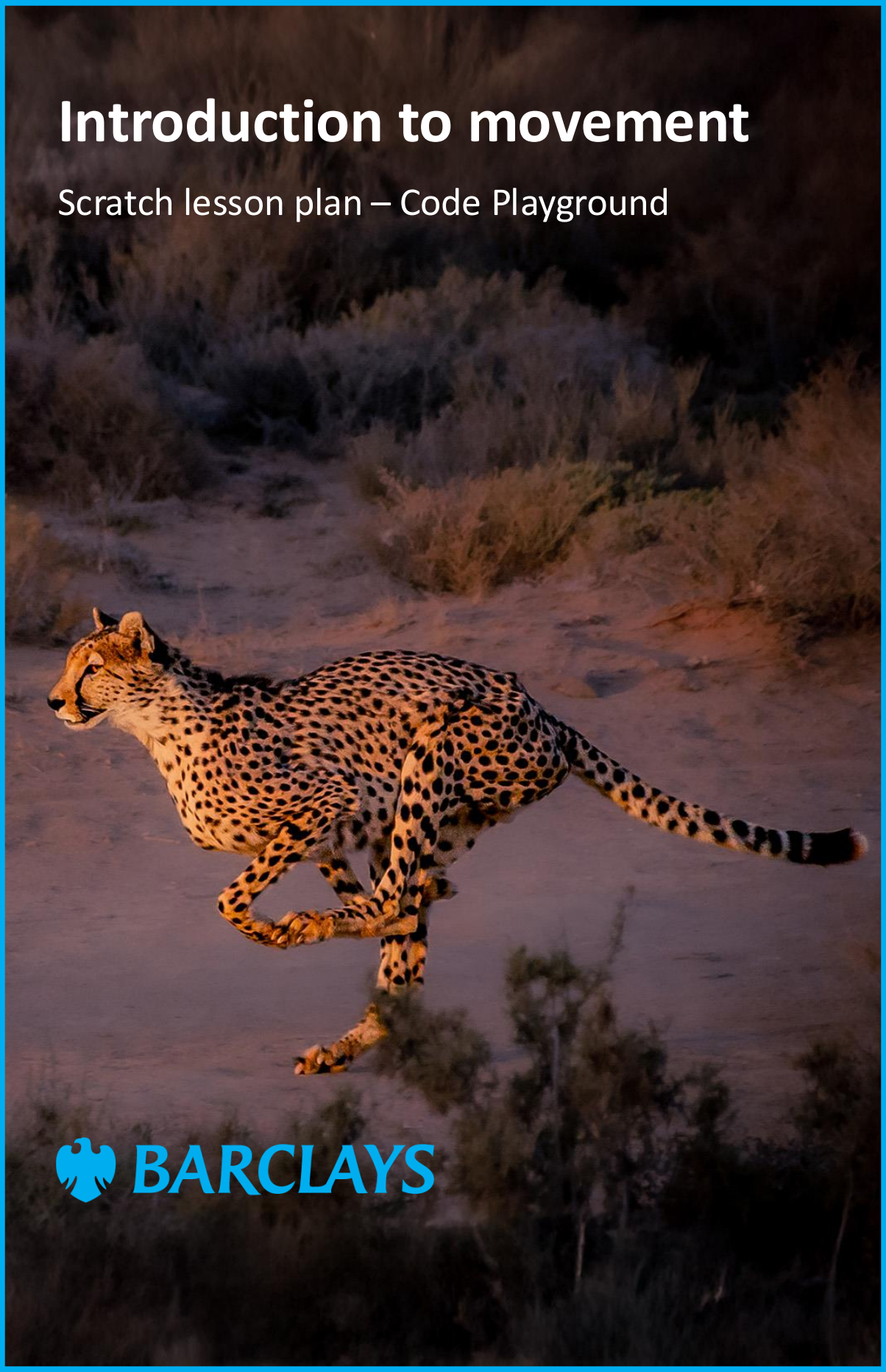
Introduction to movement

Scratch lesson plan – Code Playground



BARCLAYS

Lesson overview

This lesson introduces students to controlling movement in Scratch using X and Y coordinates and motion blocks. Students will learn to control sprite movement using keyboard keys, a mouse pointer or finger (depending on device), as navigation laying the foundation for future coding projects and games.

Time	Key learning outcomes	Resources
30 mins	• Recognise how X and Y axes control sprite positioning on the Scratch stage • Apply Scratch blocks to program keyboard, mouse and touch-based sprite movement • Observe, troubleshoot, and improve movement smoothness.	• Laptops or desktop with a mouse or trackpad, or tablets with touch input • Access to Scratch website - https://scratch.mit.edu or the Scratch app • Projector or screen for demonstration.

Content

Lesson introduction (5 minutes)

How do characters move in a game? Sometimes they follow key presses, your mouse, or on a touch screen, they follow your finger.

We'll make a sprite move with these.

X and Y coordinates

Demonstrate how moving a sprite changes its X and Y value.

- Use a laptop with a keyboard, a mouse or a tablet for comparison
- On a projector, show the sprite's X and Y values in the Scratch interface.

Teach students the phrase "X is a cross and Y reaches for the sky" to help them remember.

Open Scratch, add a sprite and demonstrate how changing X and Y values moves it on the stage.

Activity – Moving your sprite

Students will program their sprite to follow their key strokes, mouse movements or finger on a touch screen. They will explore how X and Y coordinates affect movement and use motion and control blocks to create smooth, continuous actions.

By the end of the activity, students will have a working program where their sprite follows their input, and they will understand how to adjust movement speed and direction.

Remember, the X-axis moves the sprite left and right, while the Y-axis moves it up and down. When you move your mouse or touch the screen, Scratch tracks those coordinates.

Instructions

- Ask students to choose a sprite from the library or create their own. Encourage creativity while keeping it simple (e.g. Scratch Cat, Ball)
- Use the workbook to show the blocks they'll use, or have them follow you to add the blocks to their sprite.

Ask your class

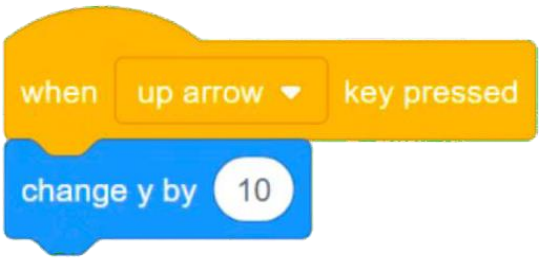
- What might happen if you increase or decrease the step size in the motion block?
- What other actions could you add to make the movement more interesting (e.g. sound or animation)?

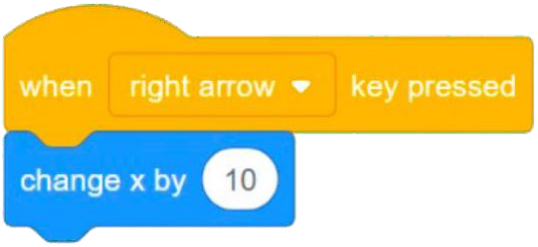
Troubleshooting

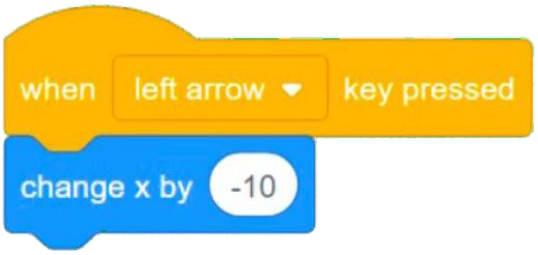
If students can't find the block, use the code snippet for colours and locations on the next page. If they experience any of the following:

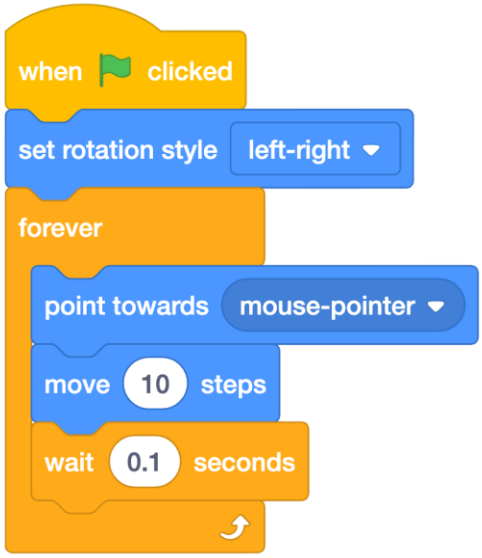
- Sprite not moving – make sure they've added the forever block around the motion blocks
- Movement is too fast or slow – Adjust the value in the move [10] steps block
- Touch input not working – restart the app or bowser to make sure it recognises the touch input.

Code snippet









Summary

The following information is an example of what a child at an expected level would be able to demonstrate when completing these activities with additional examples to demonstrate how this would vary for a child with emerging or exceeding achievements.

Assessment guidance

Differentiation – Lower ability/ASN

- Focus on a single motion block (e.g. 'move [10] steps' or 'point towards [mouse pointer]'
- Skip the 'forever' block initially and let students test the sprite's movement manually
- Provide screenshots or printed visuals showing where to find each block in Scratch
- Pair students with a buddy to assist with block selection and setup.

Differentiation – Higher ability/extension

- Introduce variables to control speed dynamically
- Add animations such as changing costumes, or sounds when the sprite moves
- Challenge the students to create a simple maze or race where the sprite must follow the pointer or avoid obstacles
- Suggest students explore other inputs, like keyboard events instead of mouse or touch input.

Plenary

- Ask students to share what worked well in your program? What did they find challenging?
- Show examples of students who explore advanced controls

Assessment questions

- What does the X coordinate control on the stage?
- Why do we use the forever block for continuous movement?
- How would make the sprite move faster?
- Can you think of games or apps where characters follow your input? How do you think they are programmed?