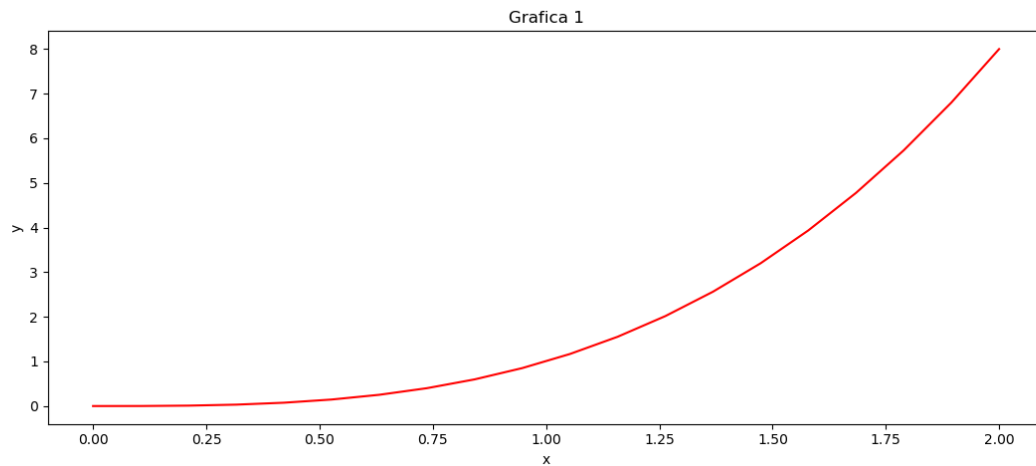

Gráficas

Programa *.py para ejecutar en la primer clase

1- Con el siguiente script implemente la siguiente gráfica



```
from pylab import *
import matplotlib.pyplot as plt
from pylab import *
import math
#-----
def funcion(xi,xf):
    x = linspace(xi,2,xf)
    y = x**3
    return x,y
#grafica 1
#-----
def grafica1(zi,zf):
    x,y=funcion(zi,zf)
    figure()
    plot(x, y, 'r')
    xlabel('x')
    ylabel('y')
    title('Gráfica 1')
    show()
    return

#main
#-----
xi=0
xf=20
grafica1(xi,xf)

#end script
```

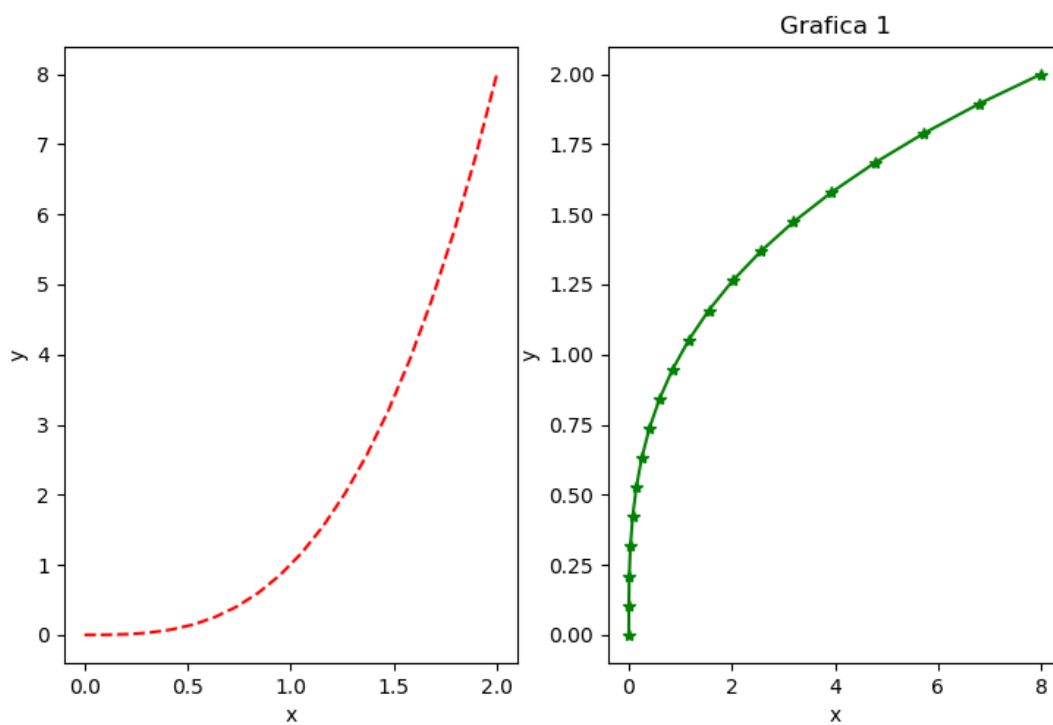
2- Experimente y explique lo que se observa cuando:

- i) use `xi=-20`
- ii) use `xi = -20.5`
- iii) use `y=math.sqrt(x)` en la `funcion(.)` para obtener otra gráfica
- iv) use `xi=0.5, xf=22.03` y modifique la `funcion(.)` de este modo para implementar otra funcion:

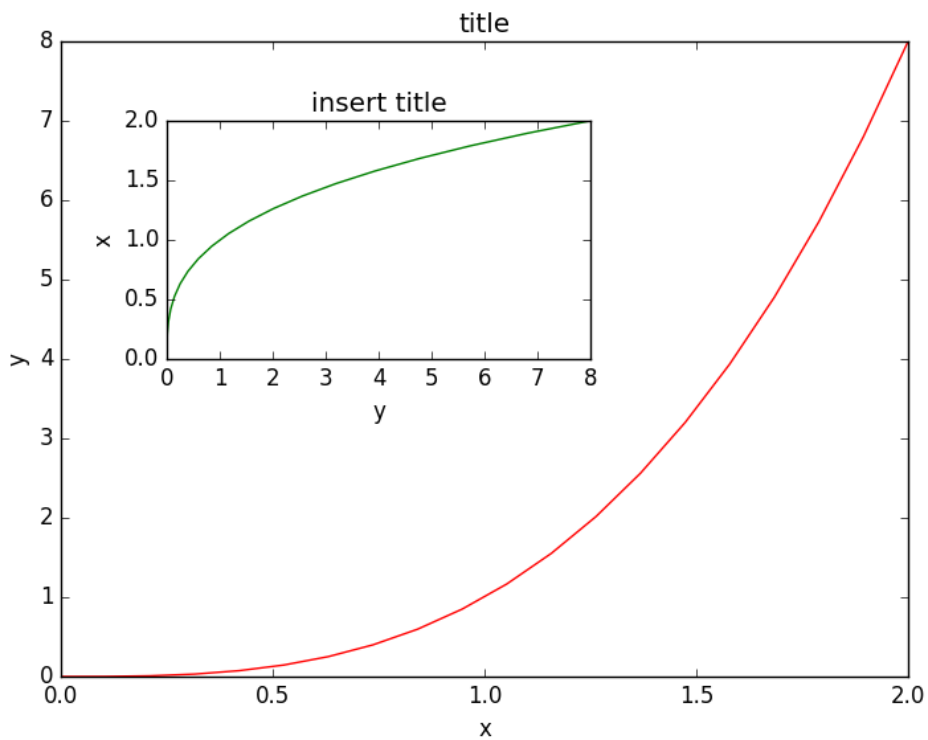
```
def funcion2(xi,xf):  
    x=np.arange(xi,2,xf)  
    y = x*np.sqrt(x)  
    return x,y
```

Observe los resultados y saque conclusiones para discutir en clase

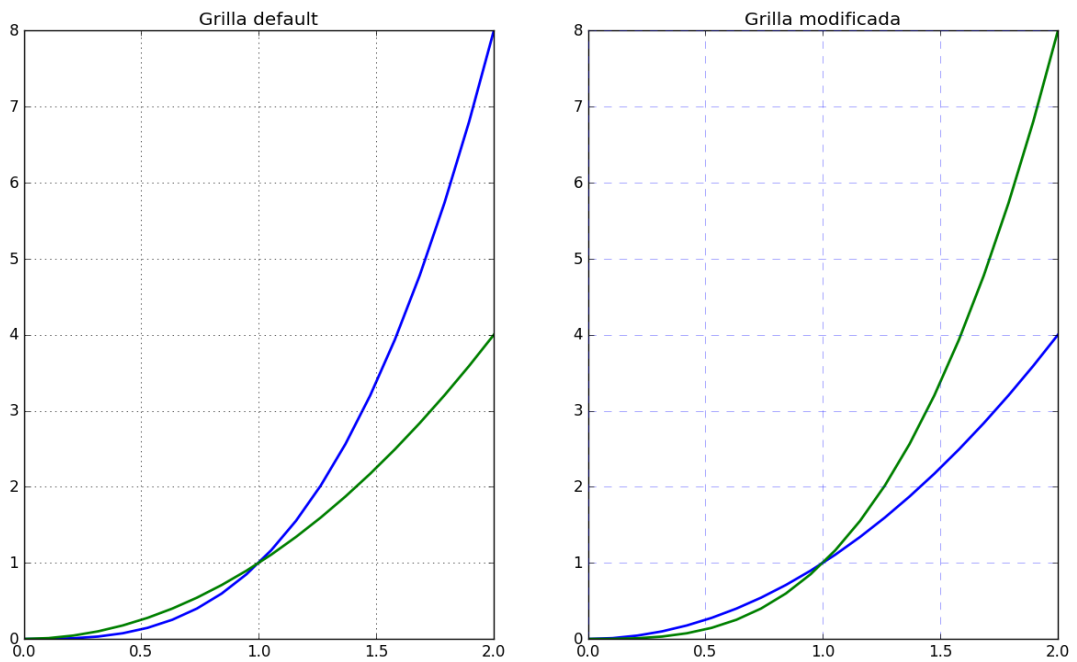
3- Modifique el script y obtenga la siguiente gráfica



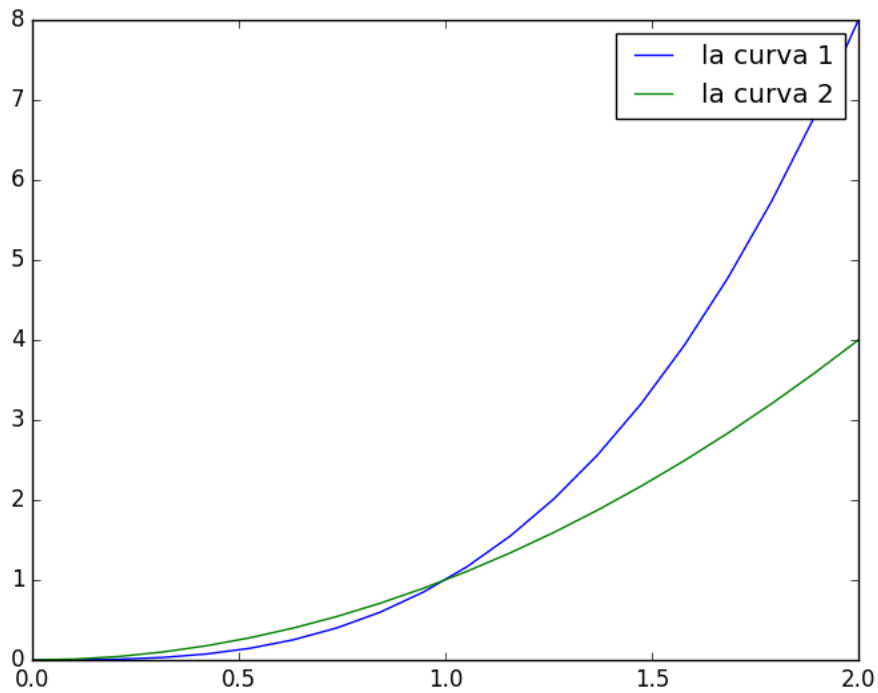
4- Modifique el script y obtenga la siguiente gráfica



5- Modifique el script y obtenga la siguiente gráfica (use $y = x^{**3}$ con $y = x^{**2}$)



6- Modifique el script y obtenga la siguiente gráfica



7- Ejecute y comente claramente los siguientes códigos
Documente con Jupiter los siguientes ejemplos:

https://matplotlib.org/examples/mplot3d/surface3d_radial_demo.html

https://matplotlib.org/examples/animation/double_pendulum_animated.html
