



**HACETTEPE UNIVERSITY**  
**ELECTRICAL AND ELECTRONICS ENGINEERING**  
**ELE492 – SPECIAL TOPICS II**  
**HOMEWORK 1 REPORT**  
**2021-2022 SPRING**

**EGEMEN CAN AYDUĞAN - 21728036**

I have not received or given any aid in this homework. All the work presented below is my own work.

Egemen Can AYDUĞAN

### QUESTION 1

```
1 print("Output:\nFahrenheit(F) Centigrade(C)")
2 for Fahrenheit in [-40, 0, 32, 68, 98.6, 212]:
3     Celcius = (Fahrenheit-32)*5/9
4     print(" ", Fahrenheit, " ", "{0:.1f}".format(Celcius))
5 print("All Done!")
```

I used a for loop. I am giving the values sequentially inside the loop of my variable, which I define as Fahrenheit. I'm returning the loop as much as the value the variable will take. Every time I cycle I convert Fahrenheit to Celsius and print them. With "{0:.1f}".format, I set the number of digits after the point.

```
"C:\Users\ecana\Desktop\Dersler\IV. Sınıf\Image Proccesing\Denemeler\Ders\Scripts\python.exe"
Output:
Fahrenheit(F) Centigrade(C)
-40          -40.0
0            -17.8
32           0.0
68           20.0
98.6         37.0
212          100.0
All Done!

Process finished with exit code 0
```

*Table 1.1. - Result of the First Question*

## QUESTION 2

```
1 def square(x):
2     return x*x
3
4 def fahr_to_cent(fahr):
5     return ((fahr-32)/9.0)*5
6
7 def cent_to_fahr(cent):
8     result = (cent/5.0)*9+32
9     return result
10
11 def abs(x):
12     if x<0:
13         return (-x)
14     else:
15         return (x)
16
17 def print_hello():
18     print("Hello, world")
19
20 def print_fahr_to_cent(fahr):
21     result = fahr_to_cent(fahr)
22     print(result)
23
24 x = 42
25 print(x)
26 result = square(3)+square(4)
27 boiling = fahr_to_cent(212)
28 cold = cent_to_fahr(-40)
29 print(result)
30 print(abs(-22))
31 print(print_fahr_to_cent(32))
```

We can return a number from the function or print the result inside the function with "print".  
We can call a function inside a function.

In the last line of the code, the function prints the answer inside the function, but because the variable is not returned, print(print\_fahr\_to\_cent(32)) writes "None".

```
"C:\Users\ecana\Desktop\Dersler\IV. Sinif\Image Proccesing\Denemeler\Ders\Scripts\python.exe"
42
25
22
0.0
None

Process finished with exit code 0
```

*Table 2.1. - Result of the Second Question*

### QUESTION 3

```
import cv2
import scipy
import openpyxl
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from skimage import io,color,data

# Displaying Image
Image = io.imread("Egemen.jpg")
io.imshow(Image)
plt.show()
# Getting Image Resolution
print("Image Resolution: ",Image.shape)

# Getting Pixel Values
DataFrame = pd.DataFrame(Image.flatten())
FilePath ='pixel_values1.xlsx'
DataFrame.to_excel(FilePath,index=False)

# Converting Color Space
Image_HSV = color.rgb2hsv(Image)
Image_RGB = color.hsv2rgb(Image_HSV)
plt.figure(0)
io.imshow(Image_HSV)
plt.figure(1)
io.imshow(Image_RGB)
plt.show()
```

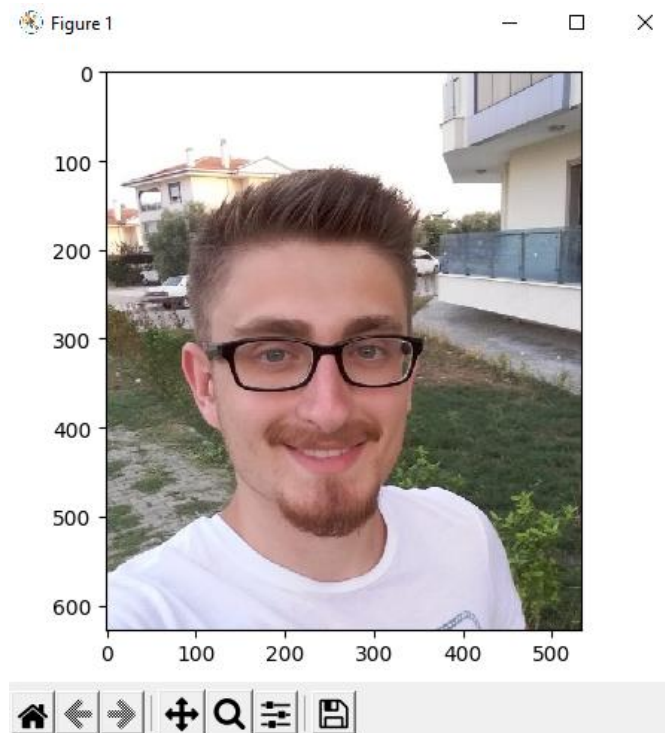


Table 3.1. - Displaying Image

```
"C:\Users\ecana\Desktop\Dersler\IV. Sınıf\Image Proccesing\Denemeler\Ders\Scripts\python.exe"
Image Resolution: (627, 533, 3)
```

*Table 3.2. - Getting Image Resolution*

We have an image with a resolution of  $627 \times 533$ , and it has three channels (because it is in the RGB color format).

1	0
2	255
3	255
4	255
5	255
6	255
7	255
8	255
9	255

•  
•  
•

1002564	61
1002565	50
1002566	56
1002567	59
1002568	48
1002569	48
1002570	51
1002571	40
1002572	40
1002573	43
1002574	32

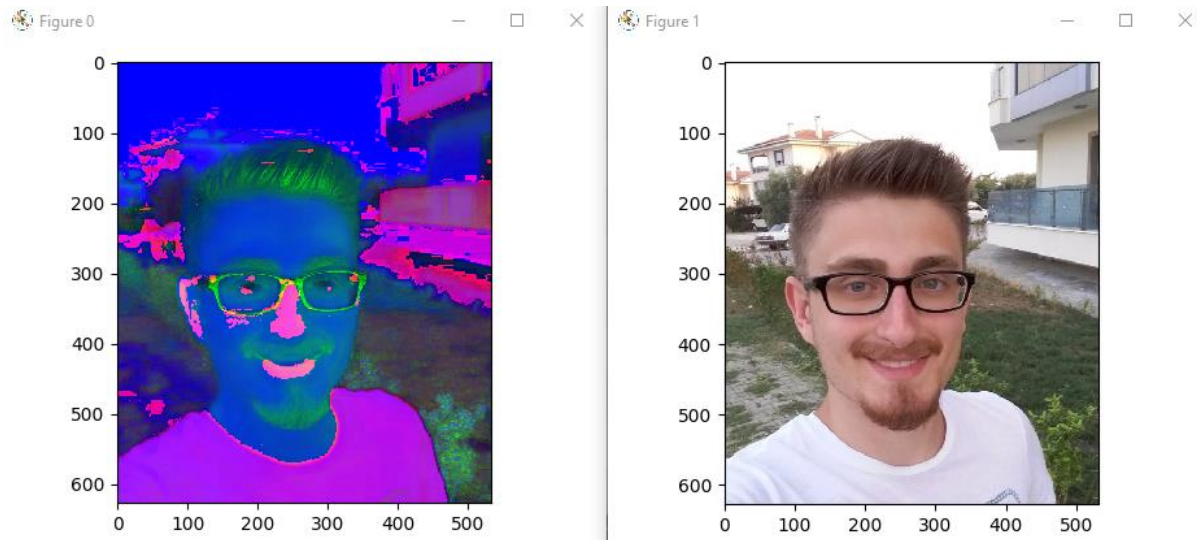
*Table 3.3. - Getting Pixel Values*

We use one row to store all the pixel values. When you look at the code that follows, you can see we are importing another module named pandas. Pandas is used to read, write, and process various file formats. The DataFrame function converts a one-dimensional array into an Excel-like format, with rows and columns.

- $(627 \times 533 \times 3)\text{Row} = (1002574 - 1)\text{Row}$

RGB Color Code (RGB = Red-Green-Blue)

- $[255, 255, 255]$  = White
- $[255, 0, 0]$  = Red
- $[0, 255, 0]$  = Green
- $[0, 0, 255]$  = Blue
- $[0, 0, 0]$  = Black



*Table 3.4. - Converting Color Space*

The final part is converting color space. Firstly, we changed the original form to HSV with `'color.rgb2hsv'`. Then, we recreated the original image with `'color.hsv2rgb'`.