(**RGB(0,255,0)**)으로 바꿔보자



중요 코드만 있습니다.
전문은 QR코드의 링크에서 확인

```
if( red_value > 200 && red value < 260 &&  
gr  
bl  
{  
im  
im  
im  
}  
&&
```

Two side-by-side windowed images. The left window, titled 'ori', shows a group of five young women with long hair against a yellow background. The right window, titled 'test', shows the same group of women, but the background has been changed to a solid green color, and some green speckles are visible on their clothing.

예제) 09_opencv/06_for_indexing_find_and_change_color2.cpp

무엇을 해볼까요?

- 특정 색상영역(**RGB(240,221,144)**)을 초록색(**RGB(0,255,0)**)으로 바꿔보자



중요 코드만 있습니다.
전문은 QR코드의 링크에서 확인



260 &&

&&

방송 크로마키에서 사용



예제) 09_opencv/06_for_indexing_find_and_change_color2.cpp