In [1]:
```
from qiskit import *
from math import pi,sqrt
from qiskit.visualization import plot_bloch_multivector,plot_histogram
```

In [2]:
```
backend_s = Aer.get_backend('statevector_simulator')
backend_q = Aer.get_backend('qasm_simulator')
```

Single Qubit Gates

Solutions

2. Digression: The X, Y & Z bases

Quick Exercises

1. Verify that $|+\rangle$ and $|-\rangle$ are in fact eigenstates of the X-gate

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

$$\begin{vmatrix} -\lambda & 1 \\ 1 & -\lambda \end{vmatrix} = 0 \implies \lambda^2 - 1 = 0 \Rightarrow \lambda = \pm 1$$

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix}$$

$\Rightarrow a = b$ and $b = a$

Hence one eigenstate is $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$.

Similarly

$$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = (-1) \begin{bmatrix} a \\ b \end{bmatrix}$$

$\Rightarrow a = -b$ and $b = -a$

The other eigenstate is $\begin{bmatrix} 1 \\ -1 \end{bmatrix}$

These eigen states are the bases $|+\rangle$ and $|-\rangle$

2. What Eigenvalues do they have?

As evident above, the $|+\rangle$ state corresponds to eigenvalue $(+1)$, and $|-\rangle$ corresponds to eigenvalue (-1)

3. Why would we not see these eigenvalues appear on the Bloch Sphere?

These eigenvalues are what is called global phase on a state. We can only distinguish states on the Bloch sphere up to a global phase. Thus $2|+\rangle$ and $1249|+\rangle$ will have the same vector representation on the Bloch sphere.

4. Find the eigenstates of the Y-gate, and their co-ordinates on the Bloch sphere.

$$Y = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$$

$$\begin{vmatrix} -\lambda & -i \\ i & -\lambda \end{vmatrix} = 0 \implies \lambda^2 + i^2 = \lambda^2 - 1 = 0 \Rightarrow \lambda = \pm 1$$

$$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix}$$

$\Rightarrow -ib = a$ and $ia = b$

Hence one eigenstate is $|\circlearrowleft\rangle = \begin{bmatrix} 1 \\ i \end{bmatrix}$.

Similarly

$$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = (-1) \begin{bmatrix} a \\ b \end{bmatrix}$$

$\Rightarrow ib = a$ and $ia = -b$

The other eigenstate is $|\circlearrowright\rangle = \begin{bmatrix} 1 \\ -i \end{bmatrix}$.

3. The Hadamard Gate

Quick Exercises

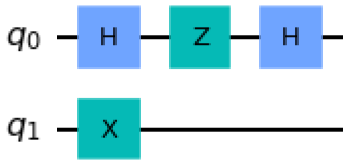
1. Write the Hadamard Gate as the outer product of the vectors $|0\rangle, |1\rangle, |+\rangle$ and $|-\rangle$

$\left(|0\rangle\langle+|\right)+\left(|1\rangle\langle-|\right)$

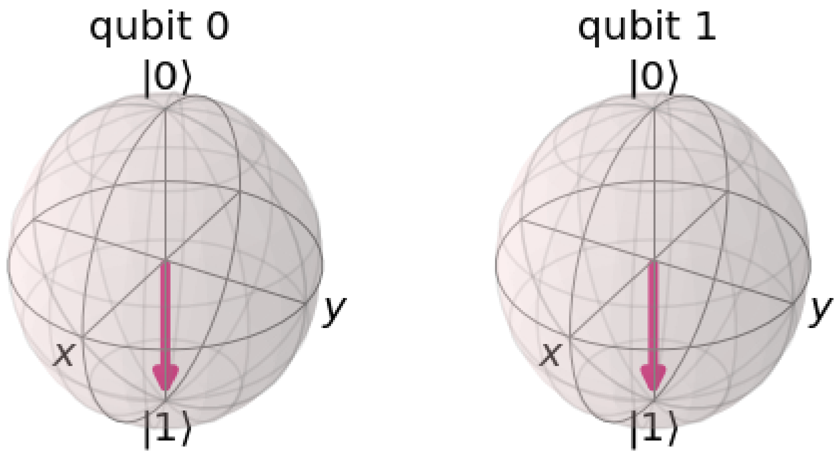
2. Show that applying the sequence of gates: HZH, to any qubit state is equivalent to applying an X-gate.

In [3]:

```
qc = QuantumCircuit(2)
qc.h(0)
qc.z(0)
qc.h(0)
qc.x(1)

display(qc.draw('mpl'))

# we use the statevector_simulator to obtain the statevector
job = execute(qc,backend_s)
job_result = job.result()
state_vec = job_result.get_statevector()
plot_bloch_multivector(state_vec)
```



Out[3]:



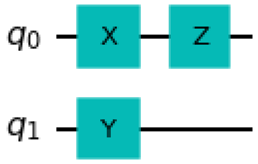
We obtain the same state.

3. Find a combination of X, Z and H-gates that is equivalent to a Y-gate (ignoring global phase)

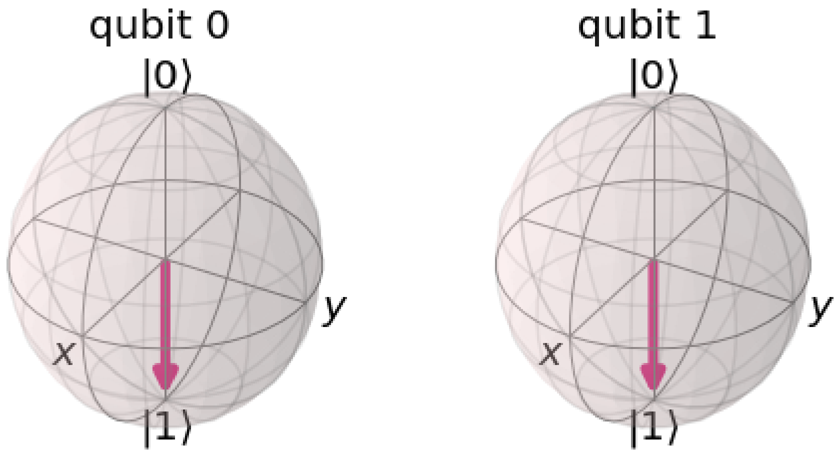
iXZ gives Y. but i is part of the global phase, so XZ is equivalent to a Y gate, as seen below:

In [4]:

```
qc = QuantumCircuit(2)
qc.x(0)
qc.z(0)
qc.y(1)

display(qc.draw('mpl'))

job = execute(qc,backend_s)
job_result = job.result()
state_vec = job_result.get_statevector()
plot_bloch_multivector(state_vec)
```



Out[4]:



4. Digression: Measuring in Different Bases

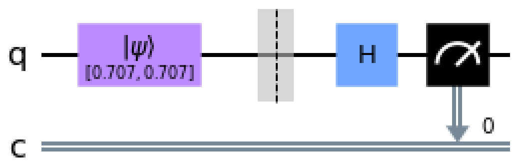
Quick Exercises

1. If we initialise our qubit in the state $|+\rangle$, what is the probability of measuring it in state $|-\rangle$?

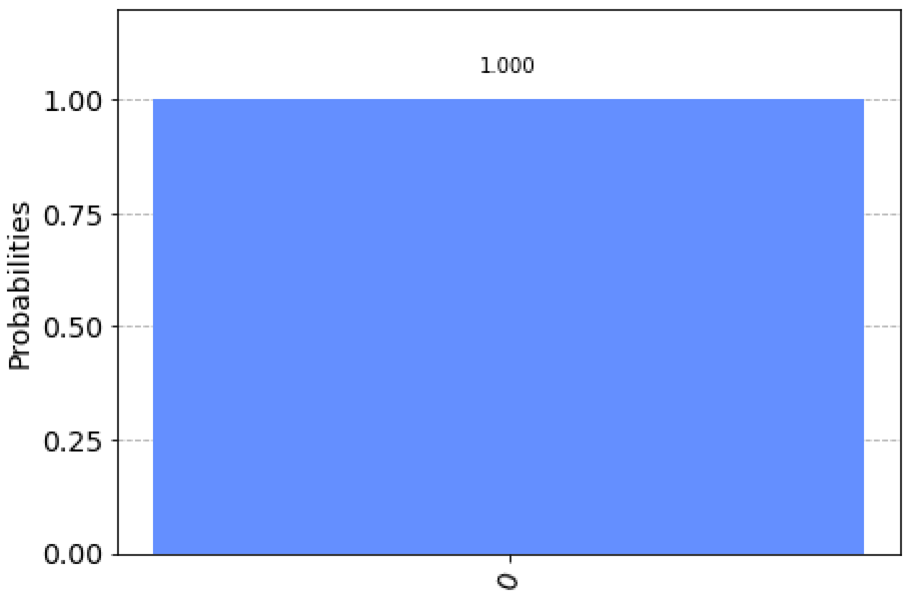
We are expect measuring $|-\rangle$ to be an impossibility. To measure a qubit initialized in the $|+\rangle$ state, we will have to bring it to the Z (0-1) basis, by applying a Hadamard gate and then measuring. If our measurement outcome is 1, then it means our initial state must have been $|-\rangle$ that particular percentage of times.

In [5]:

```
qc = QuantumCircuit(1,1)
# initialize qubit in state |+>
qc.initialize([1/sqrt(2),1/sqrt(2)],0)
qc.barrier()
qc.h(0)
qc.measure(0,0)

display(qc.draw('mpl'))

job = execute(qc, backend_q)
job_result = job.result()
res_counts = job_result.get_counts()
plot_histogram(res_counts)
```



Out[5]:

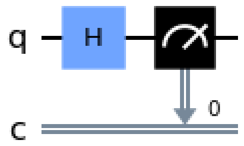


The probability of measuring it in the state $|-\rangle$ is 0, since all our measurements return 0, which correspond to the $|+\rangle$ state.

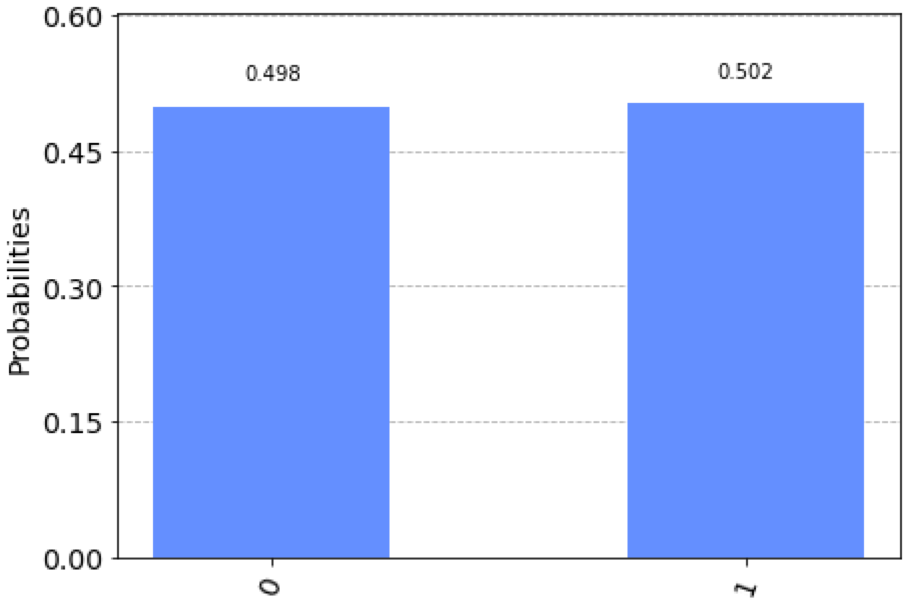
2. Use Qiskit to display the probability of measuring a $|0\rangle$ qubit in the states $|+\rangle$ and $|-\rangle$

In [6]:

```python
# initialize in |+> state:

qc = QuantumCircuit(1,1)
qc.h(0)
qc.measure(0,0)

display(qc.draw('mpl'))

job = execute(qc, backend_q,shots=4000)
job_result = job.result()
res_counts = job_result.get_counts()
plot_histogram(res_counts)
```



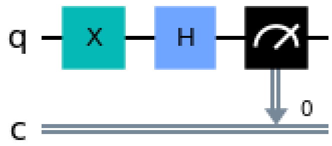
Out[6]:



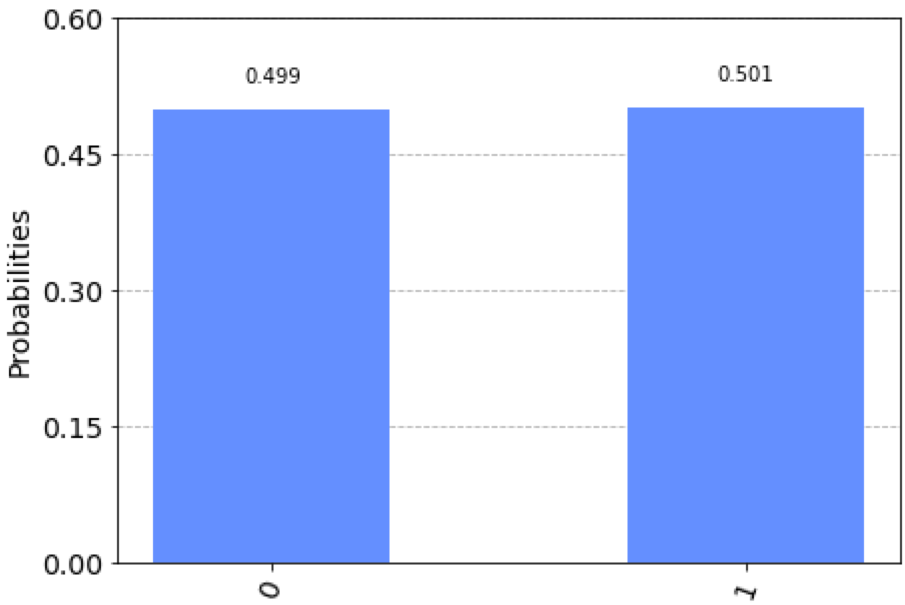
We get $|0\rangle$ around half the times and $|1\rangle$ remaining times.

In [7]:

```
qc = QuantumCircuit(1,1)
qc.x(0)
qc.h(0)
qc.measure(0,0)

display(qc.draw('mpl'))

job = execute(qc, backend_q,shots=4000)
job_result = job.result()
res_counts = job_result.get_counts()
plot_histogram(res_counts)
```



Out[7]:



3. Try to create a function that measures in the Y-basis.

A function that creates in the Y basis should manipulate the state so that it goes from being a Y eigenstate to a corresponding Z eigenstate, i.e. change Y basis to Z basis.
We have seen above what the Y-basis vectors are:

$|\circlearrowleft\rangle = \begin{bmatrix} 1 \\ i \end{bmatrix}$ and $|\circlearrowright\rangle = \begin{bmatrix} 1 \\ -i \end{bmatrix}$.

Our operation will be:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} 1 \\ i \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

and

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} \begin{bmatrix} 1 \\ -i \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

We have four variables, and 4 equations, solving which, we will get the following as our operation matrix:

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & -i \\ 1 & i \end{bmatrix}$$

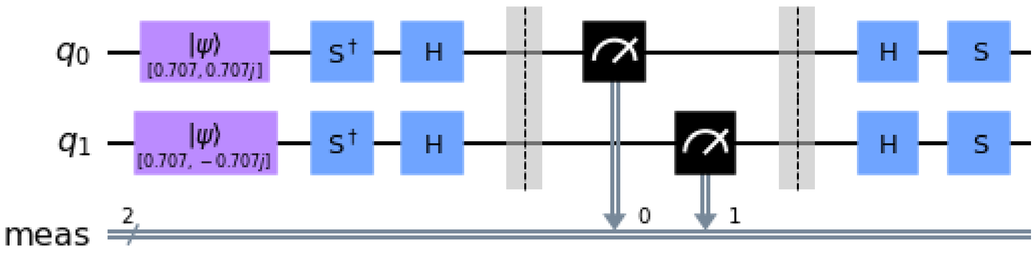
We can multiply and verify that this matrix when multiplied by the Y-basis vectors will tranform them into the corresponding Z-basis states. We can then measure and reapply this state to get it back in the Y-basis state, as the textbook demonstrated for the X-basis.

Going a little ahead in the textbook, the $S^\dagger$ gate is introduced. The operation $HS^\dagger$ gives us the above matrix, so measuring in the Y basis will look like-

In [8]:

```
qc = QuantumCircuit(2)
qc.initialize([1/sqrt(2),1.j/sqrt(2)],0)
qc.initialize([1/sqrt(2),-1.j/sqrt(2)],1)
qc.sdg([0,1])
qc.h([0,1])
qc.measure_all()
qc.barrier()
qc.h([0,1])
qc.s([0,1])
qc.draw('mpl')
```

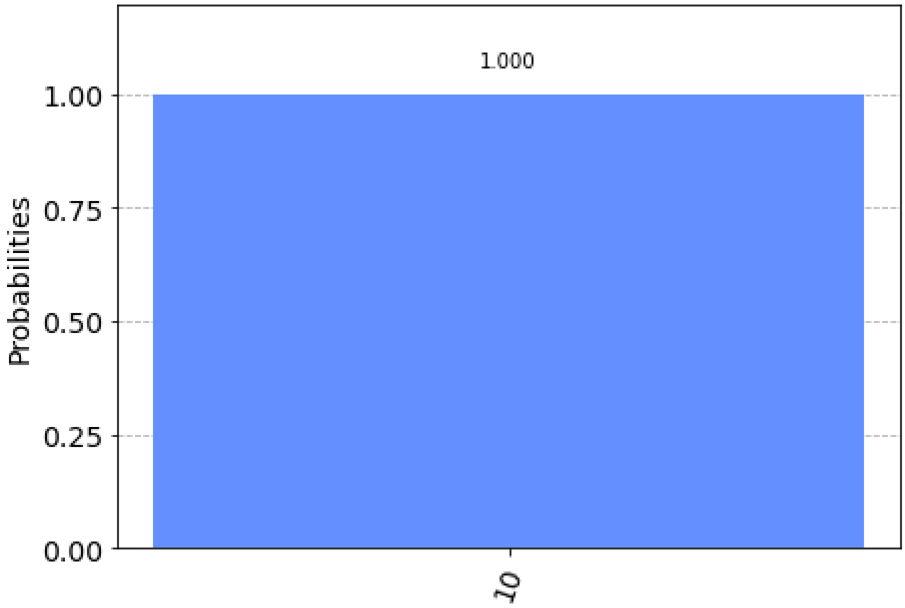
Out[8]:



Why S is used to recreate the state can be understood in the section that deals with the $S-$ and $S^\dagger-$gate.

In [9]:
```
job = execute(qc,backend_q).result()
counts = job.get_counts()
plot_histogram(counts)
```

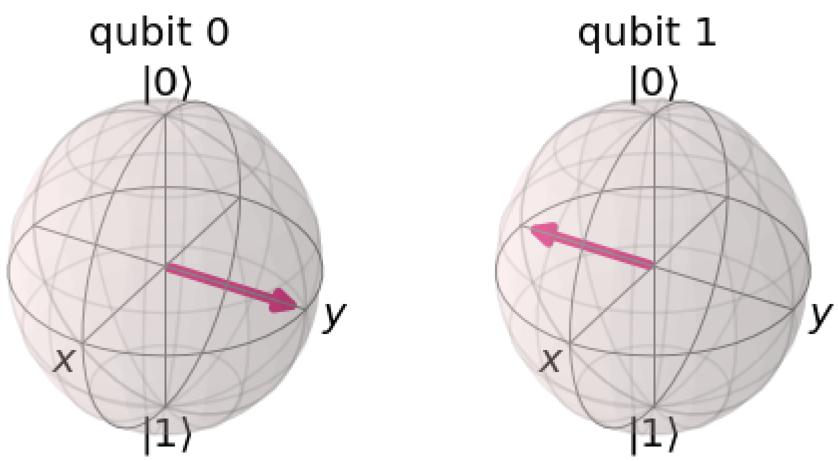
Out[9]:



We can see the results are as expected: qubit q_0 is always measured as $|0\rangle$, and q_1 is always measured in $|1\rangle$. As final confirmation, we check the bloch spheres for both these qubits.

In [10]:
```
job = execute(qc,backend_s).result()
state_vec = job.get_statevector()
plot_bloch_multivector(state_vec)
```

Out[10]:



6.1 The I gate

Quick Exercise

1. What are the eigenstates of the I-gate?

All vectors in the 2 dimensional Hilbert Space are the eigenstates of I-gate, since the eigen values of I-gate is just 1.

In [11]:
```
import qiskit
qiskit.__qiskit_version__
```

Out[11]:
```
{'qiskit-terra': '0.16.1',
 'qiskit-aer': '0.7.1',
 'qiskit-ignis': '0.5.1',
 'qiskit-ibmq-provider': '0.11.1',
 'qiskit-aqua': '0.8.1',
 'qiskit': '0.23.1'}
```

In [ ]:
```
```