

Report On
Joy Of Programming Using Python

Submitted for Summer Internship Program

By :-

DHAVAL JAIN (CS, 1900270120022)

CHAMAN KUMAR (CS, 1900270120015)

Under the Guidance of

Dr. Pratima Singh (Professor)

Mr. Binayak Parashar (Assistant Professor)



AJAY KUMAR GARG ENGINEERING COLLEGE, GHAZIABAD

YEAR 2020-21

INDEX

1.	ABSTRACT
2.	INPUT
3.	OUTPUT
4.	EXAMPLE
5.	CODE
6.	TEST CASES
7.	ASSIGNMENT

PROJECT - 13 (IMAGE PROCESSING)

ABSTRACT :-

Image processing is a method to perform some operations on an image, in order to get an enhanced image or to extract some useful information from it. It is a type of signal processing in which input is an image and output may be image or characteristics/features associated with that image.

Python provide lots of libraries for image processing including –

OpenCV -

Image processing library mainly focused on real time computer vision with applications in wide-range of areas like 2D & 3D features toolkits, facial & gesture recognitions, human computer, mobile robotics, object identification etc.

Numpy and Scripy libraries -

For image manipulations and processing.

Sckikit -

Provides lots of algorithms for image processing.

Python Image Library -

To perform basic operations on image like create thumbnails, resize, rotation, format conversion etc.

INPUT :-

In input the user has to choose one amongst the several choices given .

OUTPUT :-

In output the user will get the processed image according to the choice opted by the user.

EXAMPLE :-

INPUT :-

```
WELCOME TO IMAGEEDITS

CHOICES :-

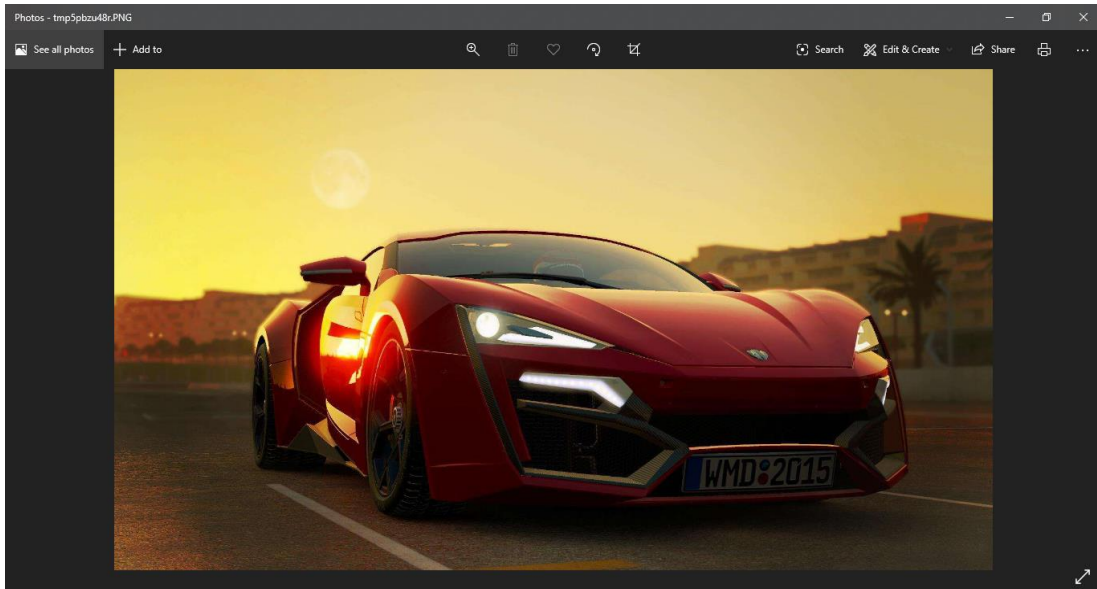
1 :- Image Rotation & Grayscale
2 :- Image Cropping & Blurring Effect
3 :- Image Flipping & Merging The RGB Bands
4 :- Merging Images
5 :- Image Resizing
6 :- Watermark On Image
7 :- Image Formation

Enter Your Choice :- 1

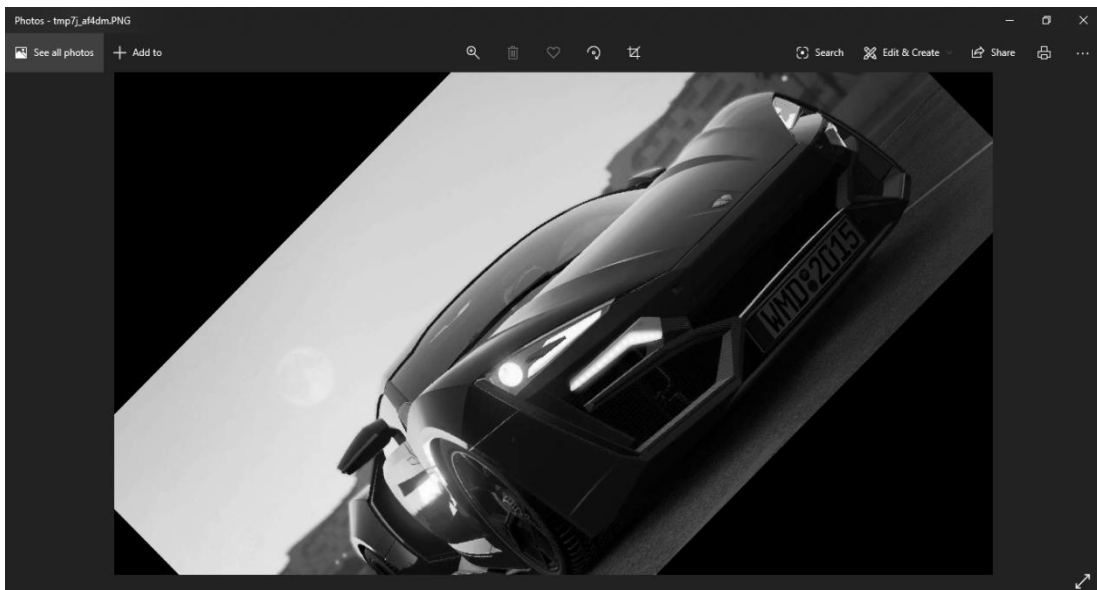
Your Choice Is :- 1
```

OUTPUT :-

ORIGINAL IMAGE



PROCESSED IMAGE



CODE :-

```
Project.py x
1
2 #JOY OF PROGRAMMING USING PYTHON
3 #PROJECT ON IMAGE PROCESSING
4 print("\n\t\tWELCOME TO IMAGEEDITS\n")
5 print("CHOICES :- \n")
6 print("\t 1 :- Image Rotation & Grayscale")
7 print("\t 2 :- Image Cropping & Blurring Effect")
8 print("\t 3 :- Image Flipping & Merging The RGB Bands")
9 print("\t 4 :- Merging Images")
10 print("\t 5 :- Image Resizing ")
11 print("\t 6 :- Watermark On Image")
12 print("\t 7 :- Image Formation")
13 for n in range(1,8):
14     n=int(input("Enter Your Choice :- "))
15     print("\nYour Choice Is :-",n)
16
17
18     if n==1:
19
20         #Import required image library
21         from PIL import Image
22
23         #Open image
24         im = Image.open("C:/Dhaval/Photos/Lykan.jpg")
25
26         #Display original image
27         im.show()
28         im = Image.open("C:/Dhaval/Photos/Lykan.jpg").convert('L')
29
30         #Image rotation & in grayscale
31         im.rotate(45).show()
32
33         #Save the image
34         im.save('C:/Dhaval/Photos/rotation&grayscale.jpg')
35
```

```
36
37 ▼ elif n==2:
38
39     #Import required image library
40     from PIL import Image, ImageFilter
41
42     #Open image
43     OriImage = Image.open('C:/Dhaval/Photos/lykan.jpg')
44
45     #Display original image
46     OriImage.show()
47
48     #Applying boxblur filter
49     boxImage = OriImage.filter(ImageFilter.BoxBlur(20))
50
51     #Save boxblur image
52     boxImage.save('C:/Dhaval/Photos/boxblur.jpg')
53
54     #left, upper, right, lower
55     #Crop
56     cropped = boxImage.crop((700,20,1500,900))
57
58     #Display the cropped & blurred image
59     cropped.show()
60
61     #Save the cropped image
62     cropped.save('C:/Dhaval/Photos/cropped.jpg')
63
64
65 ▼ elif n==3:
66
67     #Using the merge() function, we can merge the RGB bands of an image
68     #Import required image library
69     from PIL import Image
70
71     #Open image
72     image = Image.open("C:/Dhaval/Photos/lykan.jpg")
```

```

73         r, g, b = image.split()
74
75         #Display original image
76         image.show()
77         image = Image.merge("RGB", (b, g, r))
78
79         #Do a flip of left and right
80         hori_flippedImage = image.transpose(Image.FLIP_LEFT_RIGHT)
81
82         #Display the horizontal flipped image
83         hori_flippedImage.show()
84
85         #Save the image
86         hori_flippedImage.save('C:/Dhaval/Photos/merge&flip.jpg')
87
88
89     elif n==4:
90
91         #Import required image library
92         from PIL import Image
93
94         #Open the two images
95         image1 = Image.open('C:/Dhaval/Photos/lykan.jpg')
96         image1.show()
97         image2 = Image.open('C:/Dhaval/Photos/hypersport.jpg')
98         image2.show()
99
100        #resize, first image
101        image1 = image1.resize((426, 240))
102
103        #resize, second image
104        image2 = image2.resize((426, 240))
105
106        image1_size = image1.size
107        image2_size = image2.size
108        new_image = Image.new('RGB', (2*image1_size[0], image1_size[1]), (250,250,250))
109        new_image.paste(image1,(0,0))

```



```
110     new_image.paste(image2,(image1_size[0],0))
111
112     #Save the image
113     new_image.save("C:/Dhaval/Photos/merged_image.jpg","JPEG")
114
115     #Display merged images
116     new_image.show()
117
118
119     elif n==5:
120
121         #Import required image library
122         from PIL import Image
123
124         #Open image
125         im = Image.open("C:/Dhaval/Photos/Lykan.jpg")
126
127         #Display original image
128         im.show()
129
130         #Make the new image half the width and half the height of the original image
131         resized_im = im.resize((round(im.size[0]*0.09), round(im.size[1]*0.09)))
132
133         #Display the resized image
134         resized_im.show()
135
136         #Save the resized image
137         resized_im.save('C:/Dhaval/Photos/resized.jpg')
138
139
140     elif n==6:
141
142
143         #Import required image library
144         from PIL import Image, ImageDraw, ImageFont
145
146         #Open image
```

```

147         im = Image.open('C:/Dhaval/Photos/lykan.jpg')
148         width, height = im.size
149
150         draw = ImageDraw.Draw(im)
151         text = "Lykan Hypersport"
152
153         font = ImageFont.truetype('arial.ttf', 250)
154         textwidth, textheight = draw.textsize(text, font)
155
156         #calculate the x,y coordinates of the text
157         margin = 10
158         x = width - textwidth - margin
159         y = height - textheight - margin
160
161         #draw watermark on the image
162         draw.text((x, y), text, font=font)
163
164         #Display image
165         im.show()
166
167         #Save watermarked image
168         im.save('C:/Dhaval/Photos/watermark.jpg')
169
170
171     elif n==7:
172         #Import required image library
173         from PIL import Image
174         import numpy as np
175
176         arr = np.zeros([150,300], dtype=np.uint8)
177
178         #Set grey value to black or white depending on x position
179         for x in range(300):
180             for y in range(150):
181                 if (x % 16) // 8 == (y % 16) // 8:
182                     arr[y, x] = 0
183                 else:

```

```
184         arr[y, x] = 255
185         img = Image.fromarray(arr)
186
187         #Display image
188         img.show()
189
190         #Save the image
191         img.save('C:/Dhaval/Photos/greyscale.jpg')
192
193         continue
194
195
196     else:
197
198         print("\nThe entered choice is NOT VALID! Please Enter it again")
199
200     #***** END OF THE CODE *****
201
```

TEST CASE 1 :-

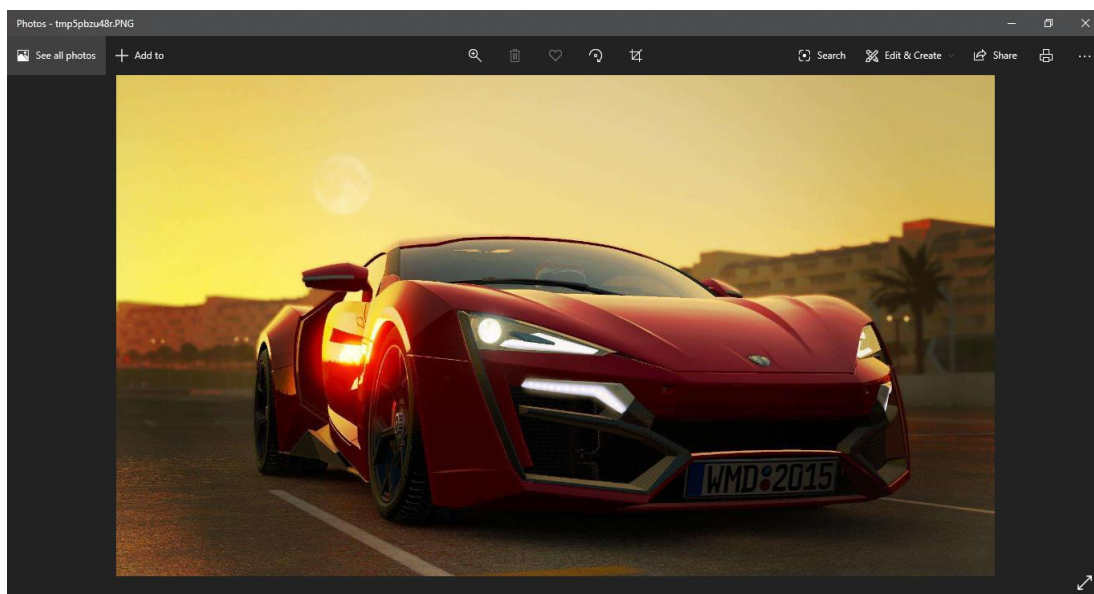
```
WELCOME TO IMAGEEDITS

CHOICES :-

1 :- Image Rotation & Grayscale
2 :- Image Cropping & Blurring Effect
3 :- Image Flipping & Merging The RGB Bands
4 :- Merging Images
5 :- Image Resizing
6 :- Watermark On Image
7 :- Image Formation

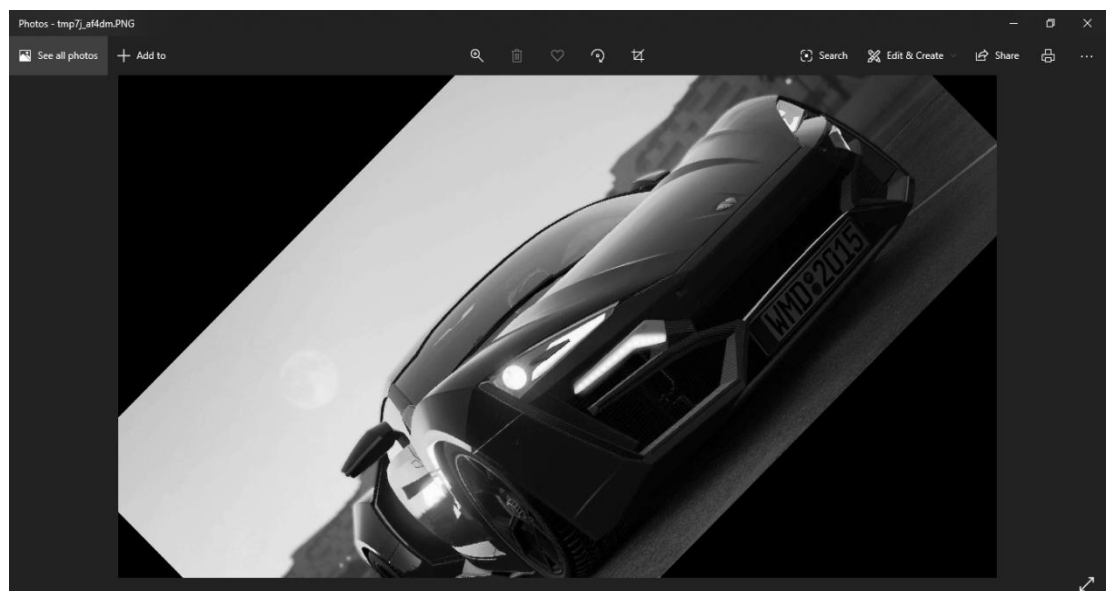
Enter Your Choice :- 1

Your Choice Is :- 1
```



ORIGINAL
IMAGE

OUTPUT
IMAGE



TEST CASE 2 :-

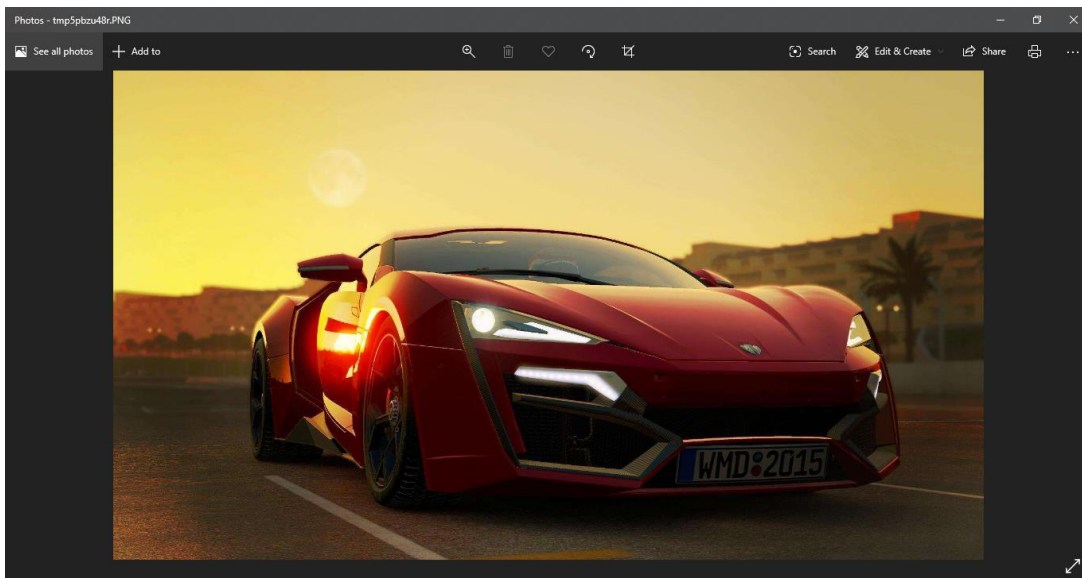
```
WELCOME TO IMAGEEDITS

CHOICES :-

1 :- Image Rotation & Grayscale
2 :- Image Cropping & Blurring Effect
3 :- Image Flipping & Merging The RGB Bands
4 :- Merging Images
5 :- Image Resizing
6 :- Watermark On Image
7 :- Image Formation

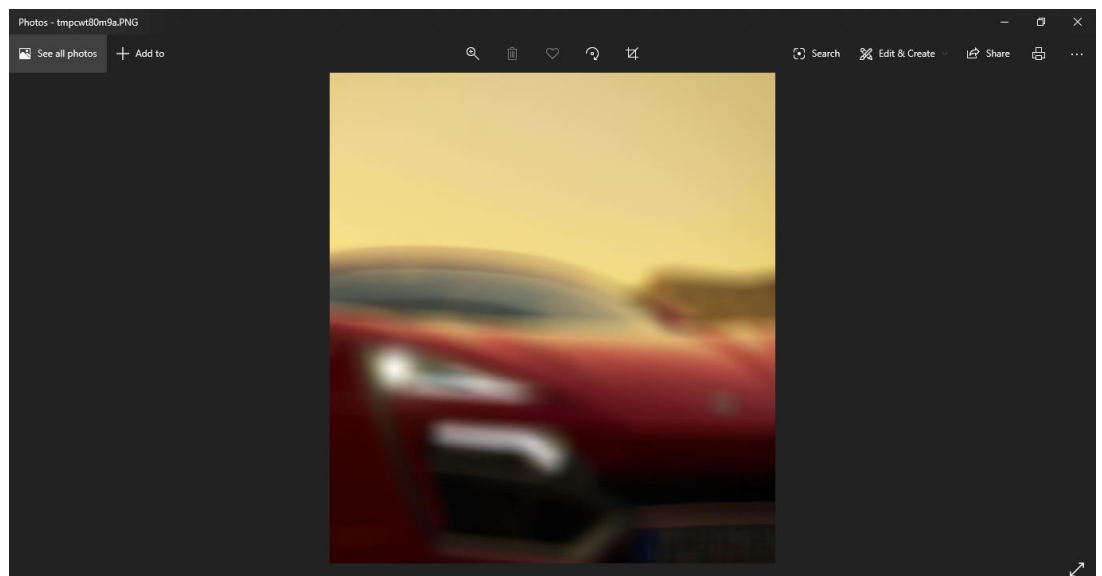
Enter Your Choice :- 1
Your Choice Is :- 1

Enter Your Choice :- 2
Your Choice Is :- 2
```



ORIGINAL
IMAGE

OUTPUT
IMAGE



TEST CASE 3 :-

```
WELCOME TO IMAGEEDITS

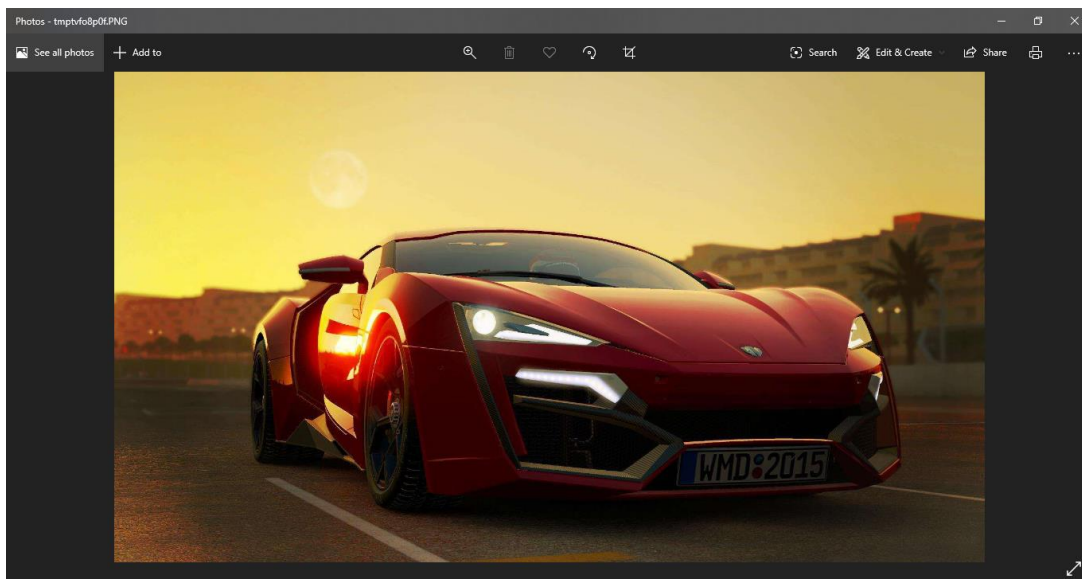
CHOICES :-

1 :- Image Rotation & Grayscale
2 :- Image Cropping & Blurring Effect
3 :- Image Flipping & Merging The RGB Bands
4 :- Merging Images
5 :- Image Resizing
6 :- Watermark On Image
7 :- Image Formation

Enter Your Choice :- 1
Your Choice Is :- 1

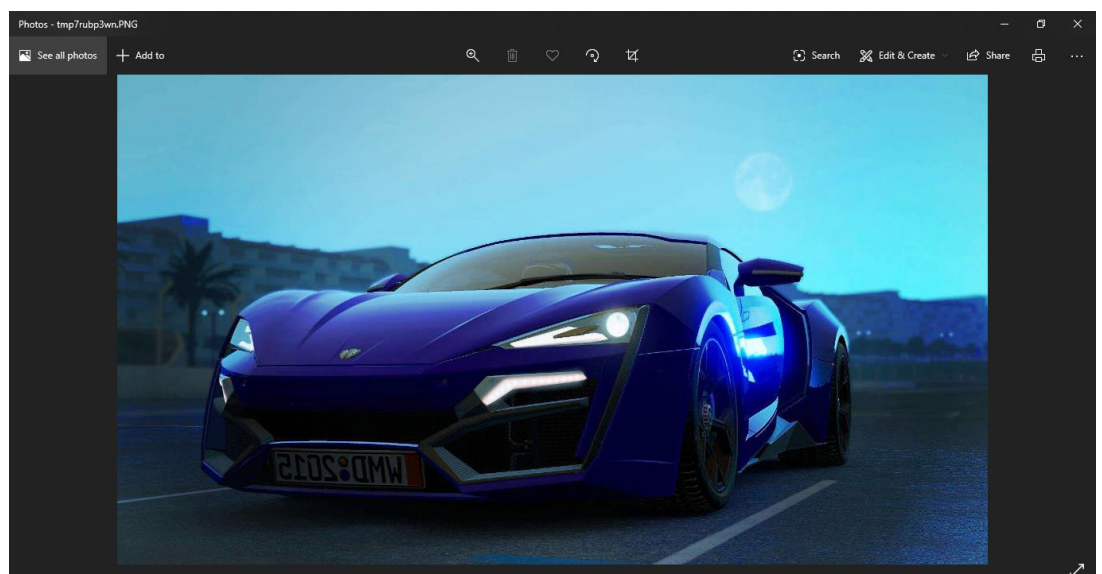
Enter Your Choice :- 2
Your Choice Is :- 2

Enter Your Choice :- 3
Your Choice Is :- 3
```



ORIGINAL
IMAGE

OUTPUT
IMAGE



TEST CASE 4 :-

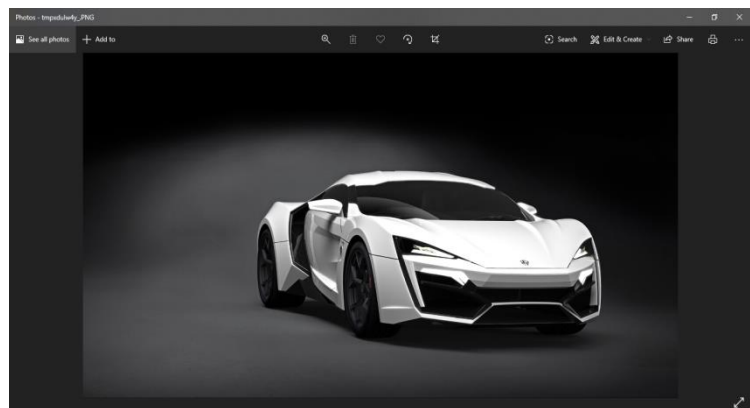
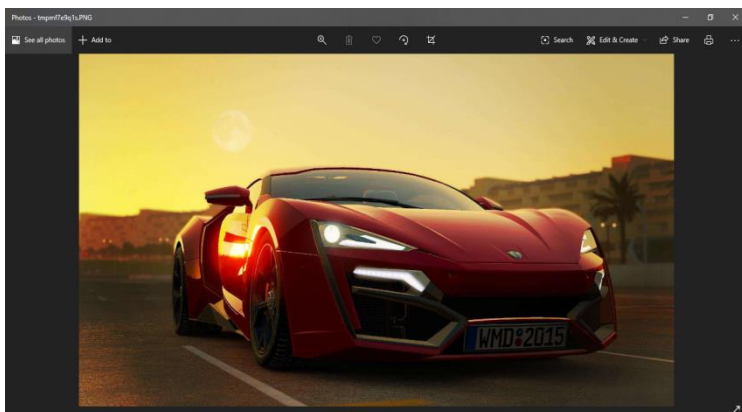
```
WELCOME TO IMAGEEDITS

CHOICES :-

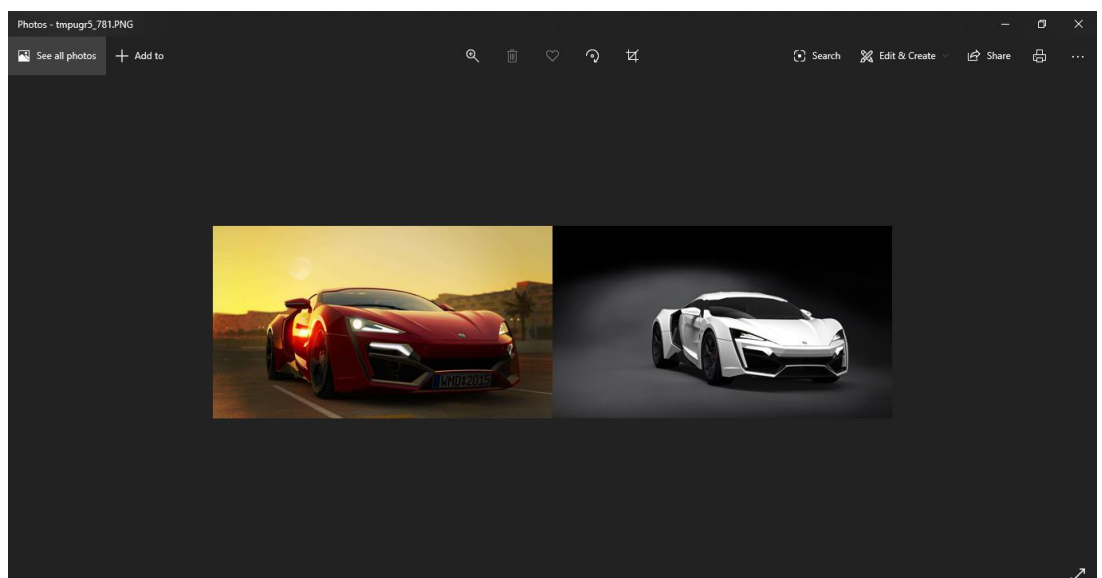
1 :- Image Rotation & Grayscale
2 :- Image Cropping & Blurring Effect
3 :- Image Flipping & Merging The RGB Bands
4 :- Merging Images
5 :- Image Resizing
6 :- Watermark On Image
7 :- Image Formation

Enter Your Choice :- 1
Your Choice Is :- 1
Enter Your Choice :- 2
Your Choice Is :- 2
Enter Your Choice :- 3
Your Choice Is :- 3
Enter Your Choice :- 4
Your Choice Is :- 4
```

ORIGINAL IMAGES



OUTPUT IMAGE



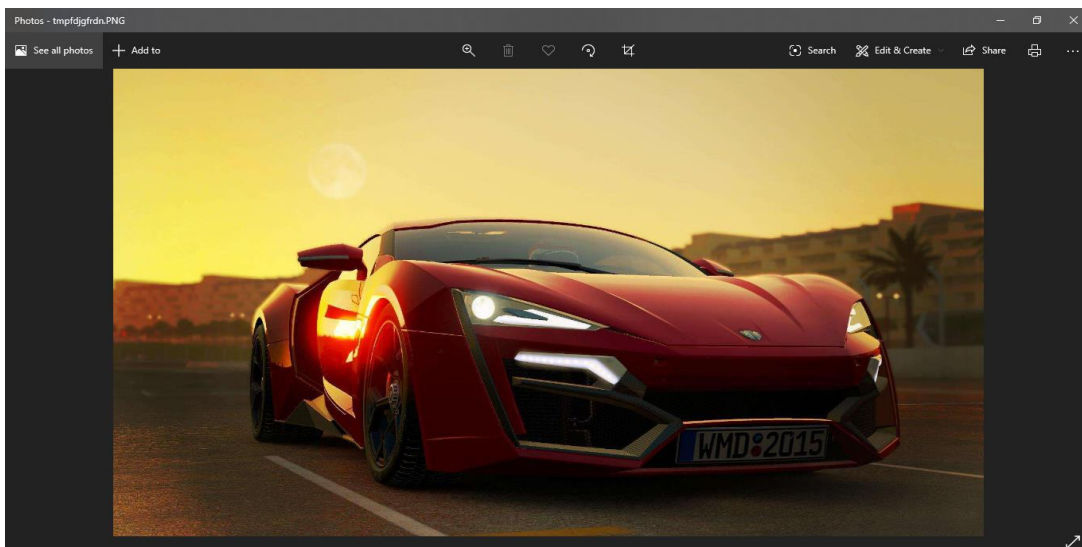
TEST CASE 5 :-

```
WELCOME TO IMAGEEDITS

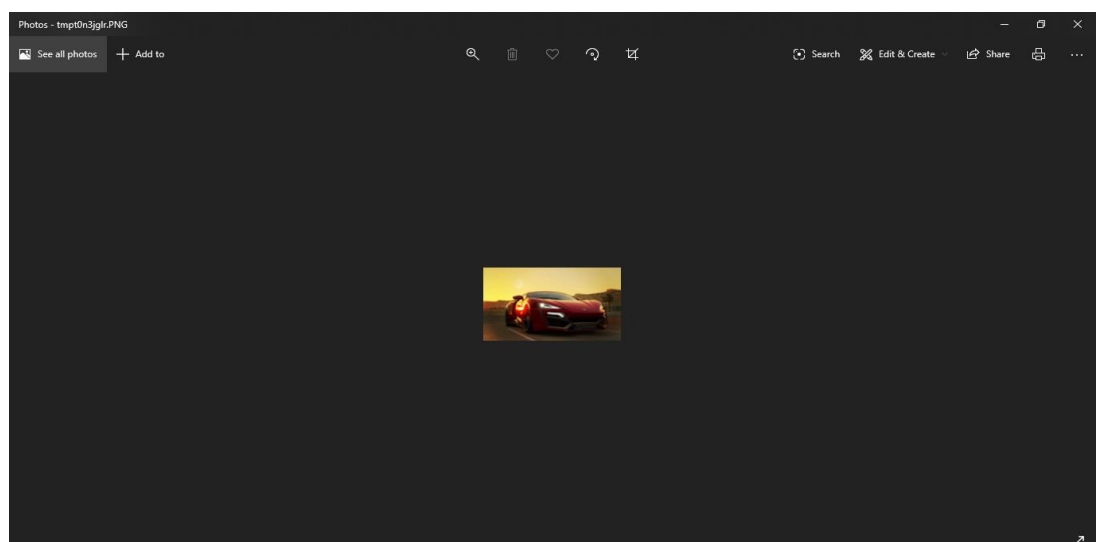
CHOICES :-

1 :- Image Rotation & Grayscale
2 :- Image Cropping & Blurring Effect
3 :- Image Flipping & Merging The RGB Bands
4 :- Merging Images
5 :- Image Resizing
6 :- Watermark On Image
7 :- Image Formation

Enter Your Choice :- 1
Your Choice Is :- 1
Enter Your Choice :- 2
Your Choice Is :- 2
Enter Your Choice :- 3
Your Choice Is :- 3
Enter Your Choice :- 4
Your Choice Is :- 4
Enter Your Choice :- 5
Your Choice Is :- 5
```



ORIGINAL
IMAGE



OUTPUT
IMAGE

TEST CASE 6 :-

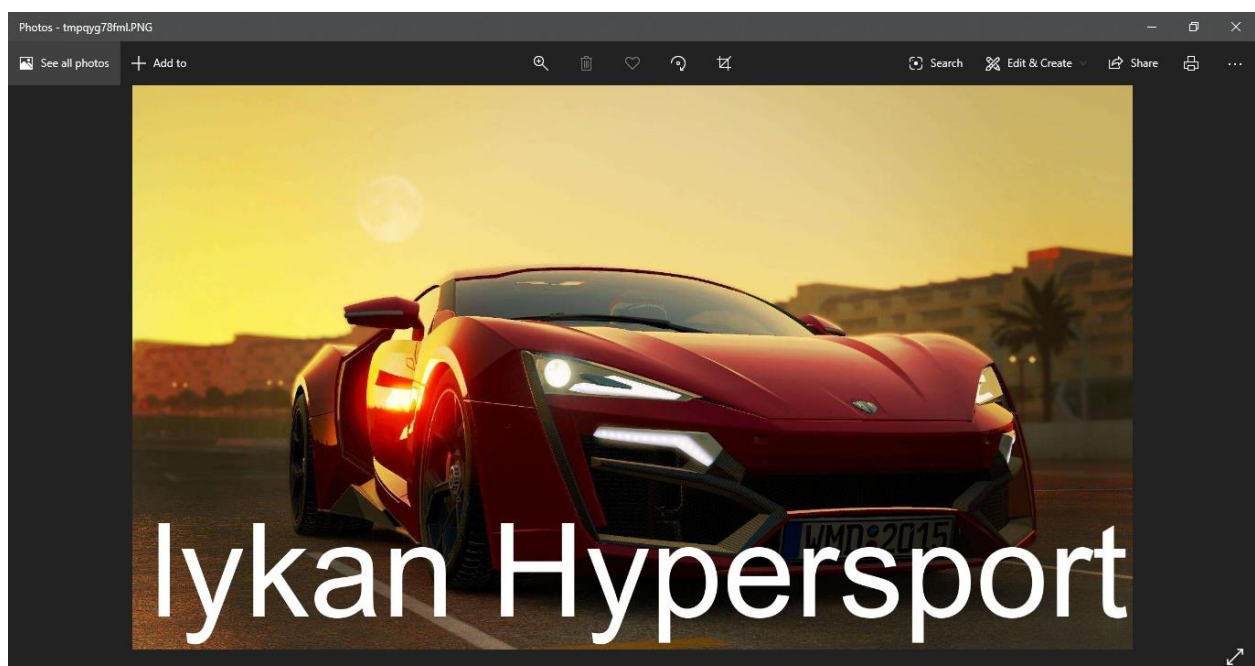
```
WELCOME TO IMAGEEDITS

CHOICES :-

1 :- Image Rotation & Grayscale
2 :- Image Cropping & Blurring Effect
3 :- Image Flipping & Merging The RGB Bands
4 :- Merging Images
5 :- Image Resizing
6 :- Watermark On Image
7 :- Image Formation

Enter Your Choice :- 1
Your Choice Is :- 1
Enter Your Choice :- 2
Your Choice Is :- 2
Enter Your Choice :- 3
Your Choice Is :- 3
Enter Your Choice :- 4
Your Choice Is :- 4
Enter Your Choice :- 5
Your Choice Is :- 5
Enter Your Choice :- 6
Your Choice Is :- 6
```

OUTPUT IMAGE



TEST CASE 7 :-

```
WELCOME TO IMAGEEDITS

CHOICES :-

    1 :- Image Rotation & Grayscale
    2 :- Image Cropping & Blurring Effect
    3 :- Image Flipping & Merging The RGB Bands
    4 :- Merging Images
    5 :- Image Resizing
    6 :- Watermark On Image
    7 :- Image Formation

Enter Your Choice :- 1
Your Choice Is :- 1

Enter Your Choice :- 2
Your Choice Is :- 2

Enter Your Choice :- 3
Your Choice Is :- 3

Enter Your Choice :- 4
Your Choice Is :- 4

Enter Your Choice :- 5
Your Choice Is :- 5

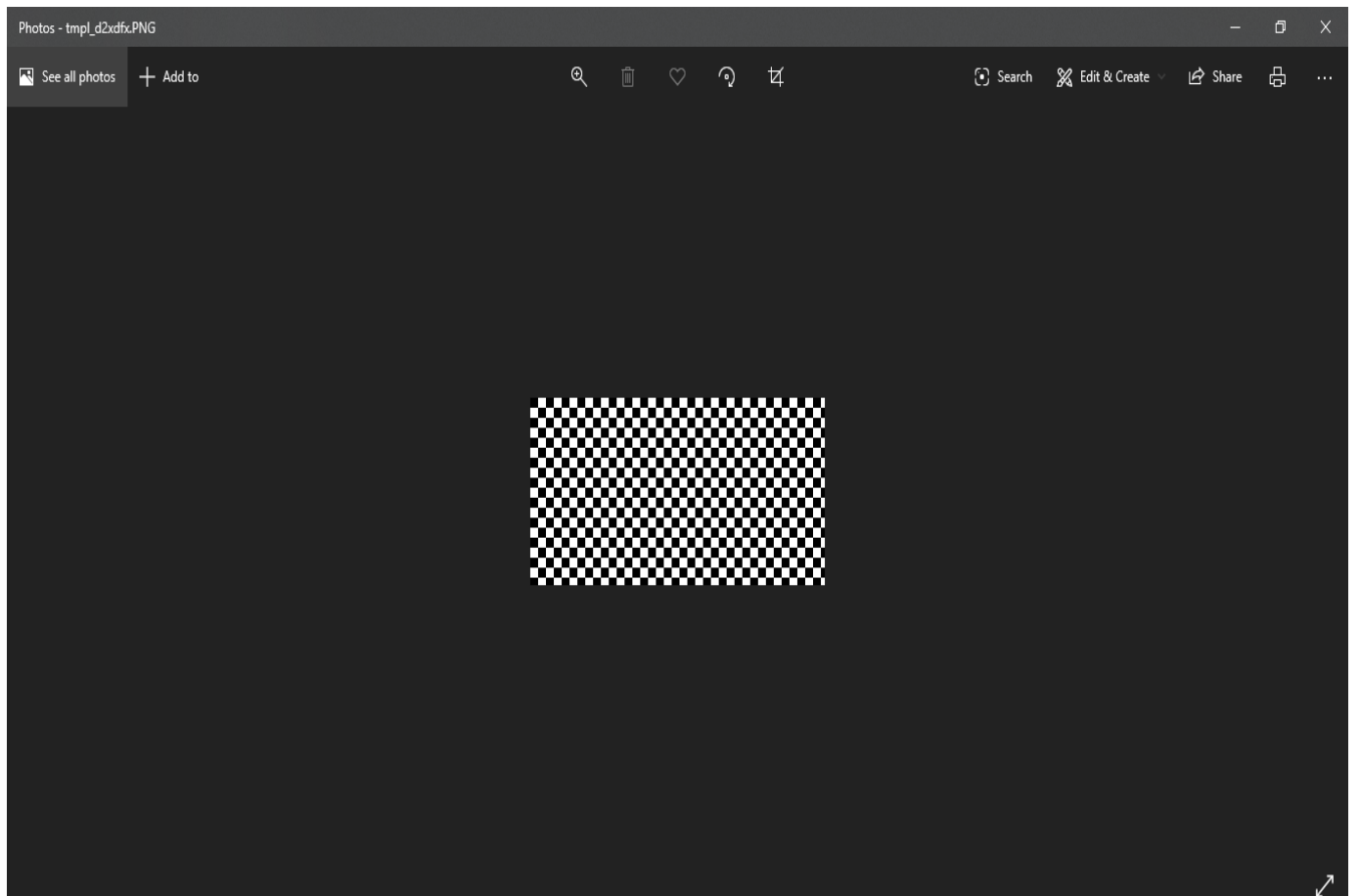
Enter Your Choice :- 6
Your Choice Is :- 6

Enter Your Choice :- 12
Your Choice Is :- 12

The entered choice is NOT VALID! Please Enter it again

Enter Your Choice :- 7
Your Choice Is :- 7
```

OUTPUT IMAGE



ASSIGNMENT :-

Q.) WAP in python to replace every element with the greatest element on right side.

CODE :-

```
C:\Users\Dhaval\spyder-py3\Greatest Integer.py
Greatest Integer.py* x

1
2
3 ▼ def replaceGreatestInteger(List):
4 ▼     for i in range(size-1):
5         List[i] = max(List[i+1:])
6     return List
7 size = int(input("Enter the size of the list: "))
8 List = []
9 ▼ for i in range(size):
10     List.append(int(input("Enter the element number " + str(i+1) + " in the List: ")))
11 print("\nThe list after replacing every element with greatest element on right side is: ", replaceGreatestInteger(List))
12
```

OUTPUT :-

```
Enter the size of the list: 7
Enter the element number 1 in the List: 6
Enter the element number 2 in the List: 5
Enter the element number 3 in the List: 4
Enter the element number 4 in the List: 4
Enter the element number 5 in the List: 3
Enter the element number 6 in the List: 2
Enter the element number 7 in the List: 1
The list after replacing every element with greatest element on right side is: [5, 4, 4, 3, 2, 1, 1]
```

Q.) WAP in python to print Fibonacci series upto N terms.

CODE :-

```
C:\Users\Dhaval\.spyder-py3\Fibonacci Series.py
Fibonacci Series.py x
1
2
3     N = int(input("\nEnter the value of N : "))
4     a = 0
5     b = 1
6     print("\nThe Fibonacci series upto", N,"is :")
7     for i in range(N):
8         print(a, end = " ")
9         c = a + b
10        a = b
11        b = c
12
```

OUTPUT :-

```
Enter the value of N : 9

The Fibonacci series upto 9 is :
0 1 1 2 3 5 8 13 21
```

Q.) WAP in python to print the pattern.

CODE :-

```
C:\Users\Dhaval\spyder-py3\Pattern.py
Pattern.py* x
1
2
3     n = int(input("Enter the Value of N : \n"))
4     for i in range(n):
5         print("* " * (n - i) + "    " * i + " *" * (n - i))
6     for i in range(2,n + 1):
7         print("* " * i + "    " * (n - i) + " *" * i)
8
```

OUTPUT :-

```
Enter the Value of N :
7
* * * * * * * * * * * * * *
* * * * *      * * * * *
* * * *      * * * *
* * * *      * * * *
* * *      * * *
* *      * *
*      *
* *      * *
* * *      * * *
* * * *      * * * *
* * * * *      * * * * *
* * * * * *      * * * * *
* * * * * * * * * * * * *
```