```
import numpy as np
import matplotlib.pyplot as plt
```

```
x = np.arange(-5.0 , 5.0 , 0.1)
```

```
x.shape
```

```
(100,)
```

Step function (계단함수)

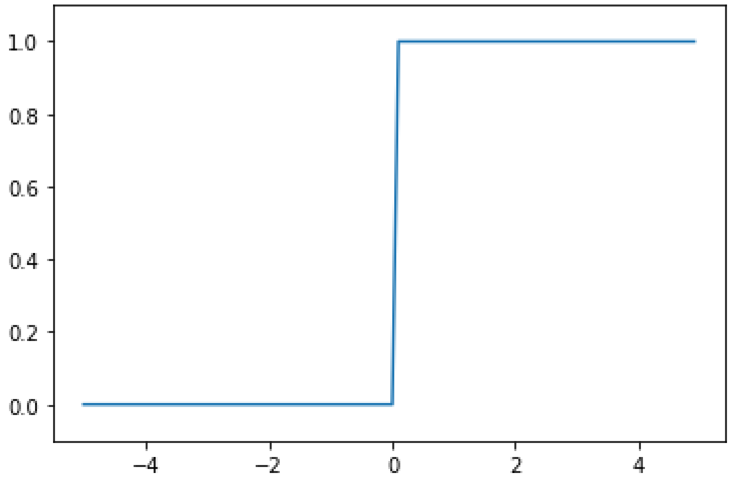
```
def step_function(x):
  return np.array(x>0, dtype=np.int)
```

```
y = step_function(x)
```

```
y.shape
```

```
(100,)
```

```
plt.plot(x,y)
plt.ylim(-0.1, 1.1)
plt.show()
```



```
plt.plot(x,y)
plt.show()
```

Sigmoid

$$f(x) = \frac{1}{1 + e^{-x}}$$

```
def sigmoid(x):
    return 1 / (1 +np.exp(-x))
```

```
x = np.arange(-5.0, 5.0, 0.1)
y2 = sigmoid(x)
```

```
plt.plot(x,y2)
plt.grid()
plt.show()
```

- 일반적인 form

$$f(x) = \frac{1}{1 + e^{-c_1 x}}$$

```
def sigmoid2(x, c1=1):
    return 1/ (1 +np.exp(-c1 * x))
```

```
y2 = sigmoid2(x, 2)
```

```
plt.plot(x, y2)
plt.grid()
```

```
for i in range(8):
  c1 = 0.2 * (i + 1)
  y2 = sigmoid2(x, c1)
  plt.plot(x, y2)
plt.grid()
```

- 보다 일반적인 form

$$f(x) = \frac{1}{1 + e^{-c_1(x - c_2)}}$$

```
def sigmoid3(x, c1=1, c2=0):
    return 1 / (1 +np.exp(-c1 * (x -c2)))
```

```
y3 = sigmoid3(x)
```

```
plt.plot(x, y3)
plt.grid()
```

```
for i in range(10):
  c2 = 0.3 * i
  c1 = 2
  y3 = sigmoid3(x, c1,c2)
  plt.plot(x, y3)
plt.grid()
```