

[과제1] OpenCV를 이용한 간단한 그림판 만들기

20210985 노하림

<코드 및 설명>

```
import cv2 as cv
import numpy as np
import math
import sys

# 1. 600*900 크기의 컬러 이미지를 만들어 흰색으로 칠하시오.
white = (255, 255, 255) # 흰색
img = np.full((900, 600, 3), white, dtype=np.uint8) # 흰색 배경 생성

BrushSiz=5 # 반지름이 5
LColor,RColor=(255,0,0),(0,0,255) # 왼쪽, 오른쪽 버튼 누를 때 색

# 예외처리
if img is None:
    sys.exit('파일을 찾을 수 없습니다.')

# 콜백 함수
def draw(event, x, y, flags, param):
    global ix, iy

    # 2. Shift 와 마우스 왼쪽 버튼의 다운/업을 이용하여 직선을 그리시오.
    # Shift 키가 눌러있는지 확인
    if flags & cv.EVENT_FLAG_SHIFTKEY:

        # 마우스 왼쪽 다운 확인
        if event == cv.EVENT_LBUTTONDOWN:
            ix, iy = x, y

        # 마우스 왼쪽 업 확인
        elif event == cv.EVENT_LBUTTONUP:
            # 파랑색 선 그리기
            cv.line(img, (ix, iy), (x, y), (255, 0, 0), 2)

    # 3. Alt 와 마우스 왼쪽 버튼의 다운/업을 이용하여 직사각형을 그리시오.
    # 4. Alt 와 마우스 오른쪽 버튼의 다운/업을 이용하여 내부가 칠해진 직사각형을 그리시오.
    # Alt 키가 눌러있는지 확인
    elif flags & cv.EVENT_FLAG_ALTKEY:
```

마우스 왼쪽 다운 확인

if event == cv.EVENT_LBUTTONDOWN:

ix, iy = x, y

마우스 왼쪽 업 확인

elif event == cv.EVENT_LBUTTONUP:

빨간색 사각형 그리기

cv.rectangle(img, (ix, iy), (x, y), (0, 0, 255), 2)

마우스 오른쪽 다운 확인

elif event == cv.EVENT_RBUTTONDOWN:

ix, iy = x, y

마우스 오른쪽 업 확인

elif event == cv.EVENT_RBUTTONUP:

빨간색 내부가 칠해진 사각형 그리기

cv.rectangle(img, (ix, iy), (x, y), (0, 0, 255), -1)

5. Ctrl 와 마우스 왼쪽 버튼의 다운/업을 이용하여 원을 그리시오.

6. Ctrl 와 마우스 오른쪽 버튼의 다운/업을 이용하여 내부가 칠해진 원을 그리시오.

Ctrl 키가 눌러있는지 확인

elif flags & cv.EVENT_FLAG_CTRLKEY:

마우스 왼쪽 다운 확인

if event == cv.EVENT_LBUTTONDOWN:

ix, iy = x, y

마우스 왼쪽 업 확인

elif event == cv.EVENT_LBUTTONUP:

반지름 두 점의 거리를 계산

radius = int(math.sqrt((x - ix) ** 2 + (y - iy) ** 2))

초록색 원 그리기

cv.circle(img, (ix, iy), radius, (0, 255, 0), 2)

마우스 오른쪽 다운 확인

elif event == cv.EVENT_RBUTTONDOWN:

ix, iy = x, y

마우스 오른쪽 업 확인

```
elif event == cv.EVENT_RBUTTONDOWN:
```

```
    # 반지름 두 점의 거리를 계산
```

```
    radius = int(math.sqrt((x - ix) ** 2 + (y - iy) ** 2))
```

```
    # 초록색 내부가 칠해진 원 그리기
```

```
    cv.circle(img, (ix, iy), radius, (0, 255, 0), -1)
```

```
# 7. 마우스 왼쪽 버튼을 누르면서 움직이면 파란색 원(반지름 5)이 따라 그려진다.
```

```
# 마우스 왼쪽이 눌린채로 이동 확인
```

```
elif event == cv.EVENT_MOUSEMOVE and flags == cv.EVENT_FLAG_LBUTTON:
```

```
    # 반지름이 5 인 파란색 원 생성
```

```
    cv.circle(img, (x, y), BrushSiz, LColor, -1)
```

```
# 8. 마우스 오른쪽 버튼을 누르면서 움직이면 빨간색 원(반지름 5)이 따라 그려진다.
```

```
# 마우스 오른쪽이 눌린채로 이동 확인
```

```
elif event == cv.EVENT_MOUSEMOVE and flags == cv.EVENT_FLAG_RBUTTON:
```

```
    # 반지름이 5 인 빨간색 원 생성
```

```
    cv.circle(img, (x, y), BrushSiz, RColor, -1)
```

```
# 수정된 이미지를 다시 그림
```

```
cv.imshow('Drawing', img)
```

```
cv.namedWindow('Drawing')
```

```
cv.imshow('Drawing', img)
```

```
cv.setMouseCallback('Drawing', draw) # Drawing 윈도우에 draw 콜백 함수 지정
```

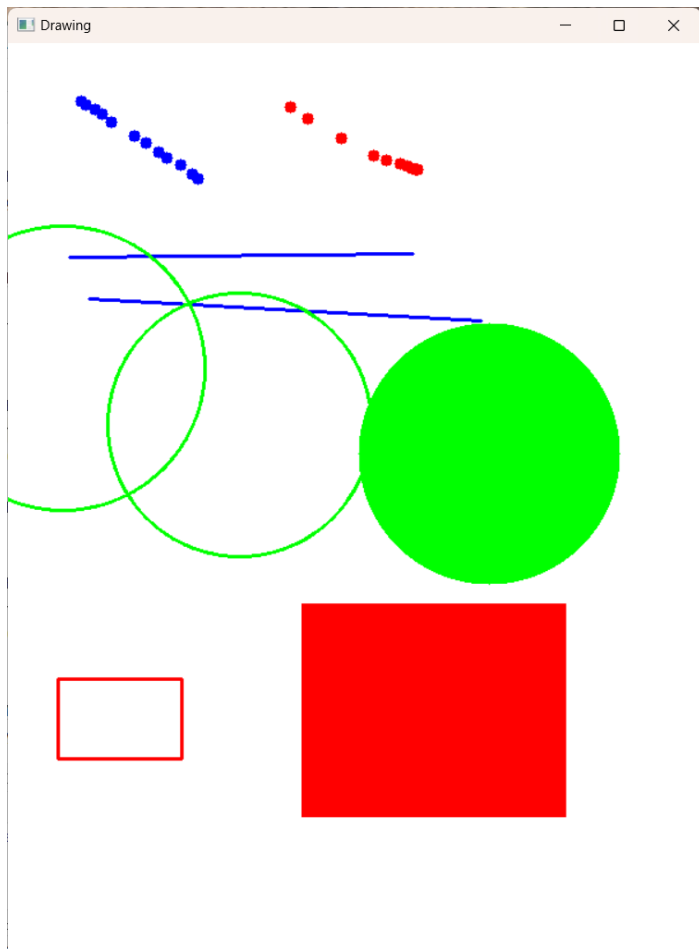
```
while (True): # 마우스 이벤트가 언제 발생할지 모르므로 무한 반복
```

```
    if cv.waitKey(1) == ord('q'): # 키보드 입력을 확인하여 'q' 키가 눌리면 루프를 종료
```

```
        cv.destroyAllWindows() # 창 닫기
```

```
        break
```

<결과물>



- shift와 마우스 왼쪽을 누르면 직선이 그려짐
- alt와 마우스 왼쪽 버튼을 누르면 직사각형, 오른쪽 버튼을 누르면 채워진 직사각형이 그려짐
- ctrl과 마우스 왼쪽 버튼을 누르면 원, 오른쪽 버튼을 누르면 채워진 원이 그려짐
- 마우스 왼쪽 버튼을 누르면서 움직이면 파란색 원이, 오른쪽 버튼을 누르면서 움직이면 빨간색 원이 그려짐