

LESSON 18

Dino Jump

60 minutes

Game Design Studio

WHAT THIS LESSON DOES

Get ready to create an exciting game with MakeCode Arcade! In this lesson, you'll build a version of the classic Google Chrome game, learning to design a map, craft a dino character, add jumps, obstacles, and scoring.

WHAT STUDENTS SHOULD BE ABLE TO DO

- Develop skills in creating an interactive game using MakeCode Arcade.
- Understand how to draw a map, create a character, and add movement and gravity effects.
- Learn to add obstacles and implement collision detection for game over scenarios.
- Gain knowledge on how to keep score and determine winning conditions in a game.
- Reflect on the game design process and the elements involved in creating an engaging game.

BEFORE THE LESSON

- Open arcade.makecode.com yourself first and confirm the school filter is not blocking it. This is the whole lesson.
- One device per pupil, each able to open a MakeCode Arcade project. Pairing is fine if devices are short.
- Sound on, or headphones ready: the jump plays a sound and some pupils rely on it to confirm their code ran.
- Have your own working Dino Jump open to show what finished looks like and to check code against when a pupil is stuck.

Dino Jump

HOW THE LESSON RUNS

Introduction
Create a new arcade project
Draw the map
Create the dino sprite
Gravity and moving the dino
Make it jump
Add the cactuses
Detect hitting a tree
Keep score
Win the game
Play and Review

TEACHING MOVES

- 1. Introduction. Show the classic Chrome Dino game briefly so pupils know the target. Say they will build every piece of it: map, character, jump, obstacles, score and win.
- 3. Draw the map. Stress the map is 100 wide and 8 high. Have them draw ground all along the bottom, a finish tile at the far end, then use the draw walls tool on the ground so the dino lands on it.
- 4. Create the dino sprite. Point out the design must have no gap at the bottom: the legs must touch the ground or gravity and jump detection will misbehave later.
- 5. Gravity and moving the dino. Have them press play and watch the dino fall and move forward. If it drops through the floor, send them back to check the ground tiles are set as walls.
- 6. Make it jump. Explain the if-then only lets the dino jump when it is touching the ground, so pupils cannot double-jump in mid-air. Test the A button and the space bar.
- 7. Add the cactuses. Let pupils place trees where they want but warn them to leave jumpable gaps. They will reopen the map editor a few times to space them fairly.
- 8. Detect hitting a tree. The commonest slip is not selecting the tree tile in the overlaps block. Have each pupil hit a tree on purpose and confirm the game over screen shows.
- 9. Keep score. Check the set score to 0 block sits inside on start, not elsewhere. Ask how they might award points for surviving longer.
- 10. Win the game. Make sure the overlaps block here targets the finish tile from step 3, not the tree tile. Have them run to the end and see the You Win screen.
- 11. Play and Review. As pupils play, ask them to name one element they added and what it does. Draw out that map, character, movement, obstacles, score and win are all parts of game design.

WHAT TO WATCH FOR

- Pupils forget to make the ground tiles walls, so the dino falls straight through the map when gravity is added. Send them back to the map editor and the draw walls tool. Nothing below the dino counts as ground until it is a wall.
- The overlaps blocks for hitting a tree and reaching the finish look identical, so pupils leave the wrong tile selected and get the wrong game over or win. Have them click the tile in each overlaps block and read it back to you: tree means lose, finish means win.
- Pupils expect a smooth double-jump and think their code is broken when the dino only jumps from the ground. Explain the if-then checks the dino is touching the ground first. This is deliberate, not a bug.

DIFFERENTIATION

- Emerging: Pair them with a partner and get the project created and the map drawn together before splitting to their own screens. Give them your open example to copy block placement from when the code will not run.
- Developing: Let them build each block and press play after every step, so a mistake shows up at the step that caused it rather than at the end.
- Proficient: Ask them to extend the map, add more trees, or change the jump velocity and tell you what it does to the difficulty. Have them explain their score idea back to you and try awarding more points the longer the dino survives.