

객체지향 프로그래밍 실습 4

2019. 3. 29 (금)

다음은 색의 3요소인 red, green, blue로 색을 추상화한 Color 클래스를 선언하고 활용하는 코드이다. 빈칸을 채우시오. 단, red, green, blue는 0~255의 값만 가진다.

```
#include <iostream>

using namespace std;

class Color {

    int red, green, blue;

public:

    Color() { red = green = blue = 0; }

    Color(int r, int g, int b) { red = r; green = g; blue = b; }

    void setColor(int r, int g, int b) { red = r; green = g; blue = b; }

    void show() { cout << red << ' ' << green << ' ' << blue << endl; }

};

int main() {

    Color screenColor(255, 0, 0); // 빨간색의 screenColor 객체 생성

    Color *p;          // Color 타입의 포인터 변수 p 선언

    (                ) // (1) p가 screenColor의 주소를 가지도록 한다

    (                ) // (2) p와 show를 이용하여 screenColor 색 출력

    (                ) // (3) Color의 일차원 배열 colors 선언. 원소는 3개

    (                ) // (4) p가 colors 배열을 가리키도록 한다.

    // (5) p와 setColor()를 이용하여 colors[0], colors[1], colors[2]가 각각 빨강, 초록, 파랑을 가지도록 함

    (                )

    (                )

    (                )

    // (6) p와 show()를 이용하여 colors 배열의 모든 객체의 색 출력. for 문 이용.

    (                )

    (                )

    (                )

}
```

실행 결과:

255 0 0

255 0 0

0 255 0

0 0 255