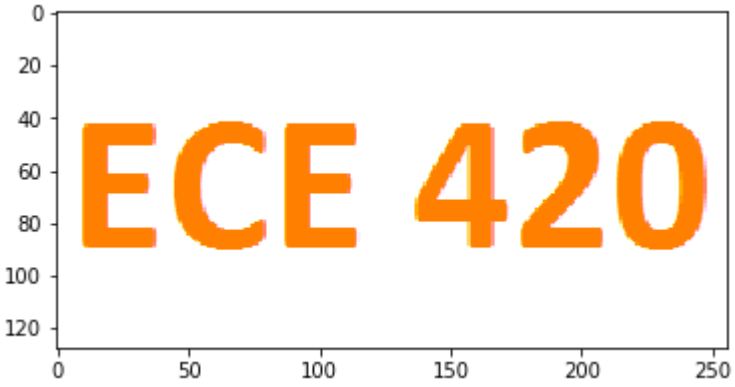
In [14]:
```
import cv2
import matplotlib.pyplot as plt

img = cv2.imread('logo.png')
plt.figure()
# OpenCV image channel is BGR so we flip the channels to RGB
plt.imshow(img[:,:,::-1])
plt.show()
```



NOTE

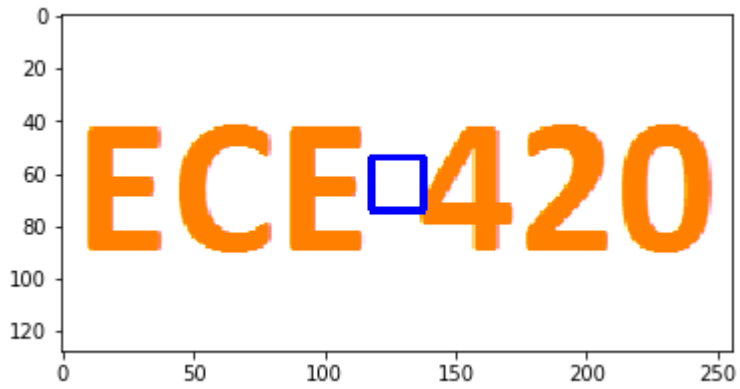
- OpenCV have its own cv2.imshow(), however it is not compatible with Jupyter Notebook, so we still uses the plt module. Note that OpenCV loaded image is still just a Numpy array.

Assignment 1

- Draw a square with size 20 and color blue on to the center of the image. Add the text "Prelab7" with font FONT_HERSHEY_SIMPLEX and color red on to the center top of the image. A sample image is shown below.
- Draw the rectangle and add the text using OpenCV functions, in particular, cv2.rectangle() and cv2.putText().

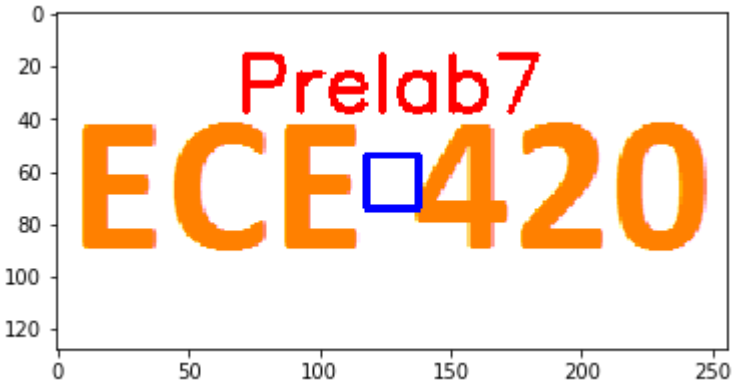
In [30]:
```
import cv2
import numpy as np
import matplotlib.pyplot as plt

img = cv2.imread('logo.png')

rh = int(len(img[:,0,0])/2)
ch = int(len(img[0,:,0])/2)
c = 10
window_name = 'Img'
# Start coordinate, here (5, 5)
# represents the top left corner of rectangle
start_point = (ch+c, rh-c)

# Ending coordinate, here (220, 220)
# represents the bottom right corner of rectangle
end_point = (ch-c, rh+c)
# Line thickness of 2 px
thickness = 2

# Using cv2.rectangle() method
# Draw a rectangle with blue line borders of thickness of 2 px
#img = np.flip(img,axis=2)
image = cv2.rectangle(img, start_point, end_point,(255,0,0), thickness)
# Displaying the image
plt.figure()
plt.imshow(image)
plt.imshow(img[:,:,::-1])
```

Out[30]:
```
<matplotlib.image.AxesImage at 0x4087590688>
```



In [31]:
```
# setup text
font = cv2.FONT_HERSHEY_SIMPLEX
text = "Prelab7"

# get boundary of this text
textsize = cv2.getTextSize(text, font, 1, 2)[0]

# get coords based on boundary
textX = int((img.shape[1] - textsize[0]) / 2)
textY = int((img.shape[0] + textsize[1]) / 4)

# add text centered on image
img2 = cv2.putText(image, text, (textX, textY ), font, 1, (0, 0, 255), 2)
plt.figure()
plt.imshow(img2)
plt.imshow(img[:,:,::-1])
```

Out[31]: <matplotlib.image.AxesImage at 0x40875ac4c8>



In [ ]:

In [ ]: