*HW6 과제 – 5/ 10*

**주제 :**

**Turtle을 이용한 사각형과 그림 그리기**

**이름 : 김민석**

**학번 : 20195125**

**메일 : planetside2@naver.com**

**소속 : 소프트웨어융합**

**과목 : 컴퓨터사고와 문제해결**

**목차**

1. **첫번째 코드**
2. **실행화면 사진**
3. **내용 설명**
4. **두번째 코드**
5. **실행화면 사진**
6. **내용 설명**
7. **첫 번째 코드**
8. **Main.py**

import turtle

from myrect import MyGoto, MyRect

quad = turtle.Turtle()

quad.speed(10)

MyRect(quad, -100, 100, 80)

1. **Myrect.py**

def MyGoto(quad, x\_xx, x\_yy):

quad.penup()

quad.goto(x\_xx, x\_yy)

quad.pendown()

def MyRect(quad, x\_xx, x\_yy, x\_length):

MyGoto(quad, x\_xx,x\_yy)

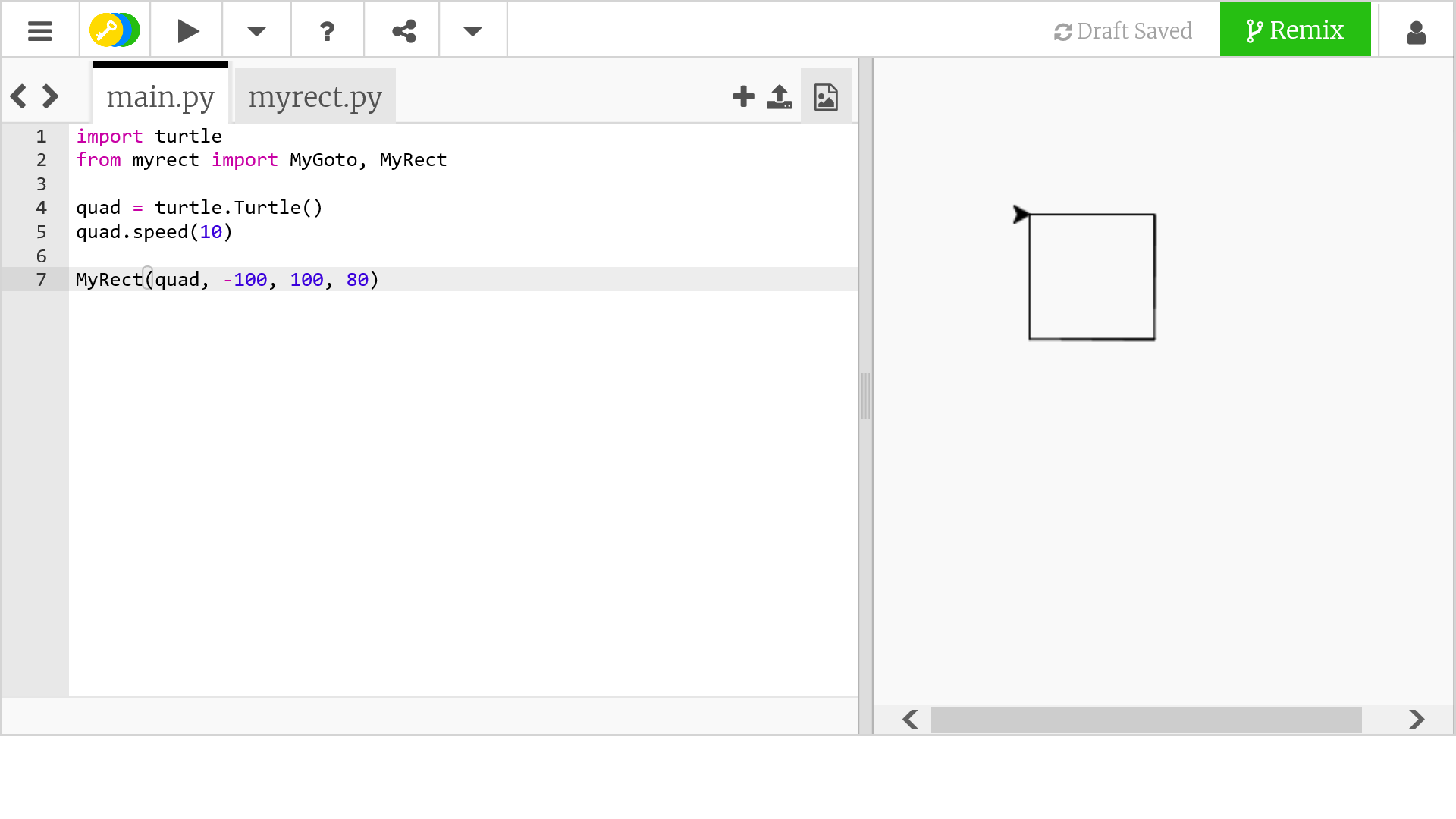
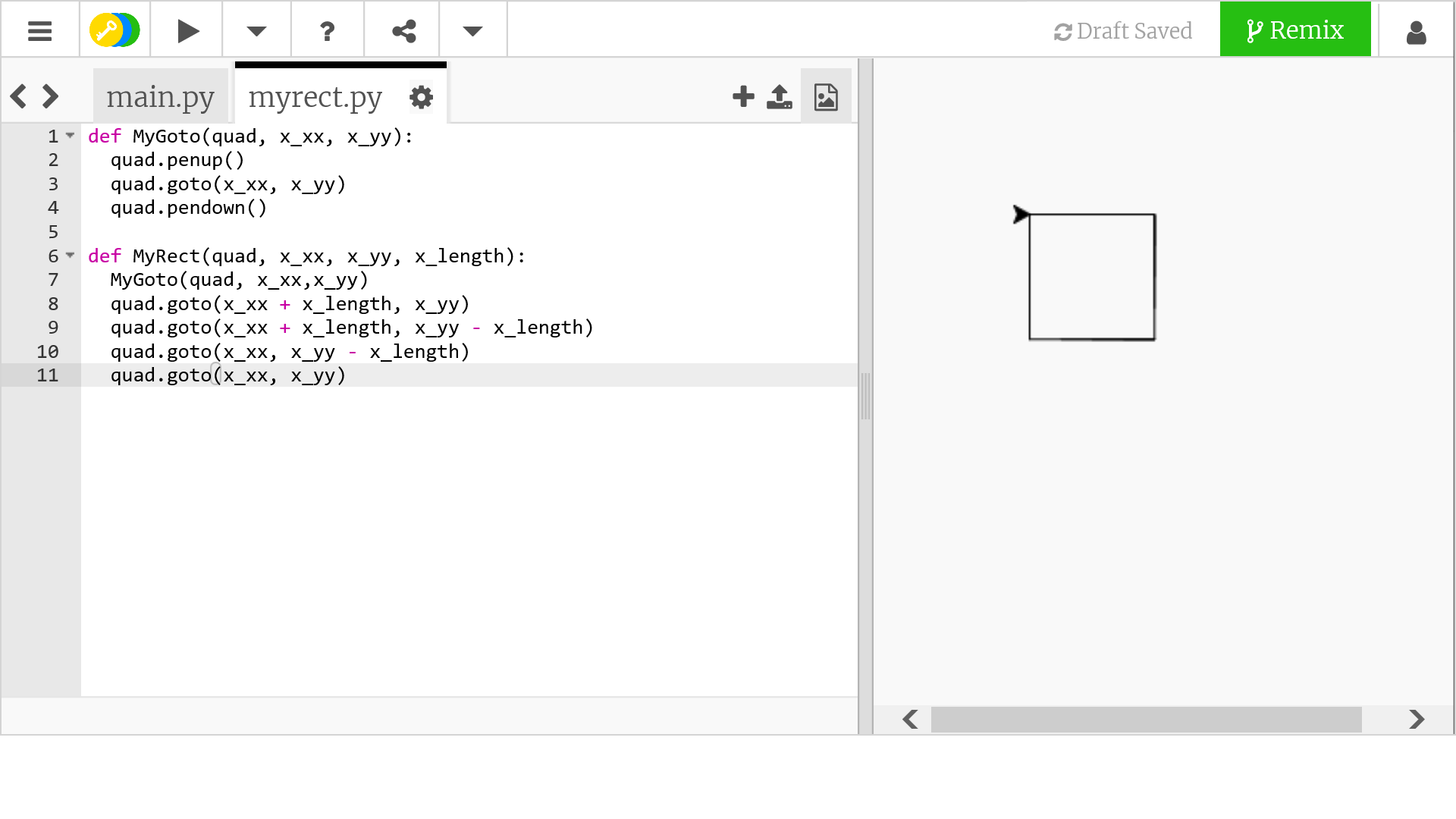
quad.goto(x\_xx + x\_length, x\_yy)

quad.goto(x\_xx + x\_length, x\_yy - x\_length)

quad.goto(x\_xx, x\_yy - x\_length)

quad.goto(x\_xx, x\_yy)

1. **실행화면 사진**

1. **구현내용 설명**

**1. import turtle**로 **turtle** 상자를 불러온다.

2. quad = turtle.Turtle()로 사각형을 그릴 수 있게 준비하고 quad.speed(10)으로 엄청난 속도로 사각형을 그리게 한다.

3. 우선 myrect 라는 새 py파일(?) 을 만든다. 그릴 때 단순하게 forward, left를 쓰지않고 goto 함수를 이용한 사각형을 만들기 위해서 **def MyGoto(x\_xx, x\_yy): ~~** 로 만든다. 그다음에 **myrect.py**에 보낼 때 대상도 같이 보내야 함으로

**def MyGoto(quad, x\_xx, x\_yy):**

**quad.penup()**

**quad.goto(x\_xx, x\_yy)**

**quad.pendown()** 로 **myrect.py** 에 보내서 **MyGoto** 함수를 쓸 수 있게 한다.

4. 다음으로 **MyGoto** 함수와 같이 **quad** 도 같이 보내면서

**def MyRect(quad, x\_xx, x\_yy, x\_length):** 로 사각형을 그릴 함수를 만들기 시작한다.

5. **MyGoto(quad, x\_xx, x\_yy)** 로 맨 처음 사각형을 그릴 때의 좌표를 받을 수 있게 해준 후, **quad.goto(~) 로 forward 와 left(right)** 대신에 이동하면서 선을 그릴 수 있게 해준다.

6. 사각형을 그릴 때 **goto함수**는 현재 화살표의 좌표에서 **goto 함수에** 입력된 좌표로 이동하면서 선이 그려지므로

**quad.goto(x\_xx + x\_length, x\_yy)**

**quad.goto(x\_xx + x\_length, x\_yy - x\_length)**

**quad.goto(x\_xx, x\_yy - x\_length)**

**quad.goto(x\_xx, x\_yy)**

로 첫번째 좌표 **x\_xx, x\_yy**에서 **x 좌표**가 **x\_length**만큼 변하고, **y좌표**가 **x\_length**만큼 변할 수 있게 해준다. 이와 같은 **goto 함수를 4번써서 forward**와 **left** 함수를 안쓰고도 **사용자가 원하는 좌표에서 원하는 길이만큼의 사각형을 그릴 수 있게 해준다.**

**7. from myrect import MyGoto, MyRect** 로 main.py에서 myrect.py를 불러서 myrect에 있는 함수(**MyGoto와 MyRect)를** 쓸 수 있게 한다.

8. 마지막으로 **main.py** 에서 **MyRect(quad, -100, 100, 80)**을 입력함으로써 **quad를** **myrect.py**에서 인식할 수 있도록 보내준 후 **(-100,100)의 좌표**로 가서 한 변의 길이가 **80인 사각형**을 그리도록 해준다.

1. **두 번째 코드**
2. **Main.py**

import turtle

from smile import \*

mycircle = turtle.Turtle()

mycircle.speed(10)

DrawCircle(mycircle, 0, -100, 150, 'green')

DrawCircle(mycircle, -40, 60, 30, 'white')

DrawCircle(mycircle, -40, 60, 10, 'black')

DrawCircle(mycircle, 50, 60, 40, 'white')

DrawCircle(mycircle, 50, 60, 10, 'black')

DrawCircle(mycircle, 0, 20, 10, 'black')

DrawCircle(mycircle, 0, -80, 40, 'brown')

1. **smile.py**

def MyGoto(mycircle, x\_xx, x\_yy):

mycircle.penup()

mycircle.goto(x\_xx, x\_yy)

mycircle.pendown()

def DrawCircle(mycircle, x\_xx, x\_yy, x\_radius, x\_color):

MyGoto(mycircle, x\_xx, x\_yy)

mycircle.color(x\_color)

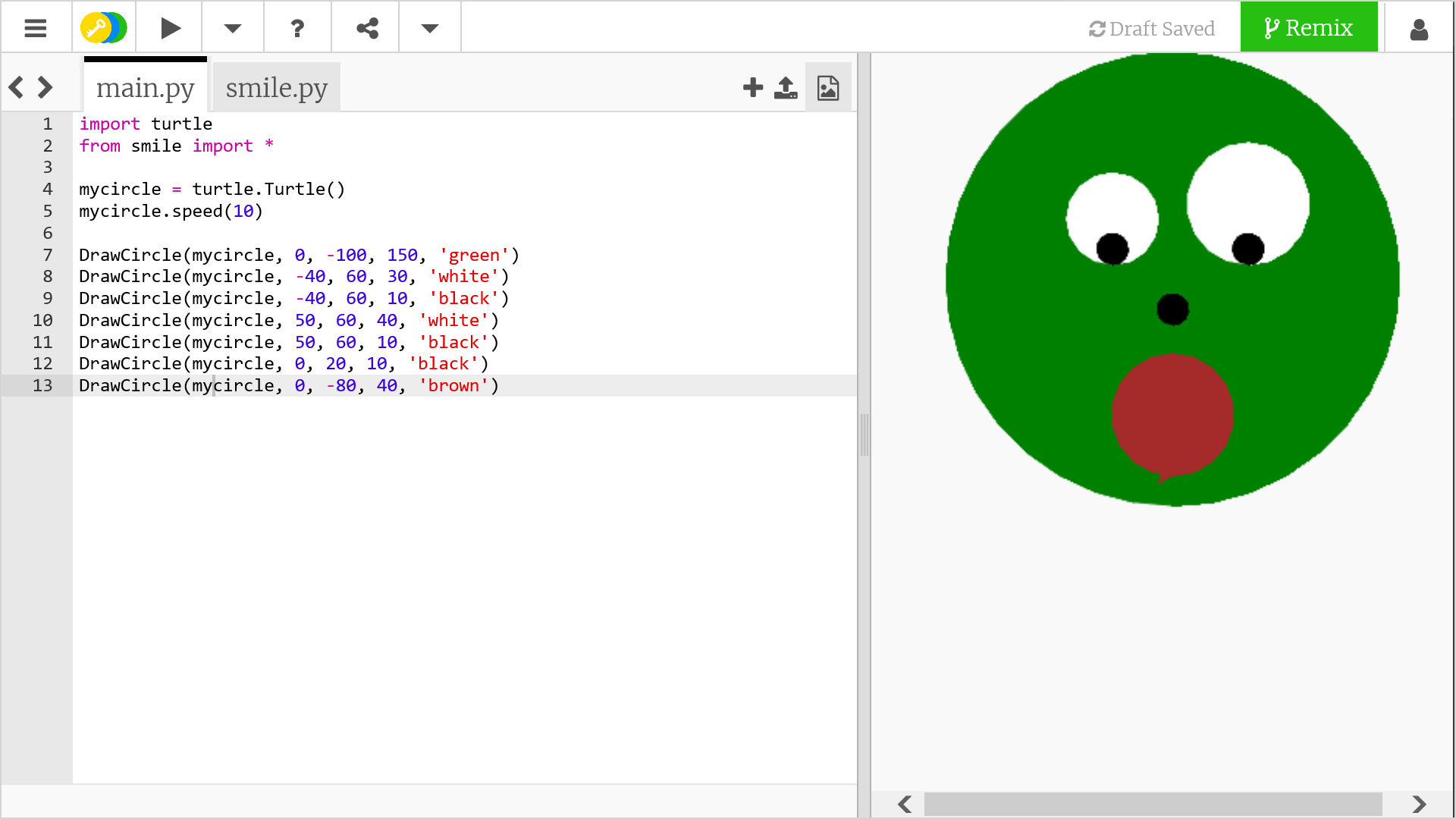
mycircle.fillcolor(x\_color)

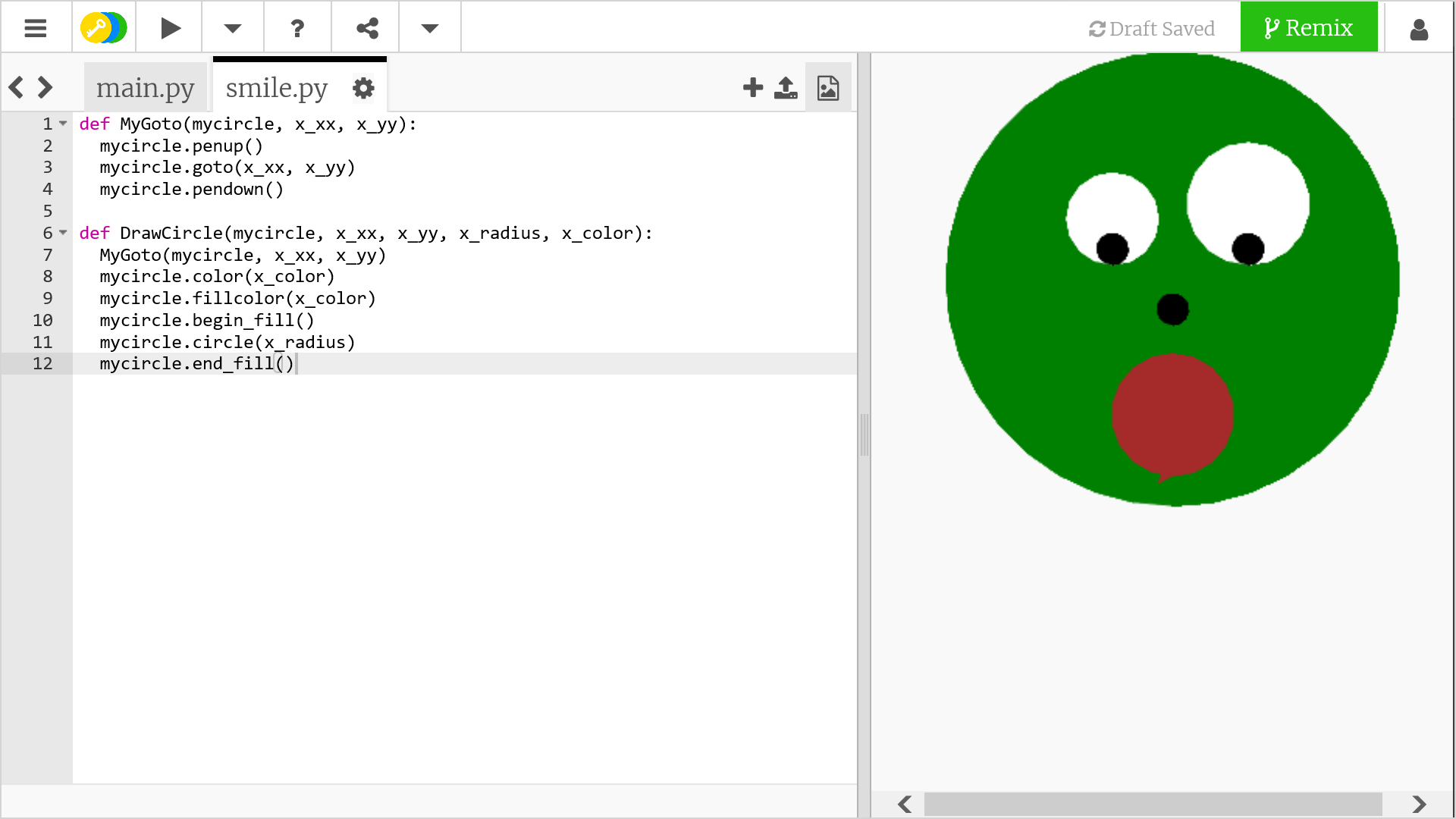
mycircle.begin\_fill()

mycircle.circle(x\_radius)

mycircle.end\_fill()

1. **실행화면 사진**





1. **구현내용 설명**

**1**. **import turtle**로 **turtle** 상자를 불러온다.

**2**. **mycircle = turtle.Turtle()**로 그림을 그릴 수 있게 하고 그림을 그리는 선의 속도를 **mycircle.speed(5)** 로 빠르게 그릴 수 있게 만든다.

**3.** 먼저 좌표를 이동시킬 수 있는 함수 **MyGoto**를

**def MyGoto(x\_xx, x\_yy): ~~** 로 만든다. 그다음에 **smile.py**에 보낼 때 대상도 같이 보내야 함으로

**def MyGoto(mycircle, x\_xx, x\_yy):**

**mycircle.penup()**

**mycircle.goto(x\_xx, x\_yy)** #펜을 올리고 그 좌표로 이동할 때 선이 안보이게 이동할 수 있게 한다.

**mycircle.pendown()**

**로 smile.py**에 넣는다.

**4.** 좌표 이동함수 뿐만 아니라 **원을 그릴 수 있는 함수**도 만들어야 되므로 4번과 같이 **def DrawCircle(mycircle, x\_xx, x\_yy, x\_radius, x\_color): 로 좌표(x, y)값과 원의 반지름, 그리고 원을 채울 색상을 정할 수 있는 함수를 smile.py**에 만든다.

**5**. **짠 MyGoto 함수를 DrawCircle** 함수에 넣음으로써 원을 그릴 때 사용자가 원하는 좌표에서 그릴 수 있도록 만들어준다.

**6.** **mycircle.color(x\_color)와 mycircle.fillcolor(x\_color) 로** 그림을 그리는 화살표와 원을 채우는 색상을 사용자가 정하고 같게 만들어준다.

**7**. **mycircle.begin\_fill()**

**mycircle.circle(x\_radius)**

**mycircle.end\_fill()**

로 원을 그리면서 그 안을 바로 위에서 사용자가 정한 색으로 채워나갈 수 있게 쓴다. 이때 **begin\_fill**이나 **end\_fill** 둘 중 하나가 없거나 둘 다 없을 때는 선만 그린다.

8. **from smile import \*** 로 원과 좌표를 이동시킬 수 있는 함수를 모아둔 **smile.py**를 부른다. 그리고 \* 대신에 불러오는 곳의 **함수 이름을 써도 된다.** (**MyGoto, DrawCircle 등**. **\*** 은 **불러오는 곳의 모든 함수를 쓴다는 뜻**이다.)

(+)mysmile.py라고 써야 되는데 복습할 겸 그전 수업시간 때 코드를 이어서 썼더니 smile.py라고 쓰게 되었다..

(+)list = [~]를 써서 2번에서 색을 list [0~4]에서 불러와서 쓸 수 있었다.