In [1]:

```
%matplotlib inline
# Importing standard Qiskit libraries and configuring account
from qiskit import QuantumCircuit, execute, Aer, IBMQ
from qiskit.compiler import transpile, assemble
from qiskit.tools.jupyter import *
from qiskit.visualization import *
# Loading your IBM Q account(s)
provider = IBMQ.load_account()
```

```
/opt/conda/lib/python3.7/site-packages/qiskit/providers/ibmq/ibmqfactory.
py:192: UserWarning: Timestamps in IBMQ backend properties, jobs, and job
results are all now in local time instead of UTC.
  warnings.warn('Timestamps in IBMQ backend properties, jobs, and job res
ults '
```

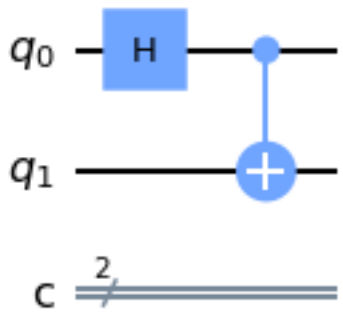
In [2]:

```
from qiskit import QuantumRegister, ClassicalRegister, QuantumCircuit
from numpy import pi

qreg_q = QuantumRegister(2, 'q')
creg_c = ClassicalRegister(2, 'c')
circuit = QuantumCircuit(qreg_q, creg_c)

circuit.h(qreg_q[0])
circuit.cx(qreg_q[0], qreg_q[1])
```

```
<qiskit.circuit.instructionset.InstructionSet at 0x7f3afdf71690>
```

In [3]:

```
circuit.draw()
```



In [4]:

```
# Let's see the result:
backend = Aer.get_backend('statevector_simulator')
final_state = execute(circuit,backend).result().get_statevector()
# Print the statevector neatly:
from qiskit_textbook.tools import array_to_latex
array_to_latex(final_state, pretext="\\text{Statevector = }")
```

$$\text{Statevector} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ 0 \\ \frac{1}{\sqrt{2}} \end{bmatrix} \text{Statevector} = \begin{bmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ 0 \\ \frac{1}{\sqrt{2}} \end{bmatrix}$$

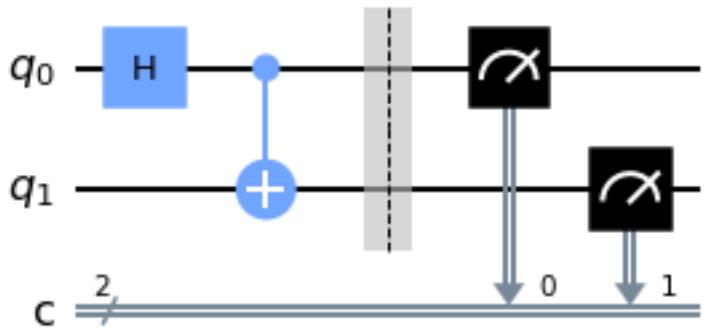
In [6]:

```
# Simulate the unitary
backend = Aer.get_backend('unitary_simulator')
unitary = execute(circuit,backend).result().get_unitary()
# Display the results:
array_to_latex(unitary, pretext="\\text{Circuit = } ")
```

$$\text{Circuit} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 \end{bmatrix} \text{Circuit} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 & 0 \end{bmatrix}$$

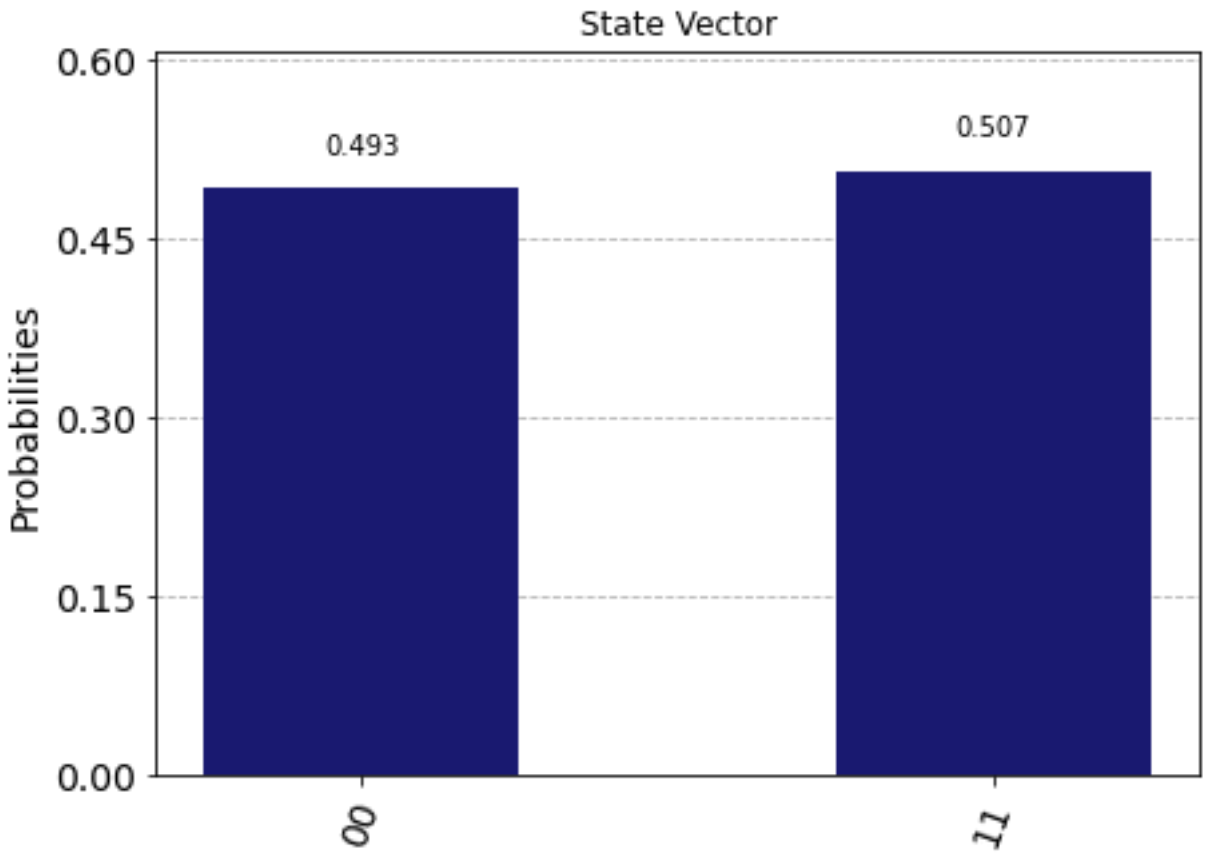
In [7]:

```
circuit.barrier()
circuit.measure(qreg_q[0], creg_c[0])
circuit.measure(qreg_q[1], creg_c[1])
circuit.draw()
```




In [8]:

```
job = execute(circuit, Aer.get_backend('qasm_simulator'), shots=1000)
plot_histogram(job.result().get_counts(), color='midnightblue', title="State Vect
or")
```

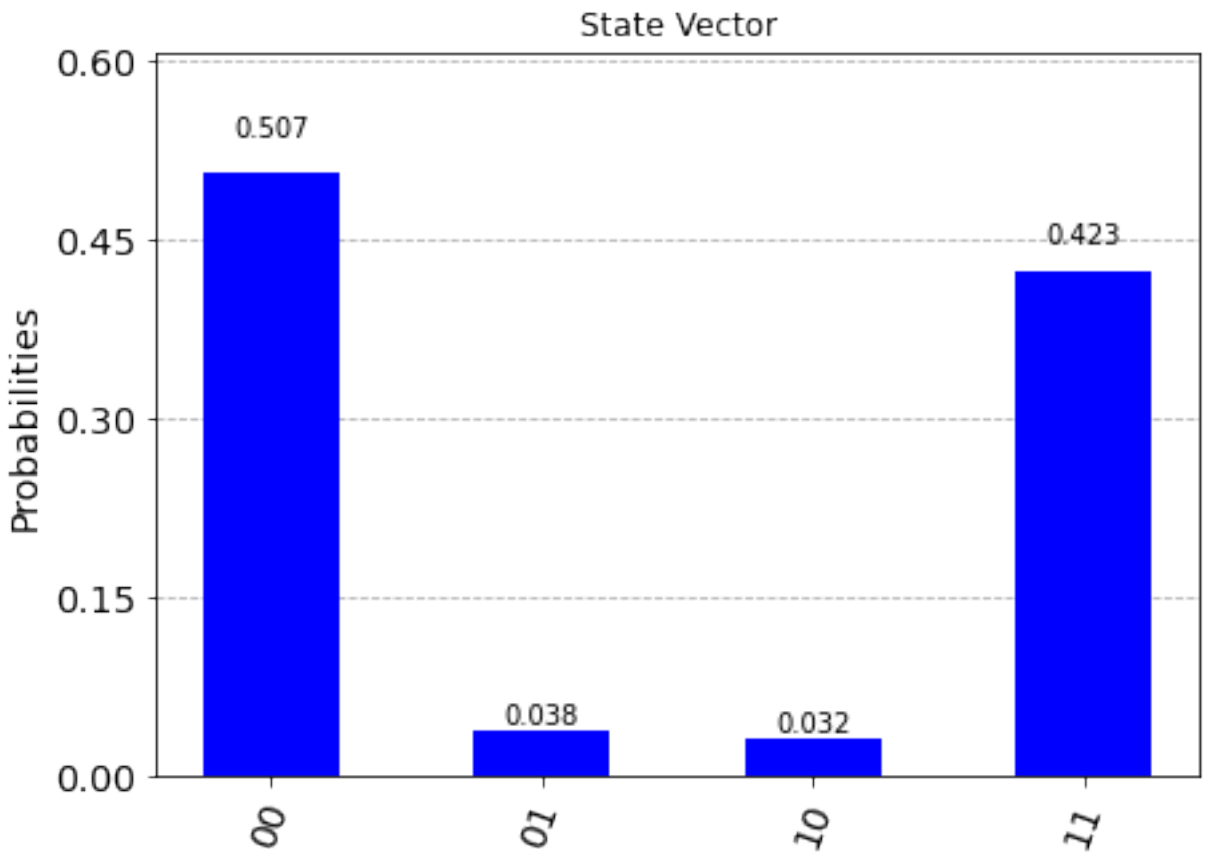



In [10]:

```
device = IBMQ.get_provider(hub='ibm-q',group='open',project='main').get_backend('
ibmq_16_melbourne')
```

In [11]:

```
jobdevice = execute(circuit, device, shots=1024)
plot_histogram(jobdevice.result().get_counts(), color='blue', title="State Vector
")
```




In [ ]: