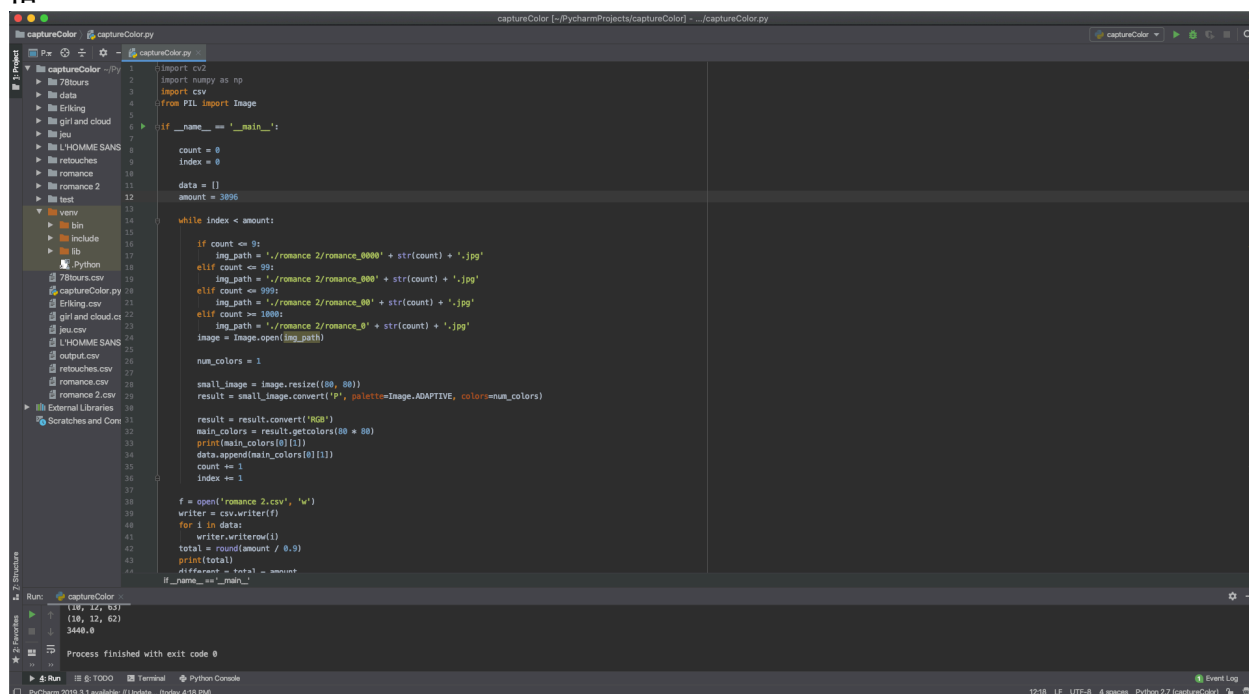


首先使用 Python 识别每一帧图片的每一个像素点的颜色，计算出每一帧图片里主要的颜色，并输出颜色的 RGB 值。分析完每一帧图片的颜色后，把所有的颜色写入一个.csv 表格。



```
captureColor.py
1 import cv2
2 import numpy as np
3 import csv
4 from PIL import Image
5
6 if __name__ == '__main__':
7     count = 0
8     index = 0
9
10    data = []
11    amount = 3896
12
13    while index < amount:
14
15        if count <= 9:
16            img_path = './romance 2/romance_0000' + str(count) + '.jpg'
17            count += 1
18        elif count <= 99:
19            img_path = './romance 2/romance_000' + str(count) + '.jpg'
20            count += 1
21        elif count <= 999:
22            img_path = './romance 2/romance_00' + str(count) + '.jpg'
23            count += 1
24        elif count <= 1000:
25            img_path = './romance 2/romance_0' + str(count) + '.jpg'
26            image = Image.open(img_path)
27
28            num_colors = 1
29
30            small_image = image.resize((88, 88))
31            result = small_image.convert('P', palette=Image.ADAPTIVE, colors=num_colors)
32
33            result = result.convert('RGB')
34            main_colors = result.getcolors(88 * 88)
35            print(main_colors[0][1])
36            data.append(main_colors[0][1])
37            count += 1
38            index += 1
39
40    f = open('romance 2.csv', 'w')
41    writer = csv.writer(f)
42    for i in data:
43        writer.writerow(i)
44    total = round(amount / 0.9)
45    print(total)
46    difference = total - amount
47    if __name__ == '__main__':
```

Run: captureColor.py

```
10, 12, 63
10, 12, 62
3440, 0
```

Process finished with exit code 0

(Python 程序)

The screenshot shows a web application interface. At the top, there's a header bar with a title "78tours" and various icons for View, Zoom, Add Category, Insert, Table, Chart, Text, Shape, Media, Comment, Collaborate, Format, and Organize. Below the header, there's a "Sheet 1" tab. The main content area displays a table with the following data:

R	G	B
71	74	38
70	74	36
71	75	37
70	75	37
70	75	36
70	75	36
70	75	36
70	75	36
70	75	36
70	75	36
70	75	36
70	75	36
71	75	36
70	75	35
70	75	35
70	74	35
70	74	35
70	74	35
71	74	35
70	74	34
70	74	35
70	74	35
70	74	35
71	75	35
71	75	34
71	75	34
71	75	35
71	75	35
71	74	35
71	74	35
71	75	35
70	75	35
70	75	35
70	75	36
70	75	36
68	73	34
68	73	34
68	72	34

On the right side of the interface, there's a sidebar with a dark background. It contains a message: "Nothing selected. Select an object to format." and a paintbrush icon.

(.csv 表格)

使用 JavaScript 和 d3 库，读取.csv 表格。根据.csv 文件包含的数据的数量，生成一个饼图，并将饼图的每个区域的颜色，填充上对应的数据中包含的 RGB 颜色。用 html 格式网页文件为载体，将生成的饼图显示出来，并生成.svg 格式矢量图。

```
1 d3.csv("./romance.csv") callback > [0] path
2 d3.csv = function(path, callback){
3   var width = 1000;
4   var height = 1000;
5   var radius = Math.min(width, height) / 2;
6   var donutWidth = 300; // NEW
7
8   var lastR = 0;
9   var lastG = 0;
10  var lastB = 0;
11
12  var svg = d3.select("#graph")
13    .append("svg")
14    .attr("width", width)
15    .attr("height", height)
16    .append("g")
17    .attr("transform", "translate(" + (width / 2) +
18      ", " + (height / 2) + ")");
19
20  var arc = d3.arc()
21    .innerRadius(radius - donutWidth) // NEW
22    .outerRadius(radius);
23
24  var pie = d3.pie()
25    .value(function(d) { return 1/3441; })
26    .sort(null);
27
28  var path = svg.selectAll("path")
29    .data(pie(data))
30    .enter()
31    .append("path")
32    .attr("d", arc)
33    .attr("stroke-width", "0")
34    .attr("stroke", "none")
35    .attr("fill", function(d){
36      if(Math.abs(d.data.R - lastR) > 20 || Math.abs(d.data.G - lastG) > 20 || Math.abs(d.data.B - lastB) > 20){
37        lastR = d.data.R;
38        lastG = d.data.G;
39        lastB = d.data.B;
40      }
41      return "rgb(" + lastR + "," + lastG + "," + lastB + ")";
42    });
43
44  return "rgb(" + lastR + "," + lastG + "," + lastB + ")";
45
46  }
```

(JavaScript 文件)