

Studio: Galaxy Ghosts

40 minutes

Project studios (optional extension sessions)

Built in MakeCode Arcade

Outcome 1.7

Outcome 1.8

WHAT THIS LESSON DOES

Students build Galaxy Ghosts in MakeCode Arcade step by step, adding a player, falling enemies with a speed variable, projectiles, scoring, a status bar and damage, then rebalance one value so the game feels hard but fair.

CURRICULUM OUTCOMES

Outcome	What the specification asks for
1.7	discuss and implement core features of structured programming languages, including variables, operators, loops, decisions, assignment and modules
1.8	test the code to determine its functionality and to identify and manage errors

WHAT STUDENTS SHOULD BE ABLE TO DO

- Build a shooter with a speed variable, a status bar and damage (LO 1.7)
- Tune the difficulty so it is hard but fair (LO 1.8)

BEFORE THE BELL

- Students logged in at devices with MakeCode Arcade ready (arcade.makecode.com).
- Check that students know how the class saves and reopens Arcade projects.

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

Studio: Galaxy Ghosts

HOW THE LESSON RUNS

4 min	introduction	Start: what we're building
2 min	content	The plan: six parts to build
2 min	content	Start a New Project
2 min	content	Create your player
3 min	content	Move left and right
2 min	content	Create the targets
3 min	content	Random position at the top
3 min	content	Create a speed variable
3 min	content	Fire!
2 min	content	Hit the target
2 min	content	Add the 'status bar' extension
2 min	content	Take damage
2 min	content	Game over
2 min	content	Wrap Up
3 min	content	Make sense: what worked, what broke
2 min	content	Reflect
1 min	summary	What you covered

TEACHING MOVES

- Model the first two build steps on the board from your own new project, then let pairs work at devices while you circulate.
- When the speed variable appears, pause and ask what happens to difficulty if the starting value is higher or lower.
- Stress kind selections on overlap blocks (Projectile/Enemy, Enemy/Player) before students test.
- Leave time for the independent practice: change one number (speed, damage or spawn interval) and defend the choice in the make-sense or reflect talk.

WHAT TO WATCH FOR

- Wrong sprite kind on an overlap block so hits never register: reopen the block and set Projectile with Enemy, or Enemy with Player.
- Destroying the wrong sprite in an overlap handler so the player or projectile vanishes: point students at the parameter name in the handler and swap it.
- Status Bar extension missing so health blocks are unavailable: add the extension, then rebuild the status bar under the player.

DIFFERENTIATION

- Support: Keep the board project one step ahead and let students copy structure from it when stuck.
- Support: Offer a small set of safe values for speed, damage and spawn interval to try first.
- Extension: After one solid rebalance, tune a second value and explain how the two interact.