In [0]:
```
# Change directory to VSCode workspace root so that relative path load
s work correctly. Turn this addition off with the DataScience.changeDi
rOnImportExport setting
# ms-python.python added
import os
try:
        os.chdir(os.path.join(os.getcwd(), '../../../..'))
        print(os.getcwd())
except:
        pass
```

Eric Reyes

Sep 2 2019

EE 110

1. The period of the signal should be about 5τ
2. It should be on for 5τ so that you can see the entire response
3. The starting voltage is 0 as all step responses start 0
4. Von of the step should be 1 so its easier to focus on the waveform
5. The rise and fall would ideally be 0 but will still act as a step response as long as
   $t_{rise} \wedge t_{fall} << \tau_c$
   $i.e. \quad t_{rise} \wedge t_{fall} < \tau_c/100$

In [1]:
```
#imports
from sympy.functions.elementary.exponential import exp
import matplotlib.pyplot as plt
import numpy as np
```

Plots for mathematically derived responses

In [2]:
```
tau = 1
y_op_step = np.vectorize(lambda t: 2*(1-exp(-t/tau)))
h_op = np.vectorize(lambda t: 2/tau*exp(-t/tau))
y_op_ramp = np.vectorize(lambda t: 2*(t + tau*exp(-t/tau)))
```

In [3]:
```
plt.rcParams["figure.figsize"] = (20,3)
t_range = np.arange(0,5*tau,.001)
plt.subplot(1,3,1)
plt.plot(t_range,y_op_step(t_range))
plt.xlabel("Time")
plt.xlabel("Voltage")
plt.title("Step Response")
plt.subplot(1,3,2)
plt.plot(t_range,h_op(t_range))
plt.xlabel("Time")
plt.xlabel("Voltage")
plt.title("Impulse Response")
plt.subplot(1,3,3)
plt.plot(t_range,y_op_ramp(t_range))
plt.xlabel("Time")
plt.xlabel("Voltage")
plt.title("Ramp Response")
```

Out[3]:
```
Text(0.5, 1.0, 'Ramp Response')
```