LESSON 5

Red v Green v Blue

60 minutes | Coding and Interactive Creators

WHAT THIS LESSON DOES

Follow this guide to build an engaging project in Scratch. Start by creating a new project, design red, green, and blue dot sprites, and code them to move, clone, and interact by changing colours through infection.

WHAT STUDENTS SHOULD BE ABLE TO DO

- Develop skills in creating and managing a new Scratch project.
- Acquire knowledge in creating and modifying sprites and costumes in Scratch.
- Understand and apply the concept of variables in Scratch programming.
- Learn to code sprite clones and manage their behaviour in Scratch.
- Enhance problem-solving skills by improving and customising the project.

BEFORE THE LESSON

- One device per pupil, logged in to Scratch and able to reach scratch.mit.edu. Check the site is not blocked by the school filter before the bell.
- Have the fallback starter project open and its link (scratch.mit.edu/projects/780048995) ready to share for any pupil who cannot draw the dot sprite.
- Try the full build yourself once beforehand so you can spot where a pupil's clone code has gone wrong.

Red v Green v Blue

HOW THE LESSON RUNS

Create a new Scratch project
Create a red dot sprite
Create the green and blue costumes
Create a 'count' variable
Create 100 clones of the dot
Appear randomly
Make them move
Make them infect each other!
Improve the project!

TEACHING MOVES

- 1. Create a new Scratch project. Show them where the paintbrush and the cat's delete button are. Getting rid of the cat sprite trips up first-timers who expect to keep it.
- 2. Create a red dot sprite. Demonstrate holding shift while dragging the circle tool to get a true circle, and turning the outline off. Name the costume 'red' out loud so they copy the exact name.
- 3. Create the green and blue costumes. Model duplicating the red costume rather than redrawing, then recolour with the paint bucket. Pair a confident pupil with anyone struggling in the editor.
- 4. Create a 'count' variable. Explain a variable as a box that holds a number the program can change. Show them naming it 'count' and setting it to 0 under the green flag.
- 5. Create 100 clones of the dot. Walk through the repeat-until logic aloud: clone, add 1 to count, stop when count passes 99. Point out that hiding the original is what leaves only clones visible.
- 6. Appear randomly. Stress that this code goes under 'when I start as a clone', not the green flag. Run the green flag together so they see 100 dots scatter and know it worked.
- 7. Make them move. Point out 'point in direction' uses a random 0 to 359, and 'if on edge, bounce' keeps dots on screen. This block goes below the show block.
- 8. Make them infect each other! Slow right down here. Draw the red-green-blue infection cycle on the board and show one nested if-then before they build the rest. This is where most errors happen.
- 9. Improve the project! Ask them aloud what they would change: speed, dot count, a score, a fourth colour. Let them try one idea rather than perfecting it.

WHAT TO WATCH FOR

- Pupils attach the clone code to the green flag instead of 'when I start as a clone', so nothing behaves as expected. Show the two starter hats side by side and name which blocks belong under each. Have them point to the right hat before adding code.
- Costume names are mistyped or capitalised differently, so the colour logic misfires later. Have them read their three costume names back to you before the coding steps: exactly 'red', 'green', 'blue', all lowercase.
- The nested if-then blocks in the infection step get placed in sequence rather than inside each other. Zoom in on the nesting together. Have pupils drag one block inside another and check the shape wraps around before continuing.

DIFFERENTIATION

- Emerging: Hand them the starter project link so they skip the sprite drawing and focus on the coding. Sit them beside a pupil who has the clones appearing, and have that pupil talk them through the clone code.
- Developing: Ask them to explain back to you what the count variable is doing before they move on to the movement step. Check their costume names and clone hat block are right before the infection step, where errors compound.
- Proficient: Send them to the improvement step early: add a score, change dot speed, or add a fourth colour to the infection cycle. Have them help a partner debug a clone that will not move or bounce.