In [ ]:
```python
import numpy as np
from scipy.integrate import odeint
import matplotlib.pyplot as plt

plt.style.use('ggplot')

t = np.linspace(0, 5, 101)
```

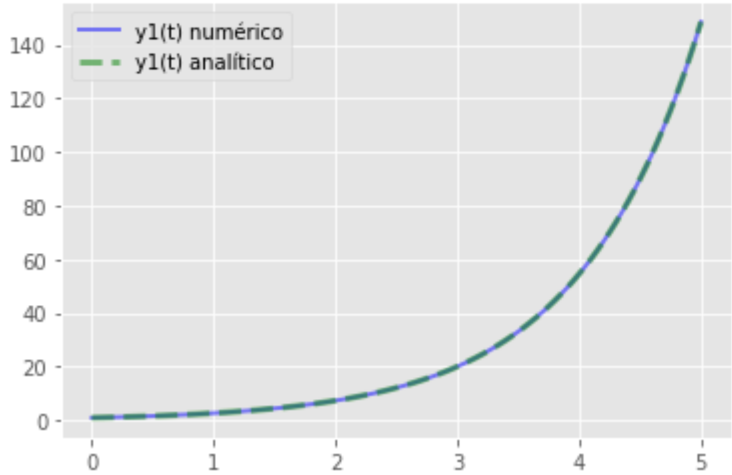
In [ ]:
```python
# 1. y' =y, com y(0) = 1 para encontrar um y(t) muito familiar

y0 = 1

def deriv1(y, t):
    dydt = y
    return dydt

sol1 = odeint(deriv1, y0, t)
eq1 = np.exp(t)

plt.plot(t, sol1, 'b', alpha=0.5, lw=2, label='y1(t) numérico')
plt.plot(t, eq1, 'g--', alpha=0.5, lw=3, label='y1(t) analítico')
plt.legend()
plt.show()
```



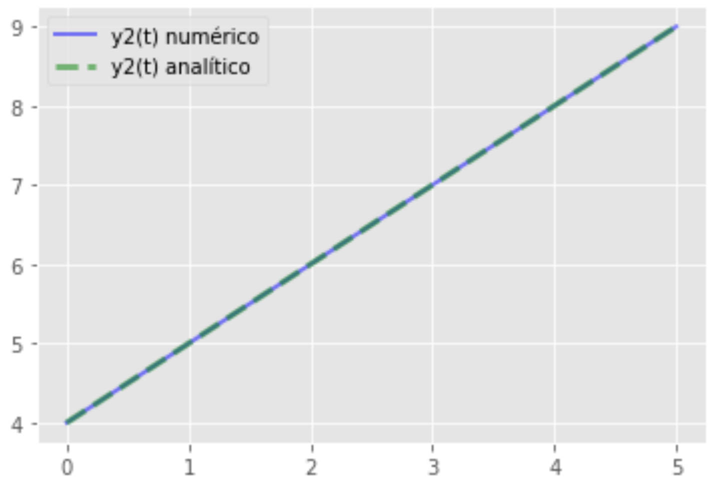
In [ ]:
```python
# 2. y' = 1 com y(0) = 4, uma pergunta trivial

y0 = 4

def deriv2(y, t):
    dydt = 1
    return dydt

sol2 = odeint(deriv2, y0, t)
eq2 = t + 4

plt.plot(t, sol2, 'b', alpha=0.5, lw=2, label='y2(t) numérico')
plt.plot(t, eq2, 'g--', alpha=0.5, lw=3, label='y2(t) analítico')
plt.legend()
plt.show()
```



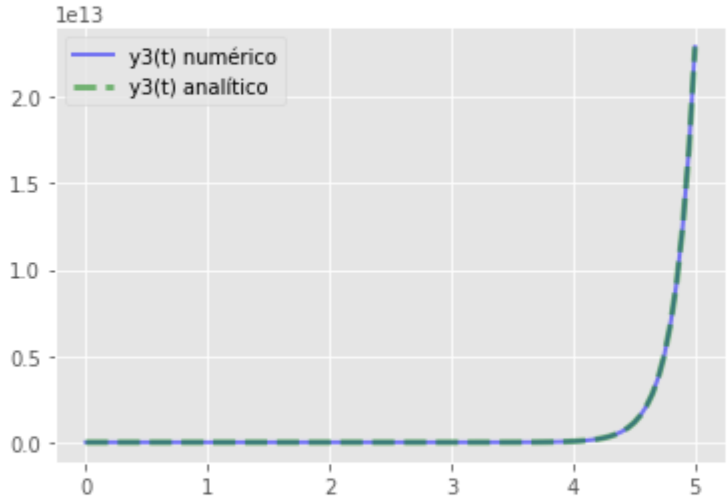
In [ ]:
```
# 3. y' - 6y = exp(-t), y(0) = 2

y0 = 2

def deriv3(y, t):
    dydt = np.exp(-t) + 6 * y
    return dydt

sol3 = odeint(deriv3, y0, t)
eq3 = np.exp(6*t) * (2 + ((1 - np.exp(-7*t)) / 7))

plt.plot(t, sol3, 'b', alpha=0.5, lw=2, label='y3(t) numérico')
plt.plot(t, eq3, 'g--', alpha=0.5, lw=3, label='y3(t) analítico')
plt.legend()
plt.show()
```



In [ ]:
```
# 4. y'' + 9y = 1, y(0) = y'(0) = 0
    # u = y, v = y' -> u' = y', v' = y''
    # u' = v
    # v' = 1 - 9y = 1 - 9u

# u0 = 0
# v0 = 0
y0 = 0, 0

def deriv4(y, t):
    u, v = y
    dudt = v
    dvdt = 1 - 9 * u
    return dudt, dvdt

sol4 = odeint(deriv4, y0, t)
u, v = sol4.T # u = y, v = y'
eq4 = (-np.cos(3*t) + 1) / 9

plt.plot(t, u, 'b', alpha=0.5, lw=2, label='y4(t) numérico')
plt.plot(t, eq4, 'g--', alpha=0.5, lw=3, label='y4(t) analítico')
plt.legend()
plt.show()
```