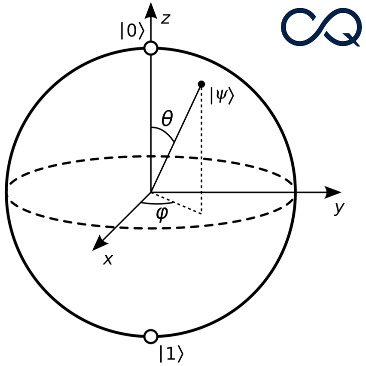
Notebook 1 : Cheat Sheet



1. **Qubits** can be in a **superposition state**
   - $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ with $|\alpha|^2 + |\beta|^2 = 1$
   - Represented by a vector in a complex vector space

2. **Dirac** notation
   - A **ket** $|\psi\rangle$ is a column vector and its associated **bra** $\langle\psi|$ is a row vector obtained by taking the complex conjugate transpose.

3. **Bloch sphere**
   - **Visual representation of a qubit** as a vector on a sphere with θ and ϕ as coordinates for the state : $|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle$

4. Quantum **gates** are **unitary matrices** and act on qubits like **rotations** on the Bloch sphere.

5. A **quantum circuit** is a sequence of unitary transformations (gates) applied to an initial state.
   - $|\psi_{\text{final}}\rangle = U_n \cdots U_2 U_1 |\psi_{\text{initial}}\rangle$
   - **Measurement** gives **probabilistic** results based on the final state of the qubit

Notebook 2 : Cheat Sheet

1. **Quantum circuits are composed of gates**

Gate	Circuit Element	Matrix Representation	Action on Basis States
Hadamard Gate H	$-\boxed{H}-$	$\frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\ 1 & -1\end{pmatrix}$	$H\|0\rangle = \frac{1}{\sqrt{2}}(\|0\rangle + \|1\rangle)$ $H\|1\rangle = \frac{1}{\sqrt{2}}(\|0\rangle - \|1\rangle)$
Pauli-X Gate X	$-\boxed{X}-$	$\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}$	$X\|0\rangle = \|1\rangle$ $X\|1\rangle = \|0\rangle$
Pauli-Y Gate Y	$-\boxed{Y}-$	$\begin{pmatrix}0 & -i\\ i & 0\end{pmatrix}$	$Y\|0\rangle = i\|1\rangle$ $Y\|1\rangle = -i\|0\rangle$
Pauli-Z Gate Z	$-\boxed{Z}-$	$\begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}$	$Z\|0\rangle = \|0\rangle$ $Z\|1\rangle = -\|1\rangle$
CNOT Gate	$\bullet$ $\oplus$	$\begin{pmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 0 & 1\\ 0 & 0 & 1 & 0\end{pmatrix}$	$CNOT\|00\rangle = \|00\rangle$ $CNOT\|01\rangle = \|01\rangle$ $CNOT\|10\rangle = \|11\rangle$ $CNOT\|11\rangle = \|10\rangle$

2. **Quantum circuits must return a measurement**
   (`qml.state`, `qml.expval`, `qml.probs`, `qml.counts`)

3. **Link a circuit and a device together with a Qnode**

Qnode

Quantum function

```
def circuit():
  …
 return qml.counts(0)
```

Device

```
qml.device(...)
```