Week 1 Plan: Qubits & Measurement Curriculum Development - Tier 2

1. Notebook Metadata & Setup		
Deliverable	**Target Audience: Grade 9-11**	
Notebook Framework	Tier 2: State Vectors and Your First Qiskit Circuit	Prerequisites: Basic programming, introductory Algebra (vectors, magnitude).

Prerequisite Scaffolding (Recap)
Requirement: A short section (2-3 markdown cells) that quickly reviews the conceptual idea of superposition and measurement randomness (the Tier 1 concepts), preparing them to move to math/code.

Concept	Objectives	Key Qiskit Tool
Qubit Representation	Introduce the qubit state as a vector in a 2-dimensional complex space. Show the standard notation and explain what the components represent.	Define the Qubit State Vector (using standard bracket notation): **State
Qiskit Activity	Guide students on how to construct a simple Qiskit circuit and apply the Hadamard gate (H), showing the resulting state vector- creates equal probability.	Focus on: Vectors, Magnitudes, Basic Algebra (solving for a or b).

2. Lab 1: Getting Started with the Qubit (Hands-On)		
Section	**Instruction in Notebook**	**Expected Student Action (using Python)**
Circuit Setup	Step 1: Import the necessary Qiskit tools and define a simple circuit with one quantum bit (1) and one classical bit (1).	Students write: from qiskit import QuantumCircuit, Aer and then qc = QuantumCircuit(1, 1)
Classical Bit (0/1)	Step 2: Use the X-gate method (qc.x()) to flip the state from the starting state **	0⟩** to the definitive **
Measurement & Runs	Step 3: Run the circuit on the Aer simulator for 100 shots and retrieve the results. This tests our classical-like operation.	Students write the code to execute the circuit and display the histogram.
Question 1 (Worksheet)	Question: Record the histogram result. Why is the outcome guaranteed to be 100% 1, even after running it 100 times? (Connect to classical logic)	Students record their observation in a markdown cell and reference the classical nature of the X-gate operation.

3. Noise: The "Glitch"		
Section	**Instruction in Notebook**	**Expected Student Action (Conceptual/Analysis)**
Noise Focus	Introduce the idea of "Energy Leakage" and "Decay" (specifically the T1 process). Explain that quantum computers are imperfect and errors can creep in.	Students read the explanation in a dedicated markdown cell.
Scientific Alignment	Explain that superconducting qubits (like the Transmon) work best near absolute zero (0 Kelvin). Relate leakage to the Kelvin Scale from their chemistry/physics class—if the device gets too "hot," the qubit state can spontaneously flip from **	1⟩** back to **
Activity/Challenge	Question: Imagine a quantum computer is running in a room that is slightly too warm. Which state is the qubit most likely to "leak" into: **	0⟩** or **

5. Summary and Reflection
The Qubit State: The quantum state is represented by a state vector, $\|\psi\rangle = (a, b)$, which lives in a multi-dimensional space.
Superposition: The Hadamard (H) gate puts the qubit into a state where it is simultaneously both $\|0\rangle$ and $\|1\rangle$.
Measurement is Probabilistic: Measurement forces the qubit to collapse to a definite state, and the probability of getting $\|0\rangle$ or $\|1\rangle$ is determined by the squared magnitude of the vector components: $\|a\|^2 + \|b\|^2 = 1$.
Qiskit is the Language: We can build and test these concepts using the Qiskit SDK by writing simple Python code.
Noise is Energy: Errors are not just glitches; they are often caused by Energy Leakage (like T1 decay) that we must understand by referencing the Kelvin Scale and the physics of the qubit hardware.

Final Reflection Questions:
The final questions should encourage the student to bridge the code, math, and science alignment established in the notebook:
Code to Math: If you ran your simple circuit 1000 times, you probably got a histogram that wasn't exactly 500 for $\|0\rangle$ and 500 for $\|1\rangle$. Why does running the code 1000 times not guarantee a perfect 50/50 split, even though the math says it should?
Science Connection: Based on the Energy Leakage concept, if we wanted to measure a qubit's state very accurately, should the quantum computer's refrigerator be warmer or colder? Explain your reasoning using the concept of energy states.
Procedural Challenge: What single line of Qiskit Python code is responsible for turning the abstract quantum state into a measurable classical result?