

LESSON 12

Game showcase: play, evaluate, improve

40 minutes

Strand 1: Introducing computer science

Outcome 1.9

Outcome 1.8

WHAT THIS LESSON DOES

You will play each other's games, give specific feedback by naming one strength and one improvement, and then make and test one improvement to your own game based on the feedback received.

CURRICULUM OUTCOMES

Outcome	What the specification asks for
1.9	Evaluate a program in groups of two or three.
1.8	Test code to find and manage errors.

WHAT STUDENTS SHOULD BE ABLE TO DO

- Evaluate each other's games in groups of two or three, naming one strength and one improvement (LO 1.9)
- Make one improvement to one of your own games based on the feedback, and test it (LO 1.8)

BEFORE THE BELL

- Prepare notebooks or paper for students to record feedback.
- Have a class sample project or two-minute starter template ready for anyone whose saved game is missing.
- Confirm students can open saved Scratch and MakeCode Arcade projects the class way; have a class sample or starter template ready for anyone whose project is missing
- Have one sample game ready on the board to model feedback; prepare a class sample or starter template for missing projects
- Student devices with saved Scratch and MakeCode Arcade projects open
- Student devices; class save method ready to name aloud in the final two minutes

Game showcase: play, evaluate, improve

HOW THE LESSON RUNS

4 min	introduction	Start
5 min	content	How good feedback works
1 min	content	Open a game ready to share
11 min	content	Play and evaluate in your group
2 min	content	Check you have your feedback written down
8 min	activity	Make one improvement
3 min	content	What done looks like
3 min	content	Make sense
2 min	content	Reflect
1 min	summary	What you covered

TEACHING MOVES

- Model clear examples of strength and improvement feedback using a sample game before groups begin.
- Circulate during the hands-on activity to encourage specific comments from students.
- Keep play-passes to about 90 seconds; if time runs short in a group of three, ensure every maker still has at least one strength and one improvement written before moving on.
- Emphasise keeping the improvement small and testable within the time limit.
- Before students run an edited game, ask what they expect to see, then let them check against that prediction.
- In the final two minutes of the improve step, stop the room and name the exact save method aloud so every student saves before you pull them back.
- Use whole-class discussion in the Make sense step to share examples of useful feedback.
- In Reflect, run a 30-second pair share before taking voluntary whole-class answers.

WHAT TO WATCH FOR

- Giving vague feedback like 'nice game' - prompt for specifics such as controls or goal.
- Making multiple changes instead of one - remind to pick only one from the feedback.
- Not testing after editing - instruct to run the game right after making the change.

DIFFERENTIATION

- Support: Offer sentence starters for feedback such as 'One thing that works well is...'.
 - Support: Suggest pairing students strategically for the group activity.
 - Extension: Challenge students to explain why their chosen improvement will improve the game.