Lab Assignment 03

1. Solve $\frac{dy(t)}{dt} = -ky(t)$ with parameter k = 0.5 and the initial condition $y_0 = 10$

In [2]:

```
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
from scipy.integrate import odeint
import numpy as np
import sympy as sy
import sympy, math
```

In [6]:

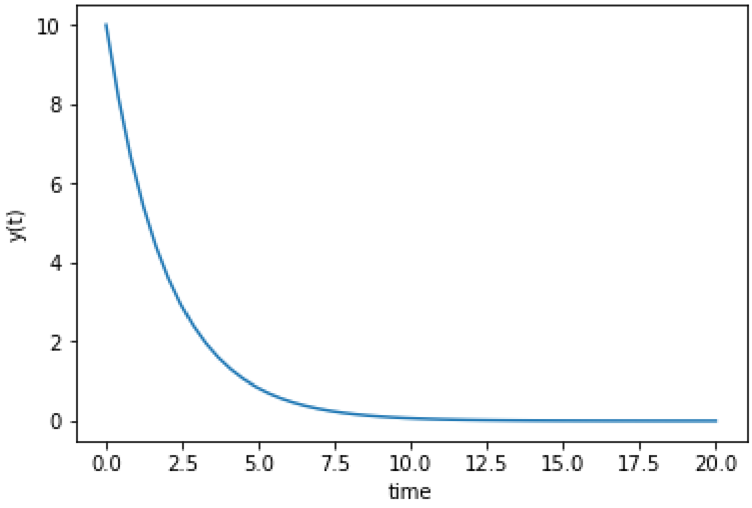
```
#function that returns dy/dt
def model(y,t):
    k = 0.5
    dydt = -k*y
    return dydt

# initial condition
y0=10

# time points
t = np.linspace(0,20)

#solve ODE
y = odeint(model, y0, t)

#plot results
plt.plot(t,y)
plt.xlabel('time')
plt.ylabel('y(t)')
plt.show()
print(y)
```




```
[[1.00000000e+01]
 [8.15395813e+00]
 [6.64870328e+00]
 [5.42132480e+00]
 [4.42052566e+00]
 [3.60447803e+00]
 [2.93907624e+00]
 [2.39651044e+00]
 [1.95410454e+00]
 [1.59336861e+00]
 [1.29922606e+00]
 [1.05938346e+00]
 [8.63816816e-01]
 [7.04352596e-01]
 [5.74326142e-01]
 [4.68303119e-01]
 [3.81852392e-01]
 [3.11360833e-01]
 [2.53882313e-01]
 [2.07014574e-01]
 [1.68798821e-01]
```

```
 [1.37637851e-01]
 [1.12229327e-01]
 [9.15113230e-02]
 [7.46179494e-02]
 [6.08431649e-02]
 [4.96112624e-02]
 [4.04528161e-02]
 [3.29850573e-02]
 [2.68958778e-02]
 [2.19307865e-02]
 [1.78822716e-02]
 [1.45811298e-02]
 [1.18893923e-02]
 [9.69456086e-03]
 [7.90490449e-03]
 [6.44562608e-03]
 [5.25573662e-03]
 [4.28550569e-03]
 [3.49438345e-03]
 [2.84930606e-03]
 [2.32331226e-03]
 [1.89441850e-03]
 [1.54470032e-03]
 [1.25954183e-03]
 [1.02702486e-03]
 [8.37431541e-04]
 [6.82838003e-04]
 [5.56782554e-04]
 [4.53998031e-04]]
```

2. Solve the above problem for k = 0.1; 0.4; 0.6; 0.7 and 0.9

In [4]:

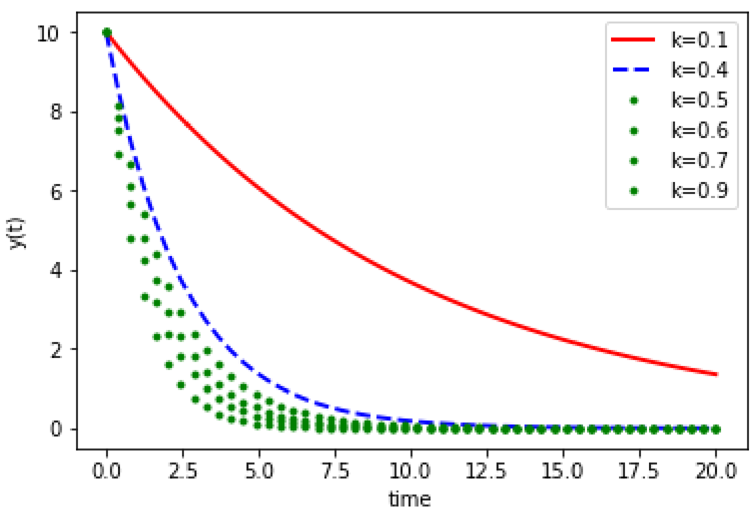
```
#function that returns dy/dt
def model(y,t,k):
    dydt = -k*y
    return dydt

# initial condition
y0=10

# time points
t = np.linspace(0,20)

#solve ODE
k = 0.1
y1 = odeint(model,y0,t,args=(k,))
k = 0.4
y2 = odeint(model,y0,t,args=(k,))
k = 0.5
y3 = odeint(model,y0,t,args=(k,))
k = 0.6
y4 = odeint(model,y0,t,args=(k,))
k = 0.7
y5 = odeint(model,y0,t,args=(k,))
k = 0.9
y6 = odeint(model,y0,t,args=(k,))


#plot results
plt.plot(t,y1,'r-',linewidth=2,label='k=0.1')
plt.plot(t,y2,'b--',linewidth=2,label='k=0.4')
plt.plot(t,y3,'g.',linewidth=2,label='k=0.5')
plt.plot(t,y4,'g.',linewidth=2,label='k=0.6')
plt.plot(t,y5,'g.',linewidth=2,label='k=0.7')
plt.plot(t,y6,'g.',linewidth=2,label='k=0.9')
plt.xlabel('time')
plt.ylabel('y(t)')
plt.legend()
plt.show()
```



3. Solve $\frac{7dy(t)}{dt} = -y(t) + u(t), y(0)$ = 2u steps from 0 to 2 at t =

12

In [8]:

```
#function that returns dy/dt
def model(y,u,t):
    dydt = (-y(t) + u(t))/7
    return dydt

# initial condition
u=np.linspace(0,1,2)
y0=2*u

# time points
t = 12

#solve ODE
y = odeint(model,y0,t)
y
```

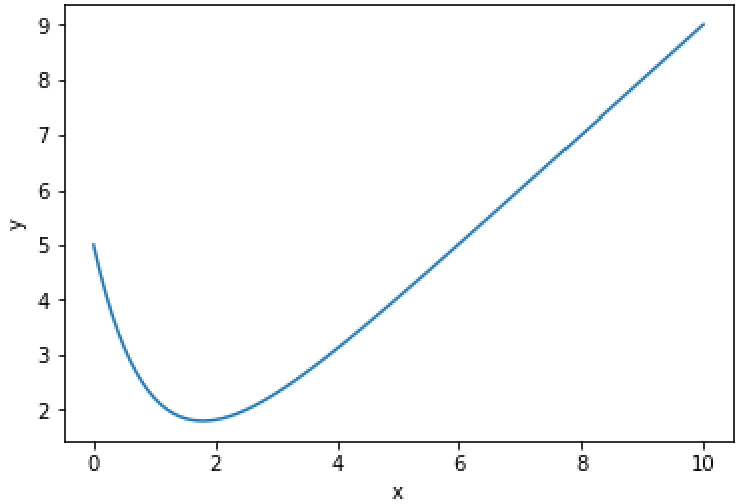
Out[8]:

```
array([[0., 2.]])
```

4. Solve $\frac{dy}{dx} - 2y = 0$ with y(0) = 5

In [20]:

```
#Define a function that calculates the derivative
def dy_dx(y,x):
    return x-y
xs = np.linspace(0,10,100)
y0 = 5.0 #the initial condition
ys = odeint(dy_dx,y0,xs)
#plot results
plt.plot(xs,ys)
plt.xlabel('x')
plt.ylabel('y')
plt.show()
print(ys)
```



In [ ]: