**Sphero Programming Unit - Year 6 Overview**

* Systematic 6-lesson programming unit using Sphero robots for hands-on physical computing experiences
* Employs Core + Extension model where all students master essential concepts before accessing differentiated challenges

**Unit Structure**

**Lesson 1: Sequential Programming**

* Students learn programming sequences and basic input/output using movement blocks
* Core objective: Create square pattern using 8 movement commands
* Establishes step-by-step programming logic understanding
* Extensions: LED corner lighting and geometric shapes with varying angles

**Lesson 2: Loops & Efficiency**

* Focus on code optimization through repetition structures
* Students rebuild square using efficient loop commands and create spiral patterns
* Learn how loops reduce code complexity
* Extensions: Mathematical patterns, star designs, animated movement sequences

**Lesson 3: Advanced Events**

* Master event-driven programming by creating responsive behaviors
* Core lesson: Program reactions to physical events (collisions, landings, free falls)
* Teach how programs respond to external stimuli
* Extensions: Spin detection, interactive art installations, system state awareness

**Lesson 4: Conditional Logic**

* Introduction to decision-making through if/else statements and boolean logic
* Create motion detectors that change LED colors based on movement (shake = red, still = green)
* Demonstrate intelligent responses to sensor data
* Extensions: Sound feedback, multi-level detection systems, motion-based games

**Lesson 5: Variables & Data**

* Learn data storage and manipulation through variable systems
* Core objective: Create scoring systems with event-triggered counters
* Teach how programs remember and modify information
* Extensions: Timer systems, multiple variable tracking, smart scoring with conditional logic

**Lesson 6: Integration & Games**

* Culminating lesson combining all previous concepts into complete game systems
* Create games integrating events, conditionals, loops, and variables
* Demonstrate mastery of multiple programming concepts working together
* Extensions: Multi-level games, competitive systems, adaptive gameplay