Session 5 Computer vision - How to draw different geometric shapes using opencv

In [1]:
```
# how to draw a line
import numpy as np
import cv2

img = cv2.imread('2.jpg',1)

cv2.imshow('original', img)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [2]:
```
# how to draw a line
import numpy as np
import cv2

img = cv2.imread('2.jpg',1)
img = cv2.line(img,(0,0), (255,255), (255,0,0), 10)    # line( filename, starting,ending coordinate, color(BGR), thickness)

cv2.imshow('original', img)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [3]:
```
# how to draw a Arrowedline
import numpy as np
import cv2

img = cv2.imread('2.jpg',1)
img = cv2.arrowedLine(img,(0,0), (230,500), (255,100,180), 10)   # line( filename, starting,ending coordinate, color(BGR), thickness)

cv2.imshow('original', img)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [4]:
```
# how to draw a rectange
import numpy as np
import cv2

img = cv2.imread('2.jpg',1)
img = cv2.rectangle(img,(400,0), (510,128), (100,100,180), 10)   # line( filename, starting,ending coordinate, color(BGR), thickness)

cv2.imshow('original', img)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [ ]:
```
# how to draw a filledrectange
import numpy as np
import cv2

img = cv2.imread('2.jpg',1)
img = cv2.rectangle(img,(400,0), (510,128), (100,100,180), -1)   # line( filename, starting,ending coordinate, color(BGR), thickness)

cv2.imshow('original', img)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [5]:
```
# how to draw a filledrectange
import numpy as np
import cv2

img = cv2.imread('2.jpg',1)
img = cv2.circle(img,(255,255), 75, (100,150,180), -1)   # line( filename,center,radius, color(BGR), thickness)

cv2.imshow('original', img)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [ ]:
```
# to put text on a image
import numpy as np
import cv2

img = cv2.imread('2.jpg',1)
font = cv2.FONT_HERSHEY_SIMPLEX
# puttext(filename, start, font,size of the font, color,thickness, linetype)
img = cv2.putText(img, 'OpenCV',(0,500),font,5,(20,20,65),5, cv2.LINE_AA)

cv2.imshow('original', img)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [6]:
```
# CREATING AN IMAGE USING NUMPY ZEROS
import numpy as np
import cv2
img = np.zeros([512,521,3], np.uint8)     # ([height, width, channel], type) it will create a black screen
#img = cv2.imread('2.jpg',1)
font = cv2.FONT_HERSHEY_SIMPLEX
# puttext(filename, start, font,size of the font, color,thickness, linetype)
img = cv2.putText(img, 'OpenCV',(0,400),font,3,(20,20,65),5, cv2.LINE_AA)

cv2.imshow('original', img)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

In [ ]:
```
#Changing frame size
import cv2
vcap = cv2.VideoCapture(0)
print(vcap.get(cv2.CAP_PROP_FRAME_WIDTH))     #3
print(vcap.get(cv2.CAP_PROP_FRAME_HEIGHT))    #4

vcap.set(3, 300)
vcap.set(4, 300)

print(vcap.get(3))
print(vcap.get(4))

print(vcap.isOpened())
while(vcap.isOpened()):
    ret, frame = vcap.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    cv2.imshow('frame', gray)

    if cv2.waitKey(10) & 0Xff ==ord('q'):     # if we increase the waitkey parameter video will get delayed
        break

vcap.release()
cv2.destroyAllWindows()
```

In [7]:
```
#print the width and height value on default camera screen
import cv2
vcap = cv2.VideoCapture(0)
print(vcap.get(cv2.CAP_PROP_FRAME_WIDTH))     #3
print(vcap.get(cv2.CAP_PROP_FRAME_HEIGHT))    #4

vcap.set(3, 1200)
vcap.set(4, 720)

print(vcap.get(3))
print(vcap.get(4))

print(vcap.isOpened())
while(vcap.isOpened()):
    ret, frame = vcap.read()
    if ret ==True:
        font = cv2.FONT_HERSHEY_SIMPLEX
        text = 'Width: '+ str(vcap.get(3))+ ' ' + 'Height: ' + str(vcap.get(4))
        frame = cv2.putText(frame,text, (10,50), font, 1,(0,255,255), 1, cv2.LINE_AA )

    cv2.imshow('frame', frame)

    if cv2.waitKey(10) & 0Xff ==ord('q'):     # if we increase the waitkey parameter video will get delayed
        break

vcap.release()
cv2.destroyAllWindows()
```

```
640.0
480.0
960.0
540.0
True
```

In [ ]:
```
#print the current date and time on default camera screen
import cv2
import datetime
vcap = cv2.VideoCapture(0)
print(vcap.get(cv2.CAP_PROP_FRAME_WIDTH))     #3
print(vcap.get(cv2.CAP_PROP_FRAME_HEIGHT))    #4

#vcap.set(3, 1200)
#vcap.set(4, 720)

#print(vcap.get(3))
#print(vcap.get(4))

print(vcap.isOpened())
while(vcap.isOpened()):
    ret, frame = vcap.read()
    if ret ==True:
        font = cv2.FONT_HERSHEY_SIMPLEX
        #text = 'Width: '+ str(vcap.get(3))+ ' ' + 'Height: ' + str(vcap.get(4))
        datet = str(datetime.datetime.now())
        frame = cv2.putText(frame,datet, (2,50), font, 0.5,(0,255,255), 1, cv2.LINE_AA )

    cv2.imshow('frame', frame)

    if cv2.waitKey(10) & 0Xff ==ord('q'):     # if we increase the waitkey parameter video will get delayed
        break

vcap.release()
cv2.destroyAllWindows()
```

In [8]:
```
#mouse click event
import numpy as np
import cv2

events = [i for i in dir(cv2) if 'EVENT' in i]
print(events)
```

```
['EVENT_FLAG_ALTKEY', 'EVENT_FLAG_CTRLKEY', 'EVENT_FLAG_LBUTTON', 'EVENT_FLAG_MBUTTON', 'EVENT_FLAG_RBUTTON', 'EVENT_FLAG_SHIFTKEY',
'EVENT_LBUTTONDBLCLK', 'EVENT_LBUTTONDOWN', 'EVENT_LBUTTONUP', 'EVENT_MBUTTONDBLCLK', 'EVENT_MBUTTONDOWN', 'EVENT_MBUTTONUP', 'EVENT_M
OUSEHWHEEL', 'EVENT_MOUSEMOVE', 'EVENT_MOUSEWHEEL', 'EVENT_RBUTTONDBLCLK', 'EVENT_RBUTTONDOWN', 'EVENT_RBUTTONUP']
```

In [9]:
```
#mouse click event
import numpy as np
import cv2

def click_event(event,x, y, flag, param):
    if event == cv2.EVENT_LBUTTONDOWN:
        print(x, ', ', y)
        font = cv2.FONT_HERSHEY_SIMPLEX
        strXY = str(x)+ ','+ str(y)
        cv2.putText(img,strXY,(x,y), font,0.5,(255,255,0), 2)
        cv2.imshow('image', img)

img = np.zeros((512,512,3), np.uint8)
cv2.imshow('image', img)

cv2.setMouseCallback('image', click_event)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

```
112 ,  179
```

In [10]:
```
#mouse click event (display BGR Channel) - Assignment
import numpy as np
import cv2

def click_event(event,x, y, flag, param):
    if event == cv2.EVENT_LBUTTONDOWN:
        print(x, ', ', y)
        font = cv2.FONT_HERSHEY_SIMPLEX
        strXY = str(x)+ ','+ str(y)
        cv2.putText(img,strXY,(x,y), font,0.5,(255,255,0), 2)
        cv2.imshow('image', img)

    if event ==cv2.EVENT_RBUTTONDOWN:
        blue = img[y,x,0]
        green = img[y,x,1]
        red = img[y,x,2]
        font = cv2.FONT_HERSHEY_SIMPLEX
        strBGR = str(blue)+ ','+ str(green)+ ','+ str(red)
        cv2.putText(img,strBGR,(x,y), font,0.5,(255,255,255), 2)
        cv2.imshow('image', img)
img = cv2.imread('2.jpg')
#img = np.zeros((512,512,3), np.uint8)
cv2.imshow('image', img)

cv2.setMouseCallback('image', click_event)

cv2.waitKey(0)
cv2.destroyAllWindows()
```

```
380 ,  416
```

In [ ]: