문제1

<코드>

import turtle

import math

t = turtle.Turtle()

def MyCircle(r):

t.pencolor('blue')

n = 200

for i in range(n):

t.forward(math.sin(math.radians(180/n))\*r\*2)

t.right(360/n)

t.pencolor('red')

t.circle(100,360)

MyCircle(100)

<실행화면>



<구현내용 설명>

매개변수로 원의 반지름(r)을 받는 함수입니다.

이 함수는 n각형의 n이 무한으로 갈수록 원에 가까워지는 것을 이용한 함수입니다. (여기선200각형)

n각형의 한 변을 밑변(x)으로하고, 원의 중심을 한 꼭지점으로 하는 이등변 삼각형을 그리고, 원의중심에서 밑변에 수선의 발을 그리면 사인공식에 의해 sin(180/n)=(x/2)/r 이 성립되고 x에대해 정리하면 x=sin(180/n)\*2r이라는 n각형의 한 변의 길이가 나옵니다.

정n각형의 각도인 360/n으로 회전합니다.

문제2-1

<코드>

import math

sum = 0

step = 10000

for i in range(360):

for j in range(step):

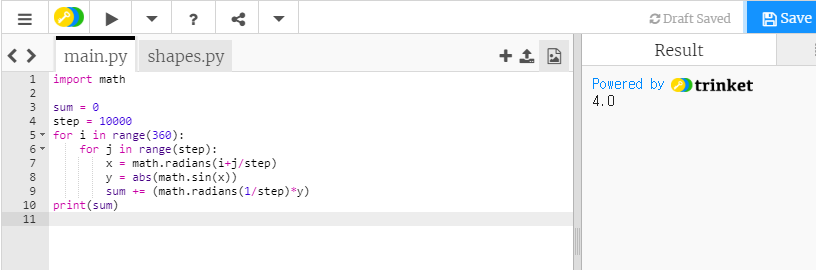
x = math.radians(i+j/step)

y = abs(math.sin(x))

sum += (math.radians(1/step)\*y)

print(sum)

<실행화면>



<구현내용 설명>

Sin함수 그래프를 무수히 많은 직사각형으로 나누어서 더함으로써 넓이를 구했습니다.

X축의 1라디안을 step으로 나눈 값을 X, X를 sin함수에 대입했을 때 값을 Y라 했을 때

X\*Y는 무수히 많은 직사각형 중 하나입니다. 이 직사각형의 넓이를 모두 합하면 sin함수의 넓이가 나옵니다.

sin함수는 pi~2pi는 음수값이므로 abs함수를 이용해 절대값으로 계산했습니다.

문제2-2

<코드>

import math

import numpy as np

step= 0.00001

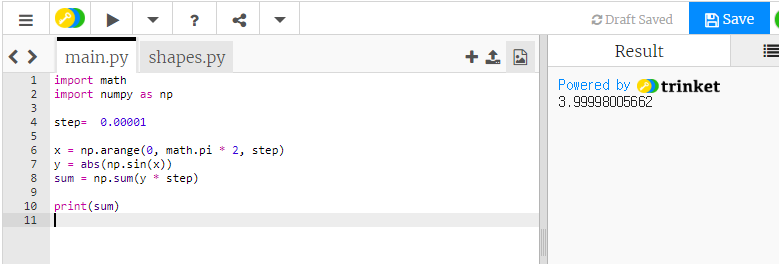
x = np.arange(0, math.pi \* 2, step)

y = abs(np.sin(x))

sum = np.sum(y \* step)

print(sum)

<실행화면>



<구현내용 설명>

np.arrange 함수를 이용해서 0부터 2pi까지 step만큼 증가하면서 생성된 값들을X, X에 있는 값들을 sin 함수에 대입한 것들을 Y로 만들고, X값과 Y값을 곱한 값들의 합이 sin함수의 넓이가 됩니다.

sin함수는 pi~2pi는 음수값이므로 abs함수를 이용해 절대값으로 계산했습니다.

문제3

<코드>

maxY=0;

maxX=0;

for i in range(15000):

x=i/1000

y=x\*((15-x)/2)

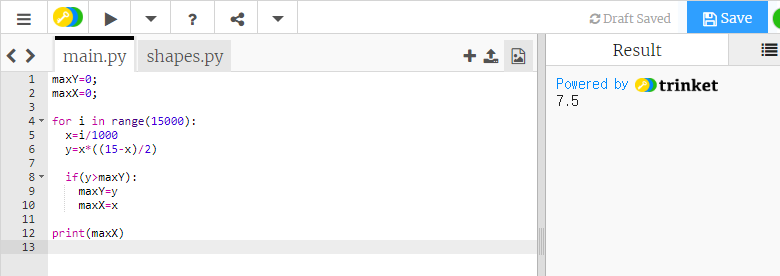
if(y>maxY):

maxY=y

maxX=x

print(maxX)

<실행화면>



<구현내용설명>

벽면과 평행인 변의 길이를 X, 울타리 넓이를 Y라 했을 때 y=x\*((15-x)/2) 라는 식이 성립됩니다

울타리의 넓이가 최대일 때의 x,y 를 maxX,maxY라 저장합니다.

x값을 1/1000씩 증가시키면서 y가 이전의 maxY값과 비교하여 y가 더 클 때 maxY에 현재 y의 값을 저장하고, maxX에는 현재 x의 값을 저장합니다.