Date 01-07-2021

NumPy and Image Assessment Test

NumPy and Image Assessment

COMPLETE THE TASKS IN BOLD BELOW.

TASK: Import NumPy

In [1]:

```
import numpy as np
```

TASK: Create a 5 by 5 array where every number is a 10

In [2]:

```
arr_1=np.zeros((5,5))
```

In [3]:

```
arr_1=arr_1+10
arr_1
```

Out[3]:

```
array([[10., 10., 10., 10., 10.],
       [10., 10., 10., 10., 10.],
       [10., 10., 10., 10., 10.],
       [10., 10., 10., 10., 10.],
       [10., 10., 10., 10., 10.]])
```

TASK: Run the cell below to create an array of random numbers and see if you can figure out how it works.

In [4]:

```
# This line sets a "seed" so you get the same random numbers we do
np.random.seed(101)
# This line creates an array of random numbers
arr = np.random.randint(low=0,high=100,size=(5,5))
```

TASK: What are the largest and smalled values in this array?

In [5]:

```
arr.max()
```

Out[5]:

```
95
```

In [6]:

```
arr.min()
```

Out[6]:

```
4
```

TASK: Use PIL and matplotlib to read and display the ../DATA/00-puppy.jpg image.

In [7]:

```
from PIL import Image
img=Image.open("00-puppy.jpg")
img
```

Out[7]:

TASK: Convert the image to a NumPy Array

In [8]:

```
arr_2=np.asarray(img)
arr_2
```

Out[8]:

```
array([[[95, 81, 78],
        [97, 83, 80],
        [98, 84, 81],
        ...,
        [25, 27, 22],
        [25, 27, 22],
        [25, 27, 22]],

       [[95, 81, 78],
        [96, 82, 79],
        [96, 82, 79],
        ...,
        [25, 27, 22],
        [25, 27, 22],
        [25, 27, 22]],

       [[95, 81, 78],
        [94, 80, 77],
        [94, 80, 77],
        ...,
        [25, 27, 22],
        [25, 27, 22],
        [25, 27, 22]],

       ...,

       [[19, 29, 20],
        [20, 30, 21],
        [20, 30, 21],
        ...,
        [23, 30, 22],
        [24, 31, 23],
        [24, 31, 23]],

       [[20, 30, 21],
        [20, 30, 21],
        [19, 29, 20],
        ...,
        [23, 30, 22],
        [24, 31, 23],
        [24, 31, 23]],

       [[20, 30, 21],
        [19, 29, 20],
        [19, 29, 20],
        ...,
        [23, 30, 22],
        [24, 31, 23],
        [24, 31, 23]]], dtype=uint8)
```

FINAL TASK: Use slicing to set the RED and GREEN channels of the picture to 0, then use imshow() to show the isolated blue channel

In [9]:

```
arr_blue_only=arr_2.copy()
arr_blue_only[:,:,:2]=0
```

In [10]:

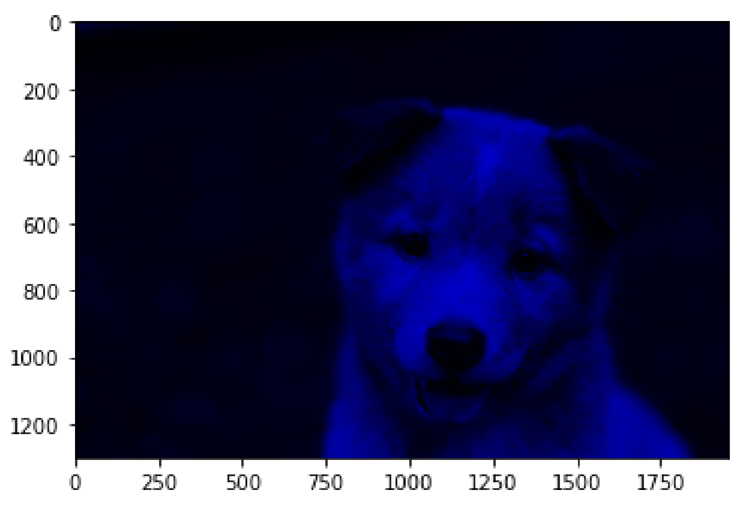
```
import matplotlib.pyplot as plt
```

In [11]:

```
plt.imshow(arr_blue_only)
```

Out[11]:

```
<matplotlib.image.AxesImage at 0x24e10491b50>
```



Great Job!