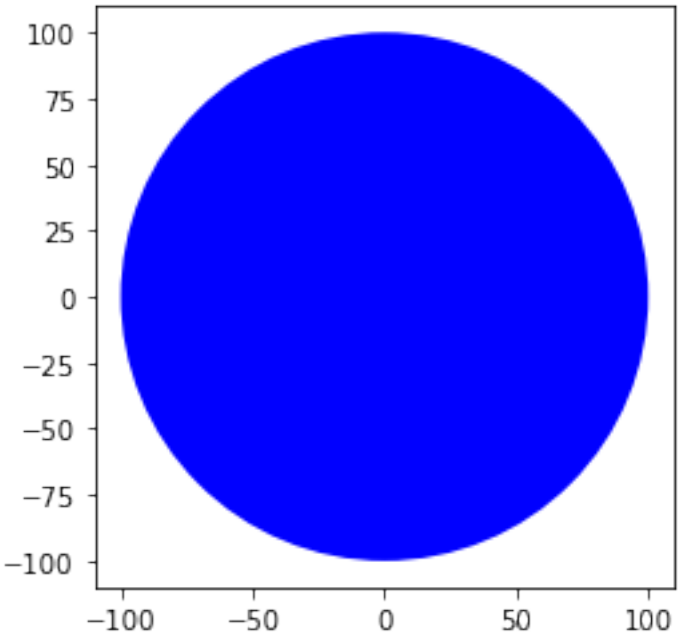
object_class

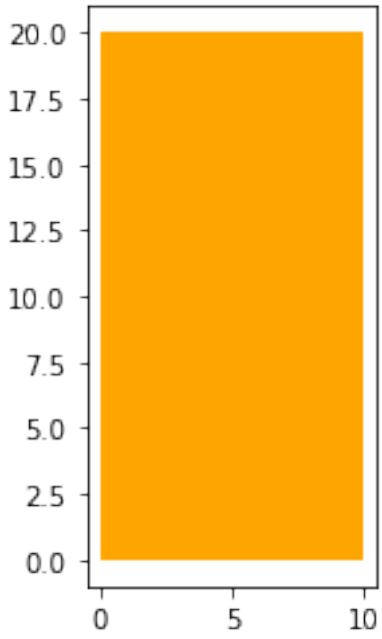
October 17, 2019

[18]:
```
import matplotlib.pyplot as plt
%matplotlib inline

#create a circle
class Circle(object):
    #contructor
    def __init__(self,radius,color):
        self.radius=radius
        self.color=color

    #method
    #def add_radius(self,r):
        #self.radius=self.radius+r
        #return (self.raius)
    #method
    def drawCircle(self):
        plt.gca().add_patch(plt.Circle((0,0),radius=self.radius,fc=self.color))
        plt.axis('scaled')
        plt.show()


#Create an object
BlueCircle=Circle(100,'blue')

#find out the methods
dir(BlueCircle)

#print the object attribute raius
BlueCircle.radius

#print color
BlueCircle.color

#call the method drawCircle
BlueCircle.drawCircle()
```

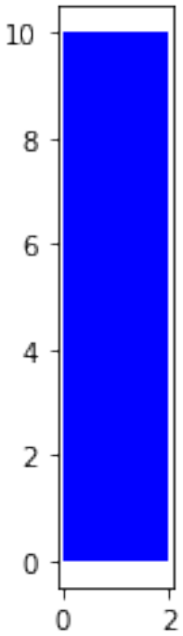


[47]:
```
class Rectangle(object):

    #def __init__(self,width=2,height=3,color='red'):
        #self.height=height
        #self.width=width
        #self.color=color
    def __init__(self, height,width, color):
        self.height = height
        self.width = width
        self.color = color

    def drawRectangle(self):
        plt.gca().add_patch(plt.Rectangle((0,0),self.width,self.height,fc=self.
 ↪color))
        plt.axis('scaled')
        plt.show()

OrangeRect=Rectangle(20,10,'orange')

dir(OrangeRect)

OrangeRect.height

OrangeRect.width

OrangeRect.color
```

```
OrangeRect.drawRectangle()
```



[40]:
```
class Rectangle(object):
    def __init__(self, width=2, height=3, color='r'):
        self.height = height
        self.width = width
        self.color = color
    def drawRectangle(self):
        plt.gca().add_patch(plt.Rectangle((0, 0), self.width, self.
 ↪height,fc=self.color))
        plt.axis('scaled')
        plt.show()

SkinnyBlueRectangle = Rectangle(2, 10, 'blue')

SkinnyBlueRectangle.height

SkinnyBlueRectangle.width

SkinnyBlueRectangle.color

SkinnyBlueRectangle.drawRectangle()
```


10
8
6
4
2
0
0
2

[ ]: