LESSON 10

Prison Break

60 minutes | Advanced Game Development

WHAT THIS LESSON DOES

Get ready to create an exciting maze game with MakeCode Arcade! You'll design a character, build a challenging maze, set a goal to reach, and add a timer to ramp up the difficulty. Let's begin!

WHAT STUDENTS SHOULD BE ABLE TO DO

- Develop skills in game creation using MakeCode Arcade.
- Understand and apply coding concepts to create and control a character sprite.
- Design and implement a maze using tilemap and wall settings.
- Integrate game elements such as a goal tile and timer to enhance gameplay.
- Apply problem-solving skills to navigate through the maze within a set time limit.

BEFORE THE LESSON

- Open arcade.makecode.com yourself and confirm the school filter allows it and that projects save. Blocked access will stop the whole lesson.
- One device per pupil, each able to reach the site and create a new project.
- Have your own finished Prison Break game saved and open so you can model each step and show what the end product looks like.

Prison Break

HOW THE LESSON RUNS

Introduction to Prison Break Game
Create a new Arcade project
Create a sprite
Move with the buttons
Draw your maze
Set the walls of your map
Camera follow the sprite
Add a goal tile
Win the game
Set a timer
Wrap up

TEACHING MOVES

- 1. Introduction to Prison Break Game. Play your own finished game before pupils start coding. Show the character moving, hitting walls, reaching the goal and the timer running out, so they know exactly what they are building towards.
- 2. Create a new Arcade project. Walk the whole class to a fresh project together before anyone codes. Confirm every pupil has a blank workspace open so nobody starts a step behind.
- 3. Create a sprite. Show the sprite editor on the board and draw your character live. Pupils will want their own design, so let them, but check the create block is placed before they get lost in colouring.
- 4. Move with the buttons. Test the controls straight after adding this block. Have pupils press each arrow and confirm the sprite responds before moving on, so control problems are caught now not later.
- 5. Draw your maze. Remind pupils to design a maze that is actually solvable. Ask them to trace a path from start to goal with a finger before they set walls.
- 6. Set the walls of your map. Point out the Draw Walls tool is the 3D wall icon, not the tile picker. Have pupils test by walking the sprite into a wall to prove it blocks them.
- 7. Camera follow the sprite. Ask pupils who noticed their sprite vanishing off screen. Add this block, then have them walk to the edge again to see the camera track it.
- 8. Add a goal tile. Stress that the goal must be a different tile from the walls and placed somewhere reachable. One goal tile only.
- 9. Win the game. The overlap block must be set to the exact tile chosen as the goal. This is where most will get stuck. Check each pupil selected the right tile in the block.
- 10. Set a timer. Test the finished game end to end: play, race the maze, and confirm both the win screen and the 30 second timeout work.

WHAT TO WATCH FOR

- Pupils think placing a wall tile in the tilemap is enough to block the player, and are confused when the sprite walks straight through it. Show that a tile is only a picture until the Draw Walls tool marks it as solid. Have them use the wall tool on each barrier tile, then test by walking into it.
- The win screen never appears because the overlap block is set to a different tile than the one actually placed as the goal. Sit beside the pupil and compare the tile in the onOverlapTile block with the tile in their maze. Reset the block to the matching goal tile and retest.
- Pupils build a maze with no possible route to the goal, then think their code is broken. Have them trace a finger path from start to goal. If there is no clear path, remove or unset a wall to open one before checking the code.

DIFFERENTIATION

- Emerging: Give a small, simple maze target: a short corridor with one turn, so getting a working game matters more than a hard maze. Pair them with a partner who has the sprite moving, and have that partner talk them through the button block.
- Developing: Ask them to explain back what the walls are doing and why the sprite stops, to check they understand rather than copied.
- Proficient: Challenge them to shorten the timer, add a second goal, or design a harder maze with dead ends once the basic game works. Have them help a pupil nearby by explaining the overlap block, which deepens their own understanding.