LESSON 9

Plan and build your first arcade game, Part 1

40 minutes | Strand 1: Introducing computer science | Built in MakeCode Arcade | Outcome 1.5 | Outcome 1.6 | Outcome 1.7

WHAT THIS LESSON DOES

Students plan a simple arcade game, then use MakeCode Arcade to create a player sprite, add movement, a tile map with walls, projectiles and lives, following a PRIMM code-along and saving their work.

CURRICULUM OUTCOMES

Outcome	What the specification asks for
1.5	develop appropriate algorithms using pseudo-code and/or flow charts
1.6	construct code to implement algorithms
1.7	discuss and implement core features of structured programming languages, including variables, operators, loops, decisions, assignment and modules

WHAT STUDENTS SHOULD BE ABLE TO DO

- Plan a small game with pseudo-code or a flow chart (LO 1.5)
- Set up a sprite and player movement using events and sequence (LO 1.6, LO 1.7)

BEFORE THE BELL

- Ensure every device is logged in and can open arcade.makecode.com in the browser
- Open MakeCode Arcade on the board ready to model the first steps
- Have scrap paper or whiteboards ready for a quick plan (pseudo-code or a simple flow chart) before coding
- Optional: check any Arcade handhelds (BrainPad, GameGo) if you plan to demo download at the end

WHAT YOU NEED

Item	Qty	Per	If you do not have it
tablet or laptop with MakeCode Arcade	1	pair	run the lesson on the IWB with students taking turns

Plan and build your first arcade game, Part 1

HOW THE LESSON RUNS

4 min	introduction	Start: what we're building
2 min	content	What is MakeCode Arcade?
2 min	content	The Code Editor
2 min	content	Create a new Arcade project
3 min	content	Add a sprite
2 min	content	Choose a sprite from the gallery
2 min	content	Move your sprite
2 min	content	Draw the tile map
2 min	content	Draw the walls
2 min	content	Camera follow sprite
2 min	content	Add some projectiles
2 min	content	Set their direction and speed
2 min	content	Detect overlap
3 min	content	Lose a life
2 min	content	Send the code to your handheld
3 min	content	Make sense: what worked, what broke
2 min	content	Reflect
1 min	summary	What you covered

TEACHING MOVES

- Model the plan on the board first: player moves, walls block, projectiles cost a life. Keep it to a few lines of pseudo-code or three boxes in a flow chart
- Build the first sprite and move block on the board before pairs continue so everyone sees where blocks live in the toolbox
- Pause after movement works and after walls work so students run the game and look at what it does before adding more
- Circulate during projectiles and overlap: those steps stack several ideas and are where most bugs appear

WHAT TO WATCH FOR

- Sprite created but move block missing or not attached: player stays still. Check that the move block targets the same sprite variable
- Tile map drawn but walls not painted with the Draw Walls tool: player walks through the border. Reopen the map editor and mark the edge tiles as walls
- Overlap runs effects but lives never drop, or projectiles are not destroyed: the change-life and destroy blocks were skipped or sit outside the overlap handler. Add them inside the overlap event and run it again

DIFFERENTIATION

- Support: Provide a three-line plan template (create player, move with arrows, lose life on hit) and stay with the gallery sprite rather than custom art
- Support: Let pairs complete movement and a bordered map first; add projectiles only when that runs cleanly
- Extension: Adjust projectile speed or interval, redesign the map layout, or note one extra feature to try in a later part of the project