In [2]:

```
import numpy as np
import scipy.stats as sps
import matplotlib.pyplot as plt
%pylab inline
```

```
Populating the interactive namespace from numpy and matplotlib
```

Построим график плотности случайного вектора $\xi = (\xi_1, \xi_2) \sim N(a, \Sigma)$, где $a = \begin{pmatrix} 1 \\ 4 \end{pmatrix}$, $\Sigma = \begin{pmatrix} 10 & 8 \\ 8 & 10 \end{pmatrix}$

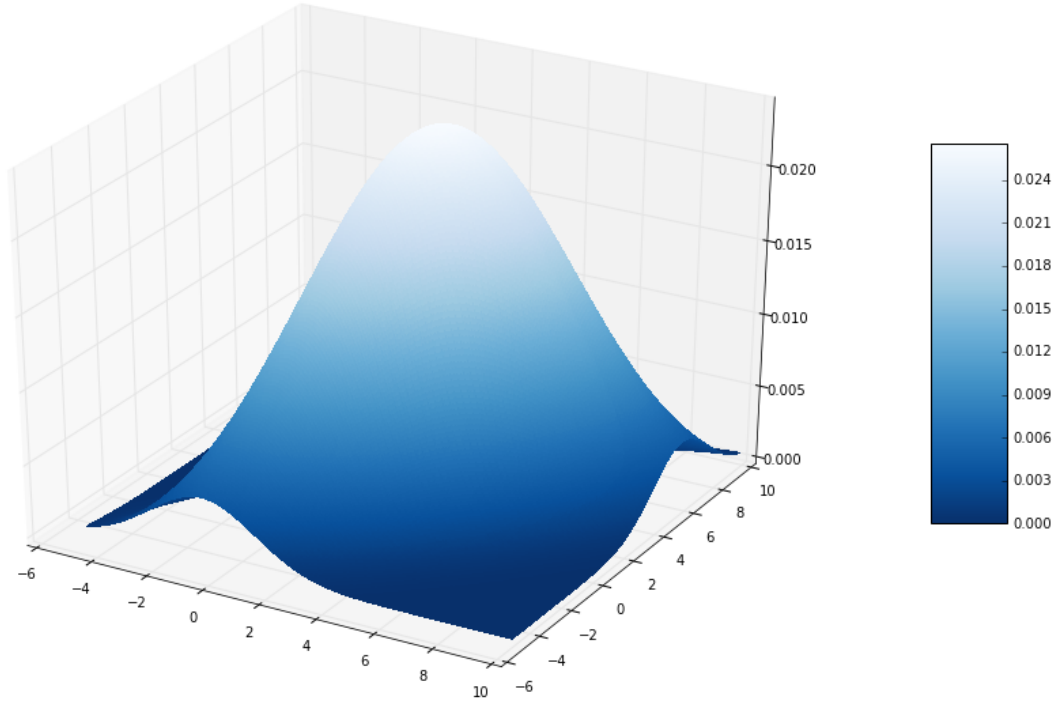
In [18]:

```
from mpl_toolkits.mplot3d import Axes3D
```

In [25]:

```
grid = np.mgrid[-5:10:0.05, -5:10:0.05]
density = np.array([[sps.multivariate_normal.pdf((grid[0, i, j], grid[1, i, j]), mean
                        for i in range(grid[0].shape[0])]
                       for j in range(grid[0].shape[1])])

fig = plt.figure(figsize=(16, 10))
ax = fig.gca(projection='3d')
surf = ax.plot_surface(grid[0], grid[1], density, rstride=1, cstride=1, cmap='Blues_r
                       linewidth=0, antialiased=False)
ax.set_zlim(0, 0.024)
fig.colorbar(surf, shrink=0.5, aspect=5)
plt.show()
```



Графики зависимости $f_{(\xi_1|\xi_2)}(x|y)$ от x

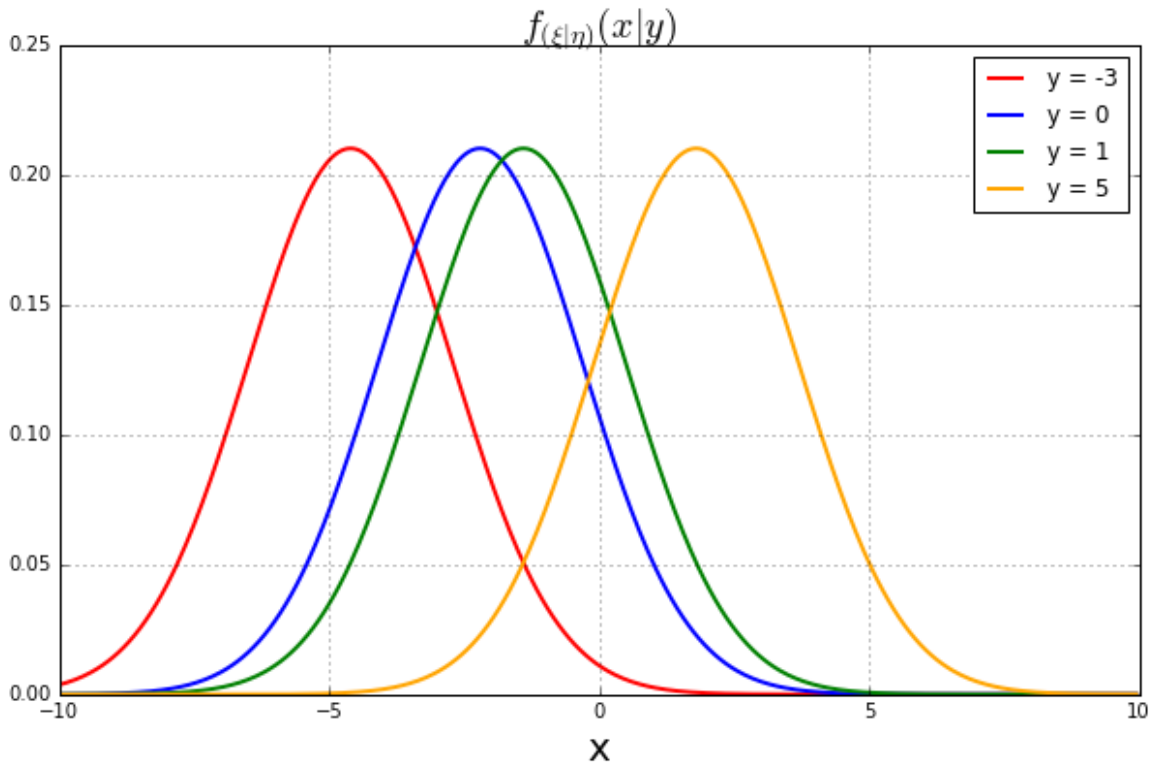
In [31]:

```
from math import sqrt
```

In [47]:

```
y = np.array([-3, 0, 1, 5])
x = np.arange(-10, 10, 0.05)
colors = ['red', 'blue', 'green', 'orange']

plt.figure(figsize=(10, 6))
for i in range(y.shape[0]):
    conditional_density = np.array([sps.multivariate_normal.pdf((x[j], y[i]), mean=[1
                      for j in range(x.shape[0])]) / sps.norm.pdf(y[i], loc = 4, scal
    plt.plot(x, conditional_density, color=colors[i], linewidth=2, label='y = '+str(y
plt.legend()
plt.xlim((-10, 10))
plt.ylim((0, 0.25))
plt.xlabel("x", fontsize = 20)
plt.title(r'$f_{(\xi_1 | \xi_2)}(x | y)$', fontsize = 20)
plt.grid()
```



In [ ]:

```
```