어서 와!
양자 컴퓨팅은
처음이지?

04. 양자 게이트 다루기:
중첩(H)과 얽힘(CX)

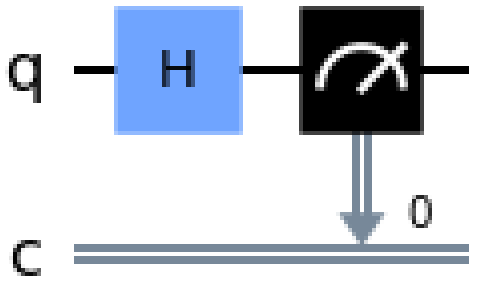
04. 양자 게이트들: 중첩과 얽힘

- **단일 큐비트 게이트**: Single Qubit Gates
  - **파울리 게이트**: Pauli-**X**, Y, Z
  - **하다마르 게이트**: Hadamard (**H**)
  - **위상 게이트들**: **U**, I, S, T, R, …

04. 양자 게이트들: 중첩과 얽힘

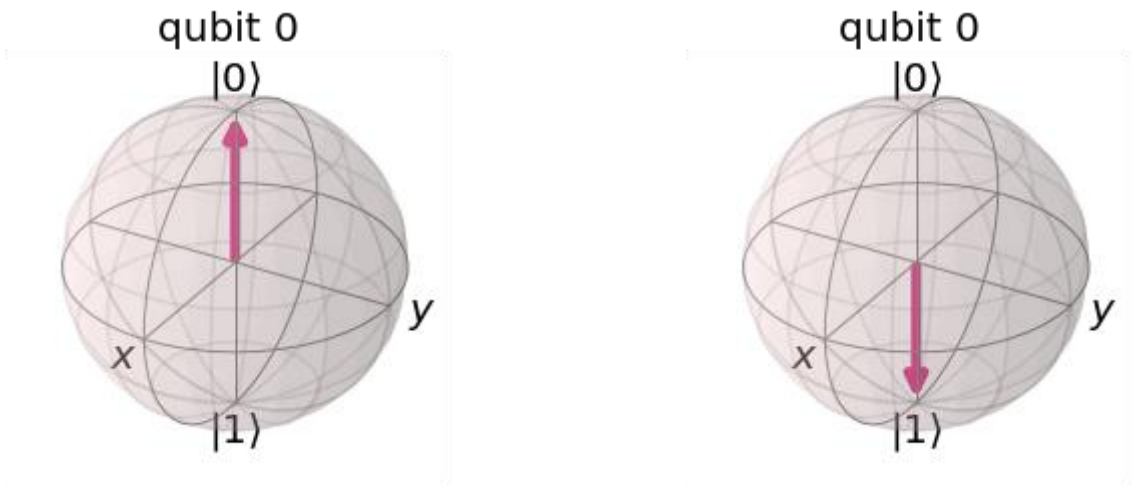
- **하다마르 게이트**: Hadamard (H) Gate
  - 단일 큐비트를 **중첩** 상태로 만드는 **유니터리 연산**을 하는 게이트.

```
from qiskit import QuantumCircuit, execute, Aer
from qiskit.visualization import plot_histogram
circuit = QuantumCircuit(1, 1)
circuit.h(0)
circuit.measure([0], [0])
circuit.draw()
```



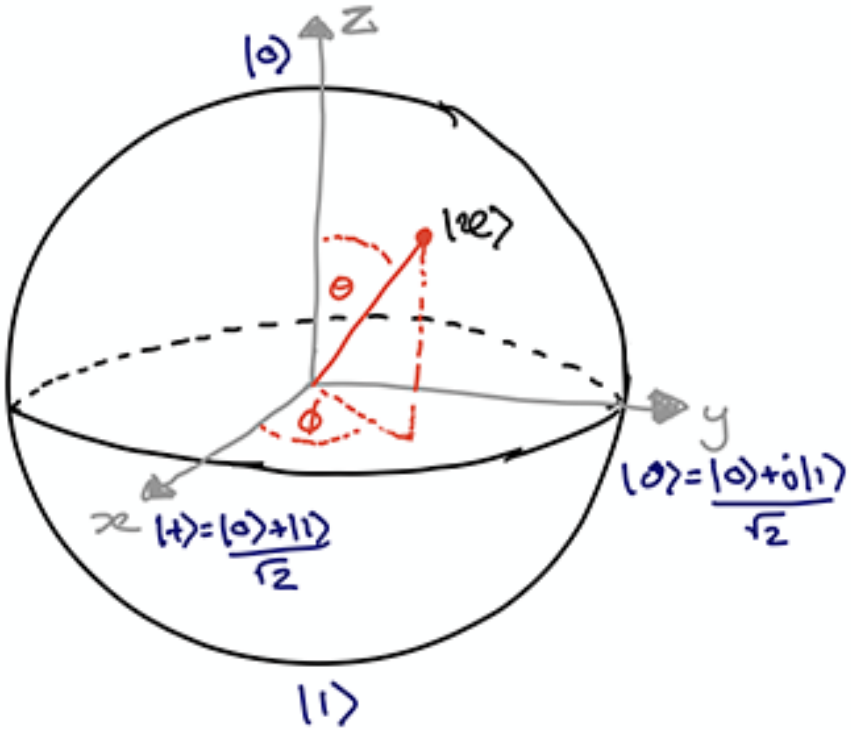
04. 양자 게이트들: 중첩과 얽힘

```
from qiskit.visualization import plot_bloch_multivector
backend = Aer.get_backend('statevector_simulator')
result = execute(circuit, backend).result()
states = result.get_statevector()
print(states)
plot_bloch_multivector(states)
```



- General ***U-Gate***
  - 단일 큐비트의 양자 상태를 마음대로 제어할 수 있는 (유일한) 물리 게이트
  - 나머지 게이트들은 U-게이트의 특별한 케이스라고 보면 됨.

```
from numpy import pi
circuit = QuantumCircuit(1, 1)
thet, phi, lamb = pi/4, pi/4, 0
circuit.u(thet, phi, lamb, 0)
circuit.measure([0], [0])
circuit.draw()
```



04. 양자 게이트들: 중첩과 얽힘

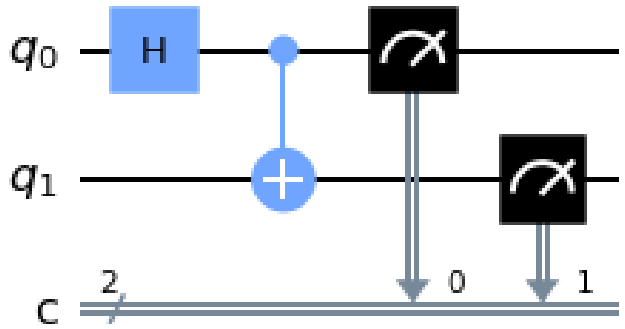
- **다중 큐비트 게이트**: Multiple Qubit Gates
  - SWAP-Gate: **두 큐비트의 상태를 서로 바꿈.**
  - CNOT-Gate: Controlled NOT (CX Gate)
  - **다른 큐비트 두 개짜리 게이트**: CRZ, CH, CU, CSWAP(**프레드킨 게이트**), …
  - **큐비트 세 개짜리 게이트**: CCX (**토폴리 게이트**)

04. 양자 게이트들: 중첩과 얽힘

■ 제어된-NOT 게이트: Controlled-NOT (CX) Gate

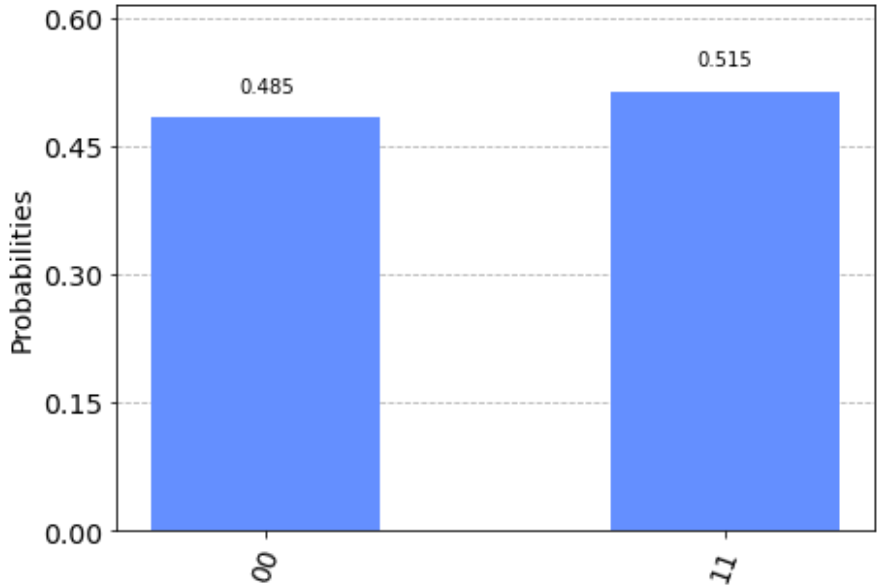
- 두 개의 큐비트를 얽힘 상태로 만드는 다중 게이트

```
from qiskit import QuantumCircuit, execute, Aer
from qiskit.visualization import plot_histogram
circuit = QuantumCircuit(2, 2)
circuit.h(0)
circuit.cx(0, 1)
circuit.measure([0, 1], [0, 1])
circuit.draw()
```



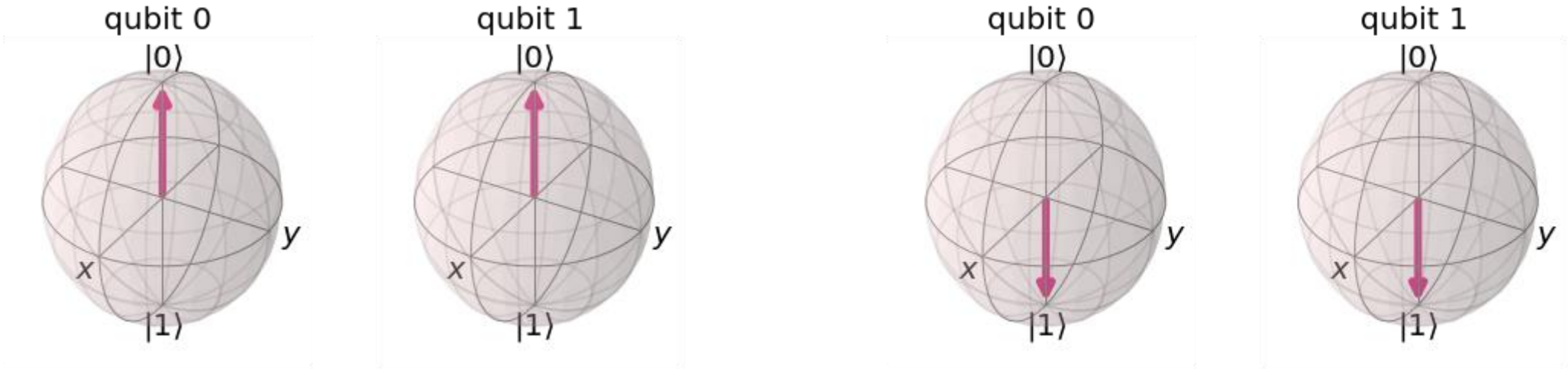
04. 양자 게이트들: 중첩과 얽힘

```
backend = Aer.get_backend('qasm_simulator')
result = execute(circuit, backend).result()
counts = result.get_counts()
print(counts)
plot_histogram(counts)
```



04. 양자 게이트들: 중첩과 얽힘

```
from qiskit.visualization import plot_bloch_multivector
backend = Aer.get_backend('statevector_simulator')
result = execute(circuit, backend).result()
states = result.get_statevector()
print(states)
plot_bloch_multivector(states)
```



Any Questions?

주니온TV@Youtube

자세히 보면 유익한 코딩 채널