In [29]:
```python
from qiskit import QuantumCircuit, Aer, assemble, execute
from math import pi, sqrt
from qiskit.visualization import plot_bloch_multivector, plot_histogram
import numpy as np
from qiskit_textbook.tools import array_to_latex
```
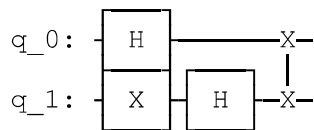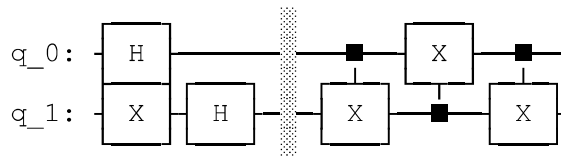
In [30]:
#2.4(1) Find a different circuit that swaps qubits in the states |+⟩ and |−⟩, and show that this is equivalent to the circuit shown above.

In [31]:
```python
circ = QuantumCircuit(2)
circ.x(1)
circ.h(0)
circ.h(1)
circ.swap(0,1)
display(circ.draw())
sim = Aer.get_backend('unitary_simulator')
job = execute(circ,sim).result()
sj = job.get_unitary()
array_to_latex(sj)
```

```
q_0: ┤ H ├──────X─
     ├───┤┌───┐ │
q_1: ┤ X ├┤ H ├─X─
     └───┘└───┘
```

$$\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \end{bmatrix}$$

In [32]:
```python
circ = QuantumCircuit(2)
circ.x(1)
circ.h(0)
circ.h(1)
circ.barrier()
circ.cx(0,1)
circ.cx(1,0)
circ.cx(0,1)
display(circ.draw())
sim = Aer.get_backend('unitary_simulator')
job = execute(circ,sim).result()
sj = job.get_unitary()
array_to_latex(sj)
```


$$\begin{bmatrix} \frac{1}{2} & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \end{bmatrix}$$

In [ ]: