

Sound effects

60 minutes

Dynamic Digital Projects

WHAT THIS LESSON DOES

This lesson guides students through the process of adding sound effects to a baseball game in Scratch. They will learn to create a new project, add and code sprites, detect hits, and incorporate sound effects for different game scenarios. The lesson concludes with a game over scenario, encouraging further exploration and creativity.

WHAT STUDENTS SHOULD BE ABLE TO DO

- Understand and apply the concept of sound effects in Scratch programming.
- Create and modify a Scratch project, incorporating different sprites and backdrops.
- Develop skills in coding for animation and sound effects using Scratch.
- Apply conditional statements in Scratch to detect interactions between sprites.
- Enhance creativity and problem-solving skills through the design and improvement of a game.

BEFORE THE LESSON

- Open Scratch on your own device first and check the school filter is not blocking it, then confirm pupil devices can reach it and are logged in.
- Check sound is on and working on pupil devices; headphones make testing sound effects far easier in a full room.
- Have a finished version open on your screen so pupils can hear the hit, cheer and gasp before they build their own.

Sound effects

HOW THE LESSON RUNS

Introduction to Sound Effects
Create a new Scratch project
Add the Baseball 2 backdrop
Add and code the Batter sprite
Add and code the Baseball sprite
Detect a hit
Add some sound effects
Add a sound effect if you miss
Game Over!

TEACHING MOVES

- 1. Introduction to Sound Effects. Play the target game once so pupils hear all three sounds. Ask them to name exactly when each one fires: on the hit, on the cheer, on the miss.
- 2. Create a new Scratch project. Show pupils where the delete option is for the cat sprite. A blank stage now avoids the cat interfering with the ball later.
- 4. Add and code the Batter sprite. Stress the wait blocks between costume switches. Without them the swing happens too fast to see. Have pupils test the space key before moving on.
- 5. Add and code the Baseball sprite. Point out that the ball passes straight through the bat here on purpose. It only reacts once the hit test is added next, so it is not broken yet.
- 6. Detect a hit. Show how to pick the exact bat colour with the eyedropper in the touching colour block. The wrong colour is the most common reason a hit is never detected.
- 7. Add some sound effects. Check pupils add both sounds from the sound library before coding. A start sound block naming a sound they have not added will silently do nothing.
- 8. Add a sound effect if you miss. Explain that x position greater than 220 means the ball has passed the batter off the right edge. Use the red-line picture to make that position concrete.

WHAT TO WATCH FOR

- Pupils think the sound will play automatically once the block is dropped in, but nothing happens because the sound was never added from the sound library first. Have them open the Sounds tab and confirm the named sound is listed before running. Sounds must be added, then chosen in the block.
- The hit is never detected because the touching colour was picked by eye rather than sampled from the bat. Have them use the eyedropper on the actual bat and compare against #ffc800. A slightly different shade will never match.
- Pupils confuse start sound (which plays and carries on) with play sound until done (which pauses the script). The gasp uses the second on purpose. Ask them why the miss sound waits but the hit sound does not. Let them swap the blocks and hear the difference.

DIFFERENTIATION

- Emerging: Give the batter and ball starting code on your screen for them to copy block by block, then have them focus only on adding the three sounds. Pair them with a pupil who has the ball moving already so they can see a working version before building their own.
- Developing: Have them explain back to you what the touching colour block is checking before they add the sounds. Prompt them to test after each sound is added rather than building all of it and then guessing what went wrong.
- Proficient: Challenge them to add a score counter that goes up on every hit. Ask them to add a further sound of their own choosing and explain exactly when in the code it should fire.