In [5]:

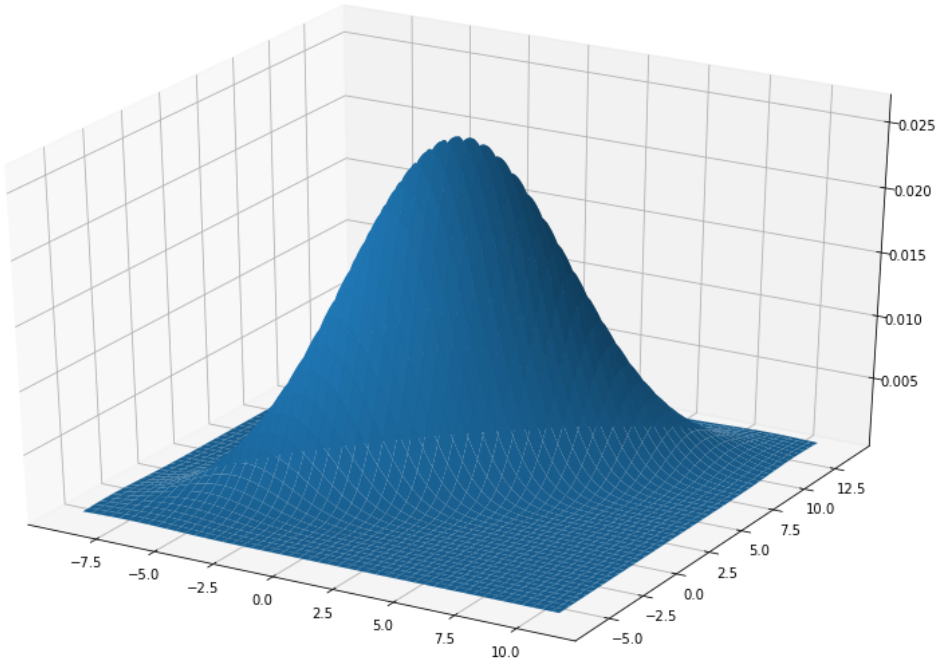
```
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
import numpy as np
from scipy.stats import multivariate_normal
plt.rcParams['figure.figsize'] = (15, 10)
```

In [37]:

```
fig = plt.figure()
ax = fig.gca(projection='3d')

rv = multivariate_normal([1, 4], [[10, 8], [8, 10]])
x, y = np.meshgrid(np.arange(-9, 11, 0.1), np.arange(-6, 14, 0.1))
z = np.zeros_like(x)
for i in range(x.shape[0]):
    for j in range(x.shape[1]):
        z[i][j] = rv.pdf((x[i][j], y[i][j]))

ax.plot_surface(x, y, z)
plt.show()
```



$$f(x,y) = \frac{1}{18\pi} e^{-\frac{1}{36}(5x^2-8xy+5y^2-32y+22x+53)}$$

$$f(x|y) = \frac{f(x,y)}{\int f(x,y)dx}$$

$$f(y) = \frac{1}{2\sqrt{5\pi}} e^{-\frac{1}{20}(y-4)^2}$$

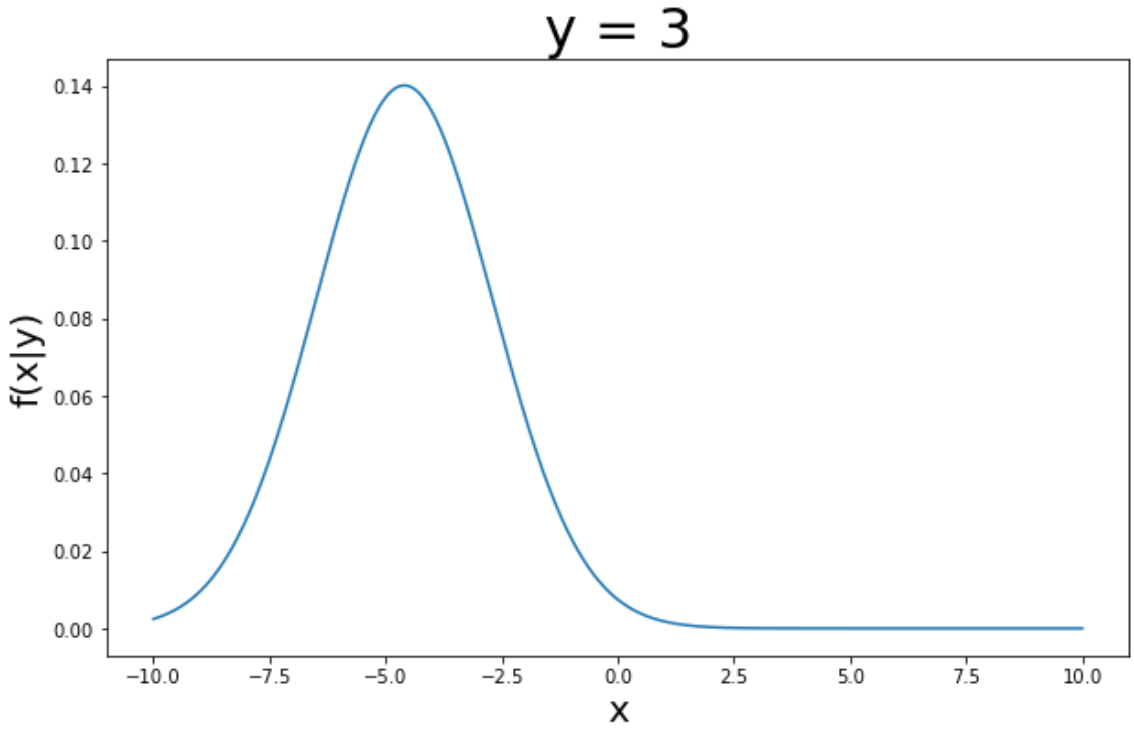
Условные распределения

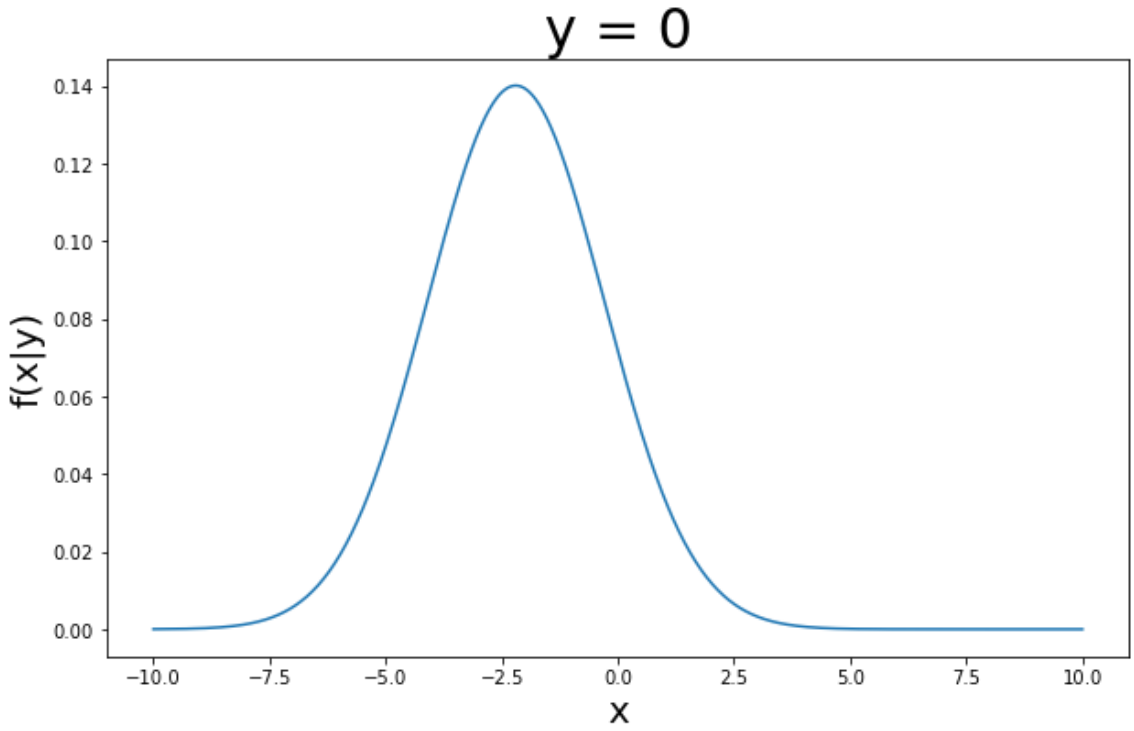
In [30]:

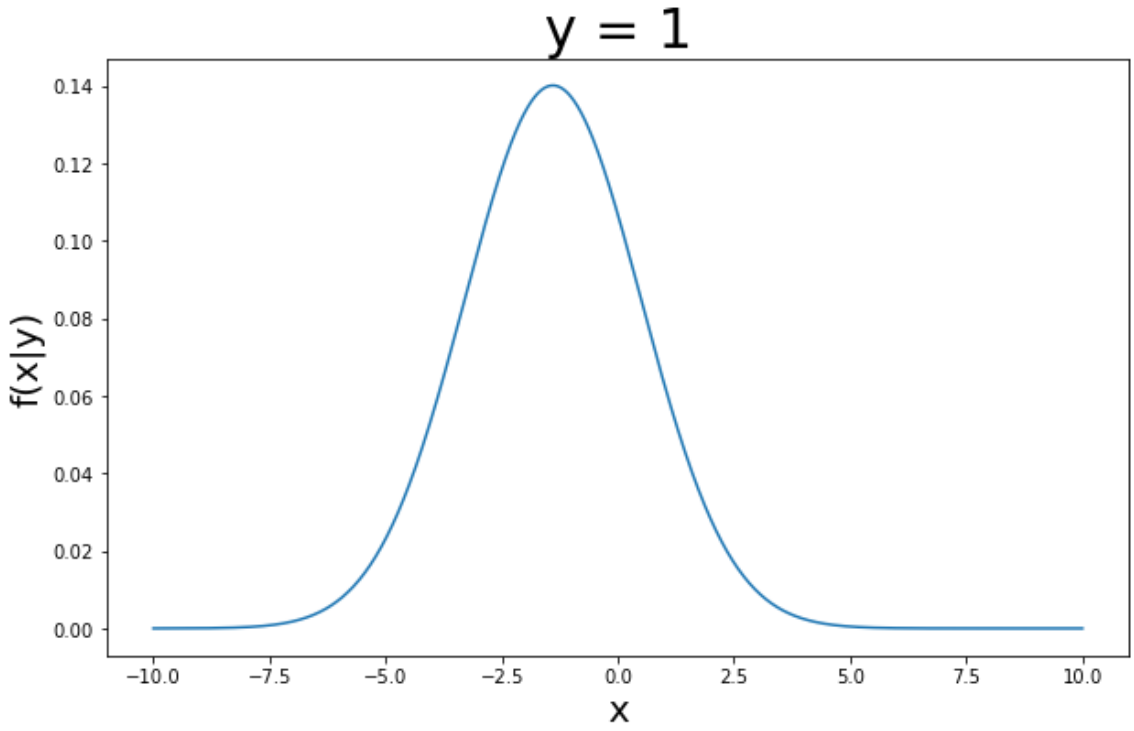
```
def cond_f(x, y):
    return 1.0/18/np.pi*np.exp(-1./36*(5*x**2 - 8*x*y + 5*y**2 -32*y + 22*x +
53))*2*np.sqrt(5*np.pi)/np.exp(- 1./20*(y-4)**2)
```

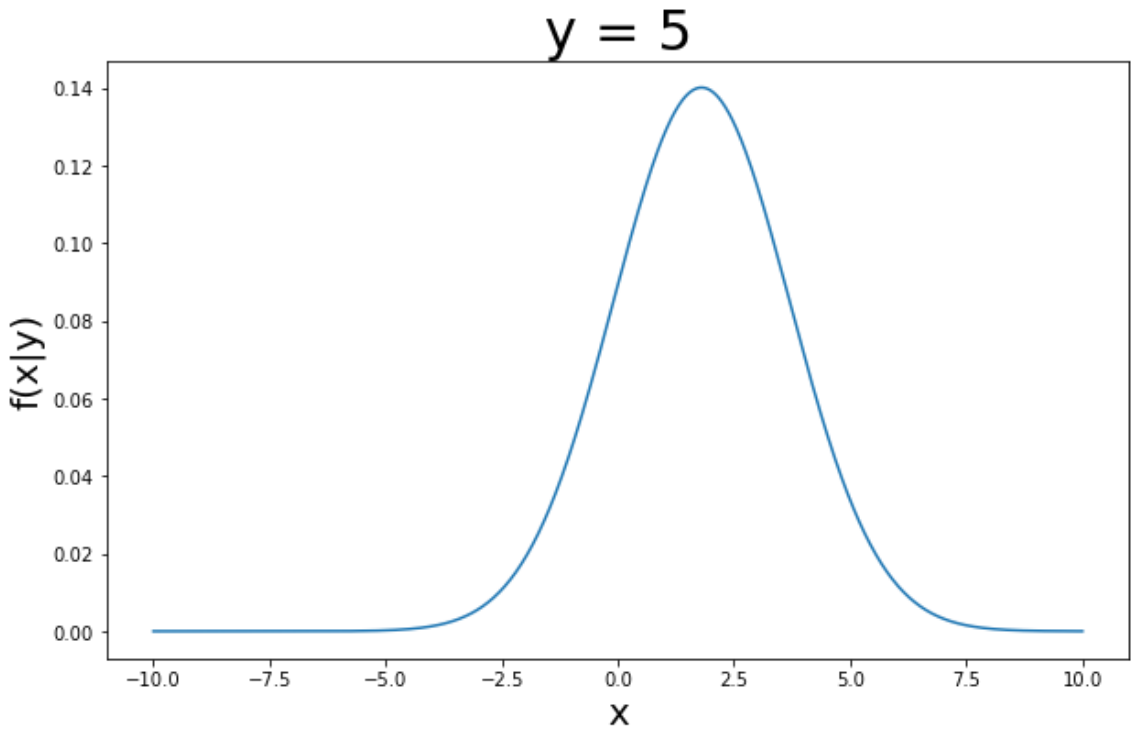
In [36]:

```
x = np.linspace(-10, 10, 1000)
plt.rcParams['figure.figsize'] = (10, 6)
plt.plot(x, cond_f(x, -3.))
plt.xlabel('x', fontsize=20)
plt.ylabel('f(x|y)', fontsize=20)
plt.title('y = 3', fontsize=30)
plt.show()
plt.plot(x, cond_f(x, 0.))
plt.xlabel('x', fontsize=20)
plt.ylabel('f(x|y)', fontsize=20)
plt.title('y = 0', fontsize=30)
plt.show()
plt.plot(x, cond_f(x, 1.))
plt.xlabel('x', fontsize=20)
plt.ylabel('f(x|y)', fontsize=20)
plt.title('y = 1', fontsize=30)
plt.show()
plt.plot(x, cond_f(x, 5.))
plt.xlabel('x', fontsize=20)
plt.ylabel('f(x|y)', fontsize=20)
plt.title('y = 5', fontsize=30)
plt.show()
```


y = 3
f(x|y)
x
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0


y = 0
f(x|y)
x
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
−10.0
−7.5
−5.0
−2.5
0.0
2.5
5.0
7.5
10.0





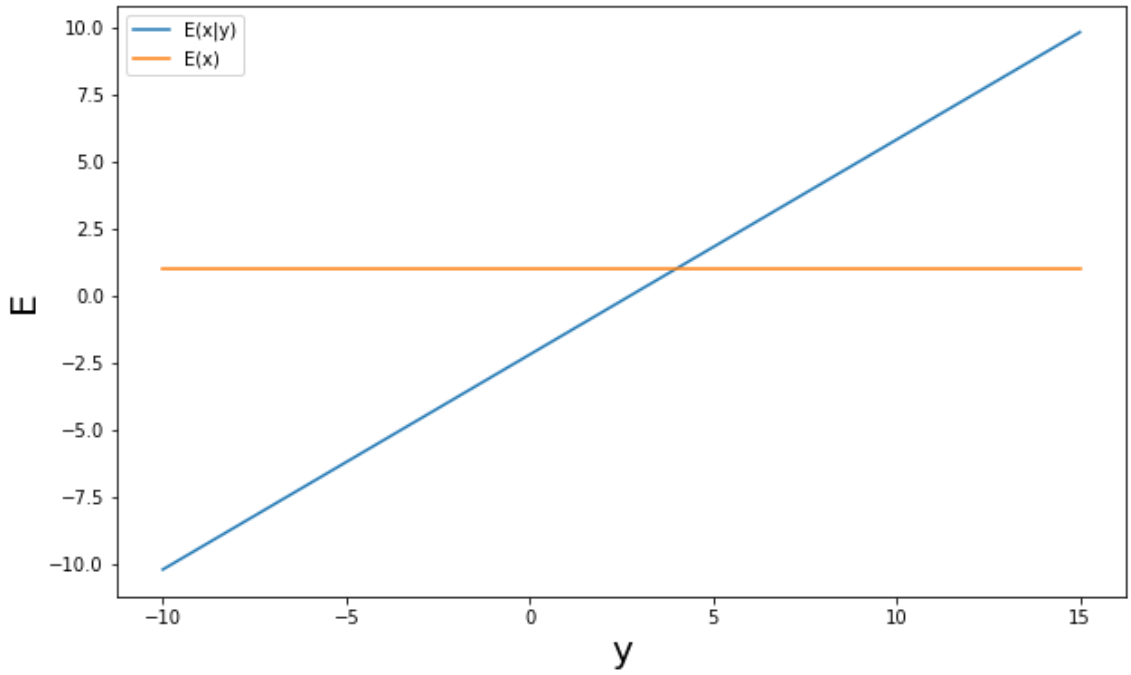
Условное матожидание

$E(x|y) = \frac{4y-11}{5}$

$E(x) = 1$

In [43]:

```
y = np.linspace(-10, 15, 1000)
plt.plot(y, (4*y - 11)/5., label='E(x|y)')
plt.plot(y, np.ones_like(y), label='E(x)')
plt.xlabel('y', fontsize=20)
plt.ylabel('E', fontsize=20)
plt.legend()
plt.show()
```



In [ ]: