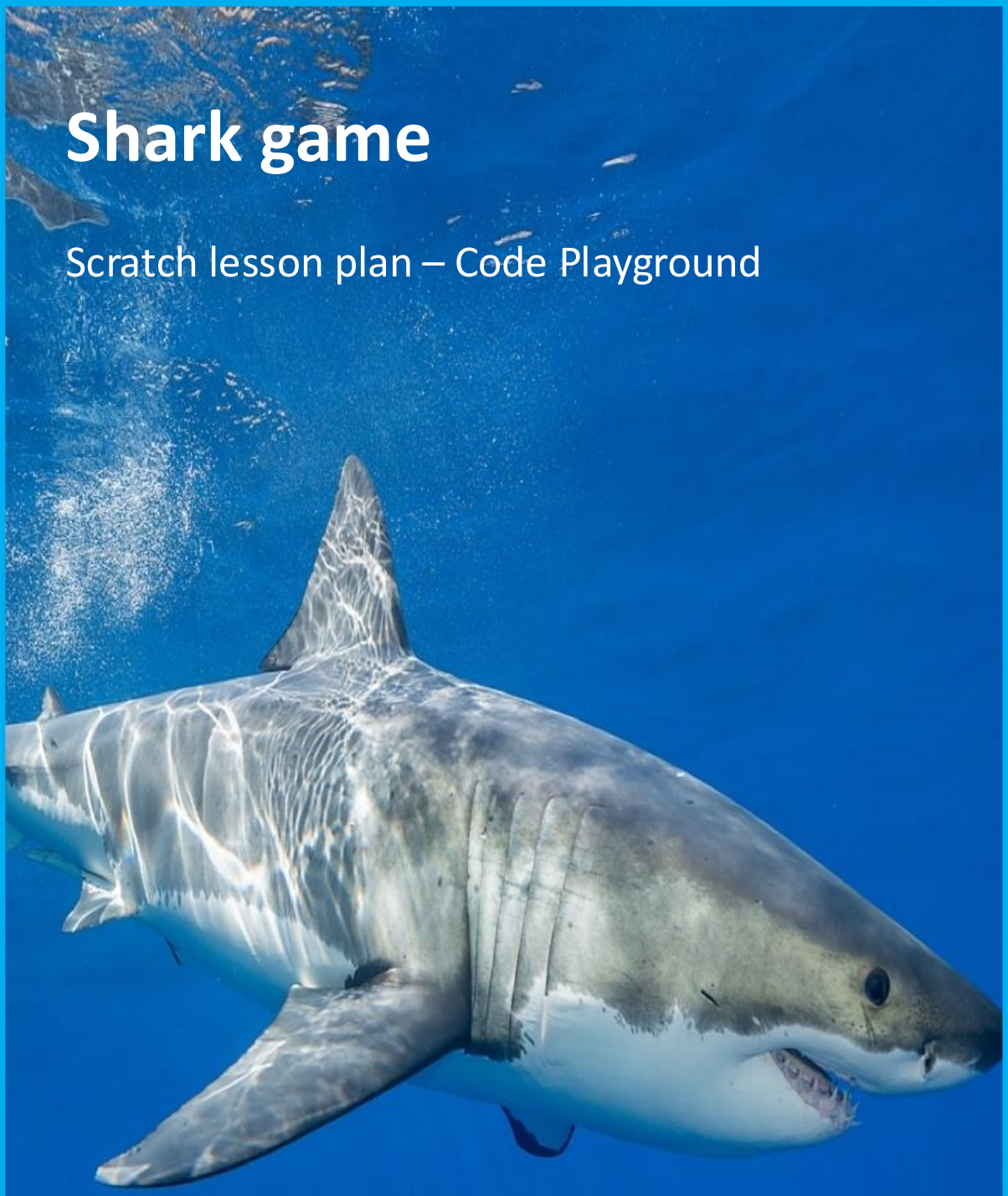
Shark game

Scratch lesson plan – Code Playground



BARCLAYS

Lesson overview

In this lesson, students will create an interactive game in Scratch called "Shark game," where a shark must catch fish while avoiding bad fish. Students will learn the basics of sprites, backdrops, and scripts while exploring game mechanics like scoring, movement, and end conditions. By the end of the lesson, students will have a functional game and understand how to expand it with custom features.

Time	Key learning outcomes	Resources
45 mins	• Learn how to select and animate sprites and backdrops in Scratch • Use scripts to program movement and interaction between sprites • Understand and implement scoring and end-game logic.	• Laptops or desktop computers • Access to Scratch website - https://scratch.mit.edu.

Content

Introduction

Begin by introducing the "Shark game" concept.

Today, we'll create a game where a shark eats fish to score points but must avoid bad fish that lower the score.

Real-life connection

This is similar to games like Pac-Man or Snake, where characters must avoid or interact with certain objects.

The activity in this project will help you understand how to create animations, use scoring, and program sprite interactions. You can also see how to make a timer to control how long the game lasts.

Scratch practical

Ask the children to log into Scratch and locate the background and sprites as described in the workbook. Show the shark game video as a guided lesson pausing regularly when the pupils need to catch up.

- Children should be able to follow along with the workbook or the guided lesson video
- By the end of the lesson children should be able to create a working program showcasing the Shark game.

Activity – Shark game

This project uses a number of features to create a fun underwater game.

Encourage students to consider the most appropriate back drop for the game.

Demonstrate how to program the character's movement using arrow keys or touch inputs.

"What happens if you change the numbers in the movement blocks? How does it affect speed?"

This programme uses collision detection. Discuss with the class why this is important to the project and how it works.

Scratch practical

Using the video and workbooks support the children to follow the instructions and complete the coding project. Have them think of other ways to enhance the project if they have extra time.

Activity wrap up

Prepare to share your project with the class

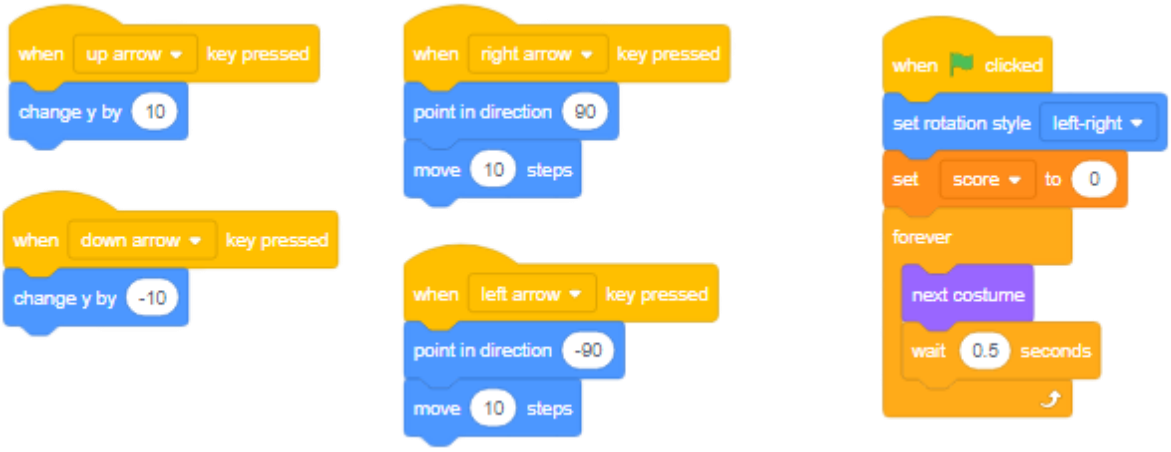
- What other ways could you end the game?
- Could you add a timer or health system?

Encourage customisation:

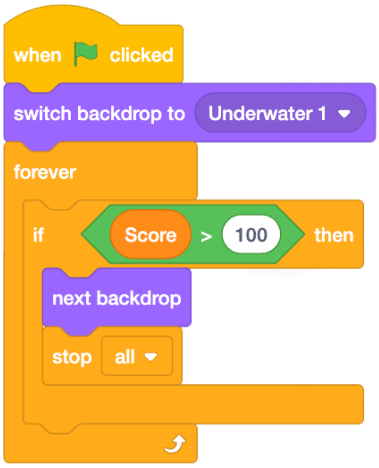
- Introduce multiple fish sprites with different movement speeds
- Add sound effects for eating fish or hitting bad fish.

Code snippets

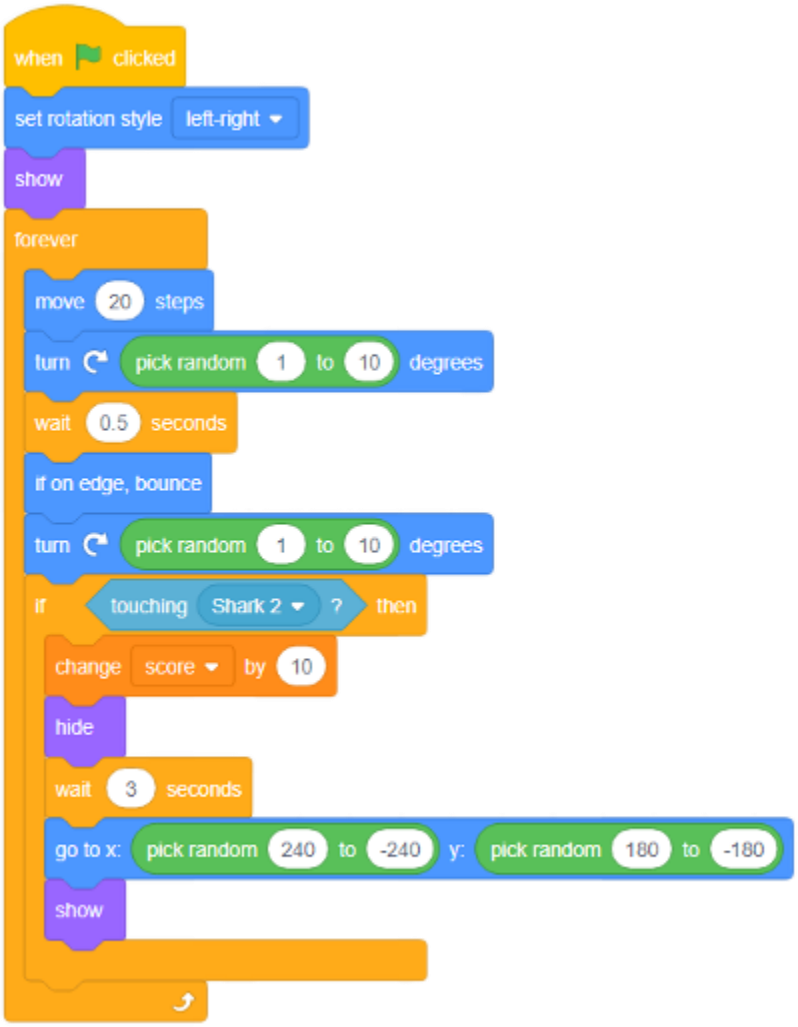
Shark sprite



Background



Fish sprite



Extension activities

Cheat sheets are provided in the back of the workbook for the children to code along.

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

- Provide visual guides or printouts of the scripts
- Focus on setting up the shark sprite and its movement before adding other features
- Allow paired work for additional support.

Differentiation – Higher ability/extension

- Challenge students to create additional features like a timer, multiple levels, or custom animations
- Encourage experimentation with mouse controls or advanced collision effects
- Introduce custom variables to track different game elements.

Plenary

- What made the game fun or challenging to build?
- How do variables like score and time improve the game experience?
- What other features could you add to make the game more exciting?

Assessment questions

1. How did you program the shark to move?
2. What block did you use to make the fish move randomly?
3. How does the game detect when the shark eats the fish?