```
import tkinter
import math

class Window:
    def __init__(self, xdim, ydim):
        self.root = tkinter.Tk()
        self.xdim = xdim
        self.ydim = ydim
        self.canvas = tkinter.Canvas(self.root, bg="white", height=ydim, width=xdim)
        self.canvas.pack()

    def drawPixel(self, x, y, colour):
        self.canvas.create_rectangle((x, y)*2, outline=colour)

    def sketchFunc(self, palette, iterations):
        for y in range(self.ydim):
            for x in range(self.xdim):
                colour = palette(x,y, iterations)
                self.drawPixel(x, y, colour)

def update(win, palette, iterations):
    win.canvas.delete("all")
    win.sketchFunc(palette, iterations)

resolutionx = 700
resolutiony = 400

iterations = [[0 for x in range(resolutionx)] for y in range(resolutiony)]
maxIterations = 100

def palette(x,y,iterations):
    i = iterations[y][x]
    if i == 0:
        return "white smoke"
    elif i < 100:
        return "grey"+str(100-i)
    else:
        return "black"

def xoff(x):
    #return x/200 - 2.5
    return x/2500 - 0.25

def yoff(y):
    #return x/200 - 1
    return y/2500 - 1

for y in range(resolutiony):
    for x in range(resolutionx):
        iteration = 0
        x0 = xoff(x)
        y0 = yoff(y)
        xn, yn = 0, 0
        while math.pow(x0,2)+math.pow(y0,2)<math.pow(2.2,2) and iteration <
maxIterations:
            #print(iteration)
            try:
                xtemp = math.pow(xn,2)-math.pow(yn,2) + x0
                yn = 2*xn*yn + y0
```

```
                xn = xtemp
                iteration += 1
            except:
                break
        #if iteration > 0:
        #    print(str(x)+","+str(y))
        iterations[y][x] = iteration
    if y%10==0:
        print(y)

win = Window(resolutionx,resolutiony)
#update(win, [[lambda x, y: iterations[int(y)][int(x)]==0,"white smoke"],[lambda x, y: iterations[int(y)][int(x)] in range(1,100),"grey"+str(100-iterations[int(y)][int(x)])],["black"]])
update(win, palette, iterations)
#print(iterations[0:10])
win.root.mainloop()
```

```
import tkinter
import math

class Window:
    def __init__(self, xdim, ydim):
        self.root = tkinter.Tk()
        self.xdim = xdim
        self.ydim = ydim
        self.canvas = tkinter.Canvas(self.root, bg="white", height=ydim, width=xdim)
        self.canvas.pack()

    def drawPixel(self, x, y, colour):
        self.canvas.create_rectangle((x, y)*2, outline=colour)

    def sketchFunc(self, palette, iterations):
        for y in range(self.ydim):
            for x in range(self.xdim):
                self.drawPixel(x, y, palette(x, y, iterations))

resolutionx = 1400
resolutiony = 800

iterations = [[0 for x in range(resolutionx)] for y in range(resolutiony)]
global maxIterations
maxIterations = 500

def update(win, palette, iterations):
    win.canvas.delete("all")
    win.sketchFunc(palette, iterations)

def palette(x,y,iterations):
    global maxIterations
    colours = ["navy","medium blue","royal blue","cornflower blue","light sky blue","LightSkyBlue2","ghost white","lemon chiffon","light goldenrod yellow","light goldenrod","black"]
    i = iterations[y][x]
    if i == 0:
        return colours[0]
    base = math.pow(maxIterations,1/(len(colours)-1))
    return colours[math.floor(math.log(i,base))]

def xoff(x):
    return x/500 - 2.5
    #return x/5000 - 0.4

def yoff(y):
    return y/500 - 1
    #return y/5000 - 0.7

for y in range(resolutiony):
    for x in range(resolutionx):
        iteration = 0
        x0 = xoff(x)
        y0 = yoff(y)
        xn, yn = 0, 0
        while math.pow(x0,2)+math.pow(y0,2)<math.pow(2.2,2) and iteration < maxIterations:
            try:
                xtemp = math.pow(xn,2)-math.pow(yn,2) + x0
```

```
                yn = 2*xn*yn + y0
                xn = xtemp
                iteration += 1
            except:
                break
        iterations[y][x] = iteration
    if y%100==0:
        print(y)

win = Window(resolutionx,resolutiony)
update(win, palette, iterations)
win.root.mainloop()

```