**Using Python from the Shell**

**When you start a program, open Python, go to FILE and New File.**

**The first time you save your program, name it, your name program name and save it to the documents folder.**

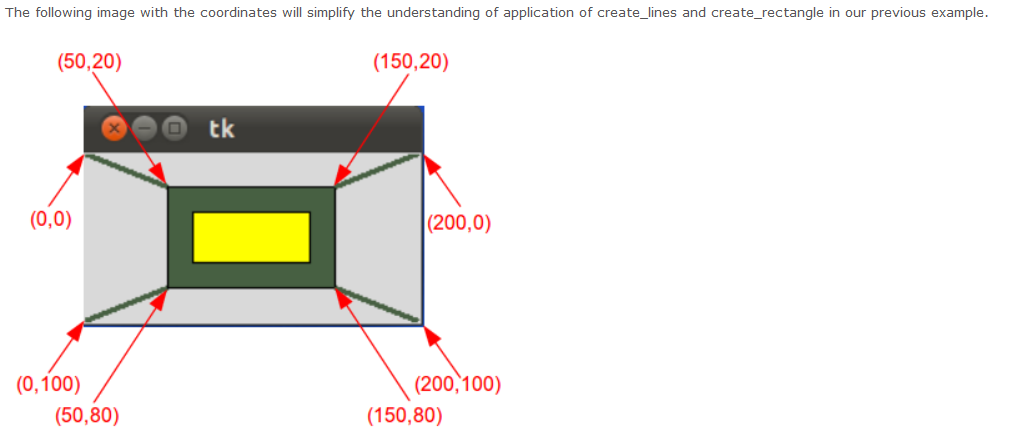
**Using Tkinter for Better Graphics**

What is Tkinter?

Tkinter is a more powerful module that allows the user to create full applications as well as graphic animations.

Tkinter is Python's standard GUI (Graphical User Interface) package.

Tkinter uses widgets to draw shapes and make animations.



**#Programming Note – order of code**

When you use variables and functions in your code:

* #Your Name
* import
* **from tkinter import \*  
  tk=Tk()  
  canvas = Canvas(tk,width=500, height=100)  
  canvas.pack()**
* define variables
* define functions
* write code to call variables and functions in buttons or other widgets

**If you write the code to call the functions, first, the compiler won't see them.**

**All programs will start with:**

from tkinter import \*  
tk=Tk()  
canvas = Canvas(tk,width=500, height=100)  
**canvas.pack()**

**We will be using**[Python for Kids.pdf](https://docs.google.com/viewer?a=v&pid=sites&srcid=ZGVmYXVsdGRvbWFpbnx3c2hzY29tcHV0ZXJzY2llbmNlfGd4Ojc1NjYwZjU2MTFiNDVhMGU) **starting on p. 163.**

**# Button function in Tkinter**

btn =Button(tk, text = "click me", command = hello)

**btn.pack()**

**Shapes:**

canvas.create\_line(0,0,500,500)

canvas.create\_rectangle(10,10,160,60,fill='red')

canvas.create\_oval(240, 240, 320,320,outline="blue", fill="yellow")

canvas.create\_rectangle(180,180,280,210, fill ="green")

canvas.create\_polygon(50,100,150,100,150,200, fill='red')

**Add color to your shapes**

fill = "color"

outline = "color"

**The following code opens a colorDialogBox and allows the user to choose a color.**

from tkinter.colorchooser import \*

def getColor():

color = askcolor()

print(color)

return color[1]

**The following code will show the x,y coordinate when you click on the canvas.**

from tkinter import \*

tk = Tk()

def callback(event):

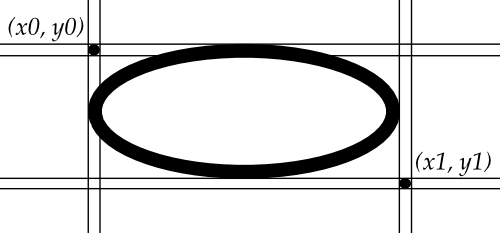
print ("clicked at", event.x, event.y)

canvas = Canvas(tk, width=500,height=500)

canvas.bind("<Button-1>", callback)

canvas.pack()

You can display one or more lines of text on a canvas by creating a text object:

 *canvas.create\_text(400, 450, text='This is a text example.')*

**Canvas oval objects**

Ovals, mathematically, are ellipses, including circles as a special case. The ellipse is fit into a rectangle defined by the coordinates (*x0*, *y0*) of the top left corner and the coordinates (*x1*, *y1*) of a point just *outside of* the bottom right corner.

The oval will coincide with the top and left-hand lines of this box, but will fit just inside the bottom and right-hand sides.

To create an ellipse on a canvas *C*, use:

*id* = *C*.create\_oval(*x0*, *y0*, *x1*, *y1*, *option*, ...)

[**http://infohost.nmt.edu/tcc/help/pubs/tkinter/web/create\_oval.html**](http://infohost.nmt.edu/tcc/help/pubs/tkinter/web/create_oval.html)