

LESSON 25

Fruit Frenzy

60 minutes (2 periods)

Introduction to Game Design

WHAT THIS LESSON DOES

In this lesson, you'll create an interactive game called 'Fruit Collector' using MakeCode Arcade. You'll learn how to create and control a character, generate falling fruits and harmful rocks, and manage scoring and lives. The lesson covers game loops, character controls, collision detection, and more. No prior coding experience is required.

WHAT STUDENTS SHOULD BE ABLE TO DO

- Develop skills in using MakeCode Arcade to create a game.
- Understand and implement game loops and character controls.
- Learn and apply concepts of collision detection and scoring.
- Gain knowledge in creating and controlling game elements such as sprites.
- Enhance problem-solving skills by adding complexity to the game.

BEFORE THE LESSON

- Open <https://arcade.makecode.com> on the school network and check the filter allows it.
- One device per pupil or pair, charged and with sound available if needed.
- A shared screen or projector helps when modelling the first code blocks.

Fruit Frenzy

HOW THE LESSON RUNS

Introduction

Create a new project

Create Player Character

Setting Player Controls

Create the Fruit sprites

Make them fall

Create Harmful Rocks

Set the number of lives

Detect overlap with fruit

Detect overlap with rocks

Wrap Up

TEACHING MOVES

- 1. Introduction. Frame the goal: build Fruit Collector, control a character, gather falling fruit, and dodge rocks while learning loops, controls, collisions and score.
- 2. Create a new project. Get every pupil onto arcade.makecode.com and into a new blank project before any blocks are added. Name it anything clear.
- 3. Create Player Character. Model creating the player sprite together. Pause until every screen shows the character before moving on.
- 4. Setting Player Controls. Place the sprite at the bottom and enable left and right only. Let pupils test movement before the next step.
- 5. Create the Fruit sprites. Show the `onUpdateInterval` that spawns a strawberry every 2 seconds along the top. Check each project is creating fruit.
- 6. Make them fall. Make fruit fall, then destroy it off the bottom. Stress why: leftover sprites pile up and the game gets laggy.
- 7. Create Harmful Rocks. Add rocks on a 5-second timer, falling like fruit. Have pupils draw their own rock image for `mySprite3`.
- 8. Set the number of lives. Set lives to 3. Explain rock hits will remove lives once the collision code is in place.
- 9. Detect overlap with fruit. On Player and Food overlap, add 1 to score and destroy `otherSprite`, not the player. Play-test that points rise.
- 10. Detect overlap with rocks. On Player and Enemy overlap, lose a life, destroy `otherSprite` with fire, and shake the camera. Then chase high scores.
- 11. Wrap Up. Celebrate the finished game. Invite extensions named in the lesson: more fruit or rock types, speed changes, or levels.

WHAT TO WATCH FOR

- Pupils put the player sprite in the destroy block instead of otherSprite, so the character vanishes on contact. Freeze the class on the overlap block. Ask which sprite should disappear, then point them to otherSprite.
- Fruit or rocks are created without the matching SpriteKind (Food or Enemy), so overlap events never fire. Have them click the sprite kind on each create block and check it matches the overlap they wrote.
- Pupils forget to destroy sprites that leave the screen, so the game fills with off-screen fruit and rocks and slows down. Replay a laggy run, then show the off-screen destroy and ask why the game needs to forget those sprites.

DIFFERENTIATION

- Emerging: Sit them with a partner and model one block at a time until the player sprite appears and moves. Stay on create and controls longer; join the later spawn steps once movement works.
- Developing: Ask them to explain why fruit is destroyed off-screen before they add the overlap blocks. Prompt through the fruit overlap first, then have them mirror the pattern for rocks.
- Proficient: Challenge them to add another fruit type or change spawn speed once the base game runs. Ask them to try a simple level idea, such as faster rocks after a set score.