

# AUTISMFIT

Promoting the Inclusion and Wellbeing of Youth with  
Autism



## GAME 10

Level 1-2

## Table of Contents

|                                                                                    |          |
|------------------------------------------------------------------------------------|----------|
| Introduction to Makey Software for Game Creation .....                             | 3        |
| Why Makey?.....                                                                    | 3        |
| Key Features and Capabilities .....                                                | 3        |
| Makey Makey: A Fun Tool for Learning Nutrition .....                               | 4        |
| Setup Makey Makey .....                                                            | 4        |
| <b>Experiment 4: Taste Test Soundboard – Creative Expression Through Food.....</b> | <b>5</b> |
| Experiment Explanation .....                                                       | 5        |
| Coding Overview .....                                                              | 6        |
| Outcome .....                                                                      | 8        |

## Introduction to Makey Software for Game Creation

This document provides an overview of Makey software - a tool that enables the creation of interactive and accessible games, which also can be beneficial for youth with autism. Designed to support interactive learning, Makey offers schools and parents a direct way to introduce **STEM-based projects** into classrooms and homes.

Since its inception in 2010, Makey has become a popular platform worldwide, known for its simplicity and extensive resources that make learning projects easy to find and implement.

This document will walk you through Makey's key features and capabilities, highlighting its potential for creating customizable, interactive games. With a focus on sensory engagement and adaptability, Makey is a powerful tool for developing games that are both educational and enjoyable.



## Why Makey?

Makey software is a platform that simplifies **game creation**, making it accessible even for those **without a technical background**. With its user-friendly interface, visual scripting, and sensory customization, Makey is an ideal tool for developing interactive, multi-sensory experiences with an aim to diverse cognitive and sensory needs.

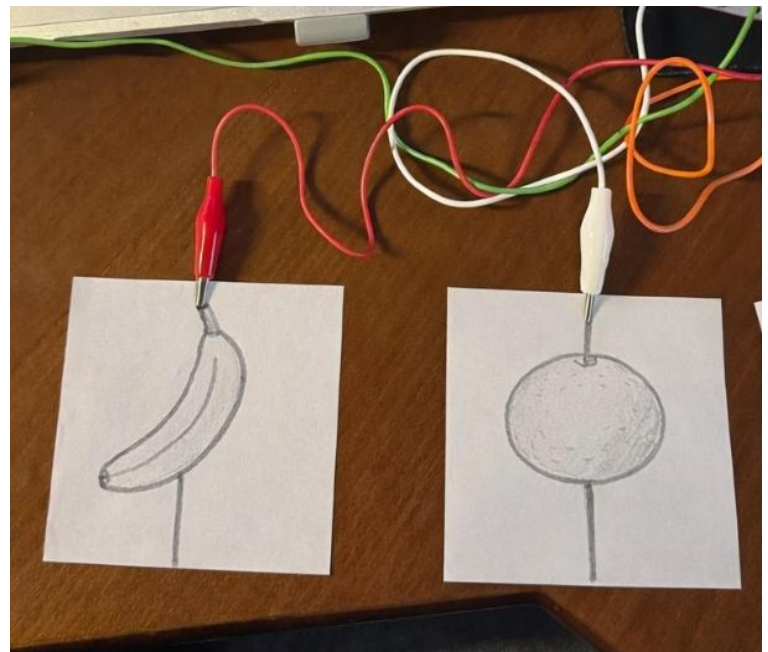
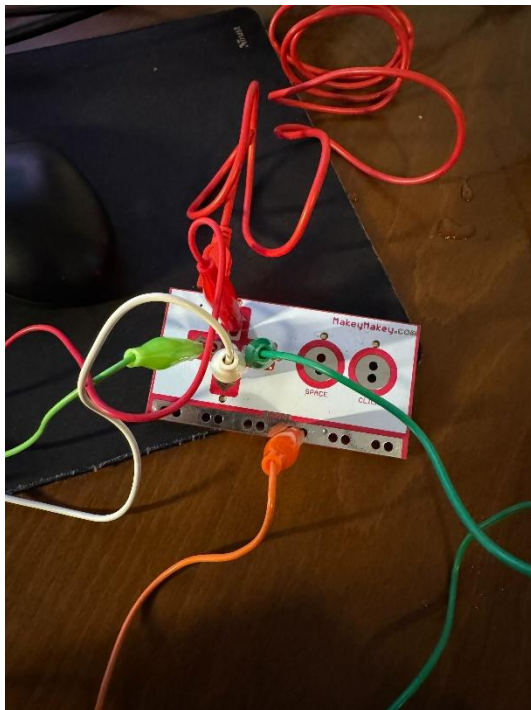
## Key Features and Capabilities

- **Customizable Sensory Inputs:** Makey supports various input types, such as sound, visuals, and touch, allowing us to adapt games to the sensory preferences of **youth with autism**.
- **Visual Scripting:** Instead of traditional coding, Makey uses visual **block coding**, enabling game creation through drag-and-drop logic blocks that simplify the design process.
- **Adaptable Game Elements:** Game components can be adjusted easily, sensory effects, and interactions, making it easy to create content that suits each **user's comfort level**.
- **Educational and Recreational Focus:** Makey supports a wide range of game types, from memory and matching games to interactive storytelling, all of which can be customized to improve development and social engagement.

## Makey Makey: A Fun Tool for Learning Nutrition

Makey Makey makes learning about food and nutrition interactive and memorable by turning simple drawings of fruits into touch-responsive game controls. By using their own pencil-drawn bananas and oranges to play, young people physically interact with food concepts, strengthening recognition, familiarity, and quick decision-making. This hands-on, game-based experience combines creativity, movement, and technology to reinforce healthy food awareness in a fun, engaging way that encourages lasting positive learning habits.

### Setup Makey Makey

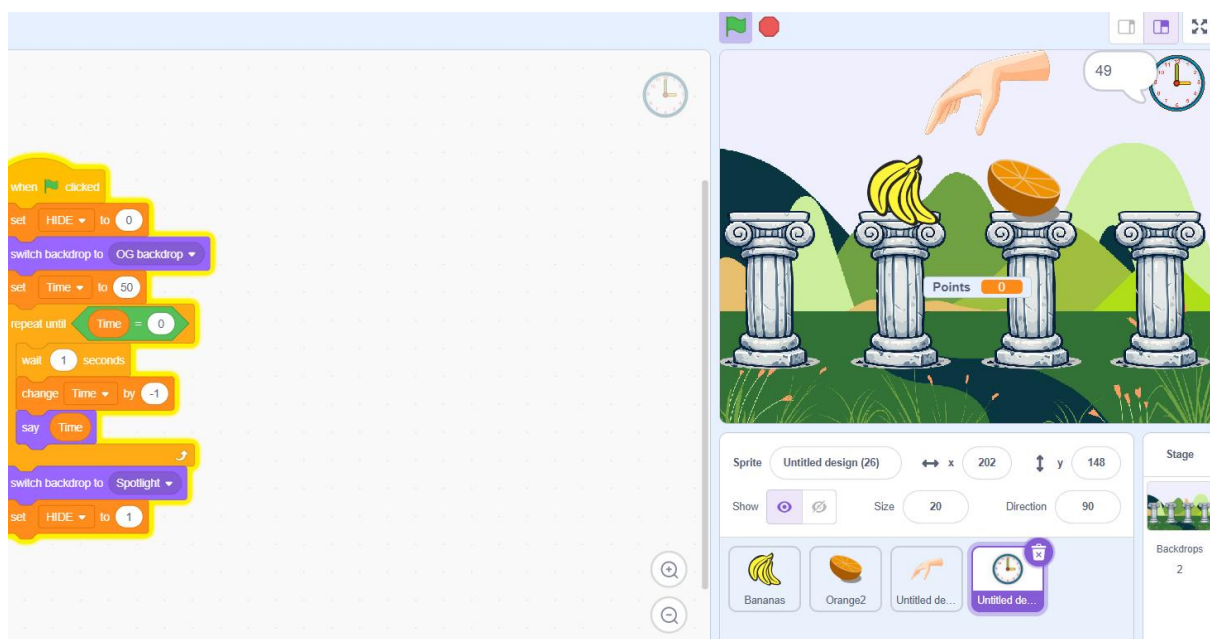


### To set up the Makey Makey for this game:

1. Plug the Makey Makey into the computer using a USB cable.
2. Connect one alligator clip to the ground (Earth/GND) on the Makey Makey. The player must hold or touch the ground clip at all times for the system to work.
3. Connect **two** alligator clips to the **up** and **down** arrow inputs on the Makey Makey.
4. Attach the other end of each clip to a pencil drawing, making sure the pencil line touches the clip so it is conductive: **banana (up arrow)** and **orange (down arrow)**.
5. Open the Scratch project on the computer. Touching the banana acts like pressing the **up** arrow, and touching the orange acts like pressing the **down** arrow, allowing players to control the game during the 50-second round.

## Experiment 4: Play a video game with a drawing LEVEL 1

### Experiment Explanation



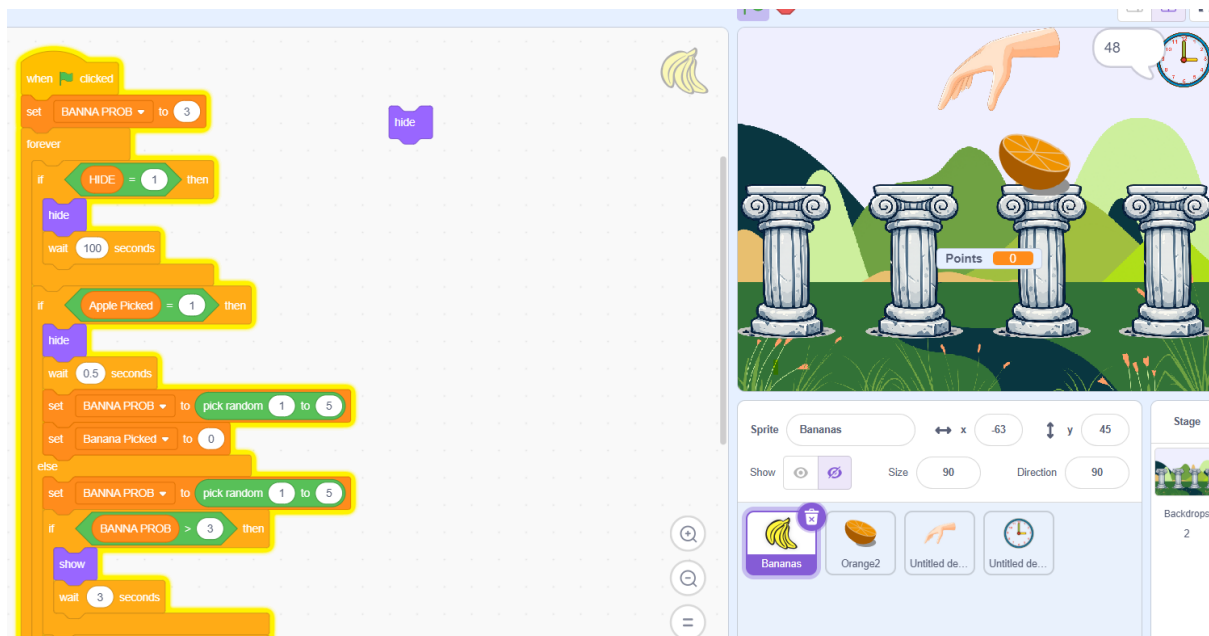
- **Goal:** Challenge players to catch as many fruits as possible in a fun, interactive game using Makey Makey and Scratch. Players use their own pencil drawings as controllers while practicing quick reaction and coordination skills.

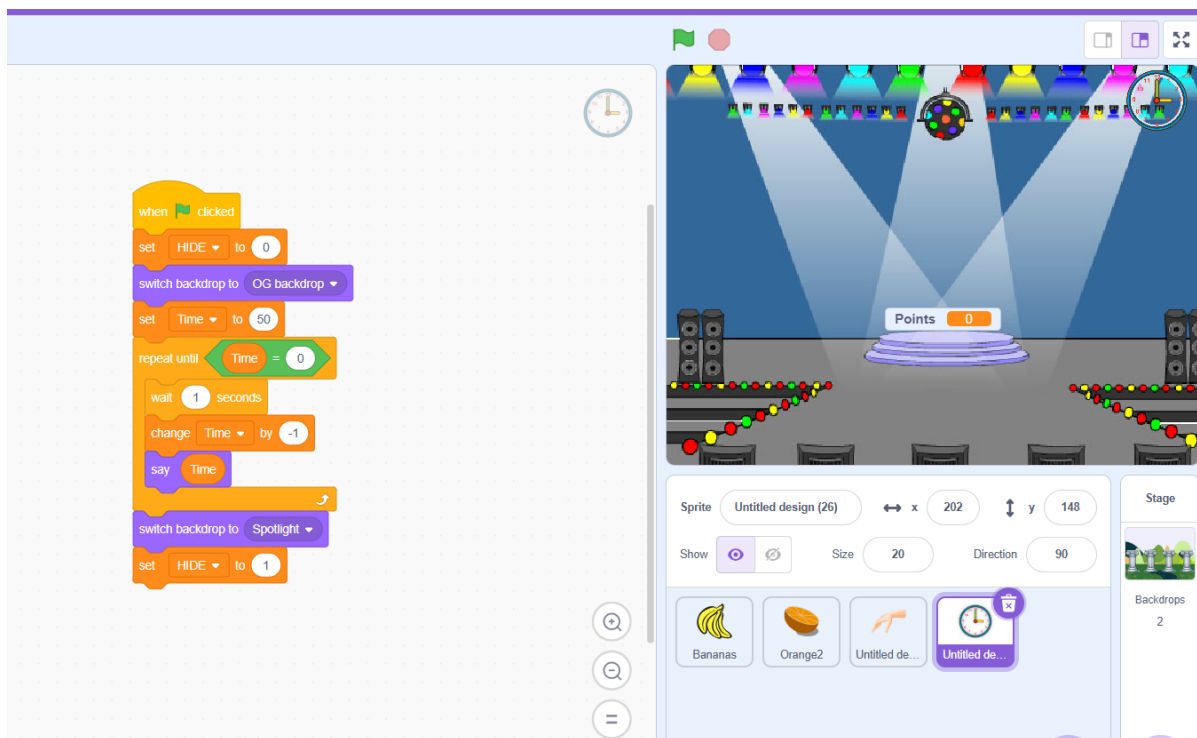
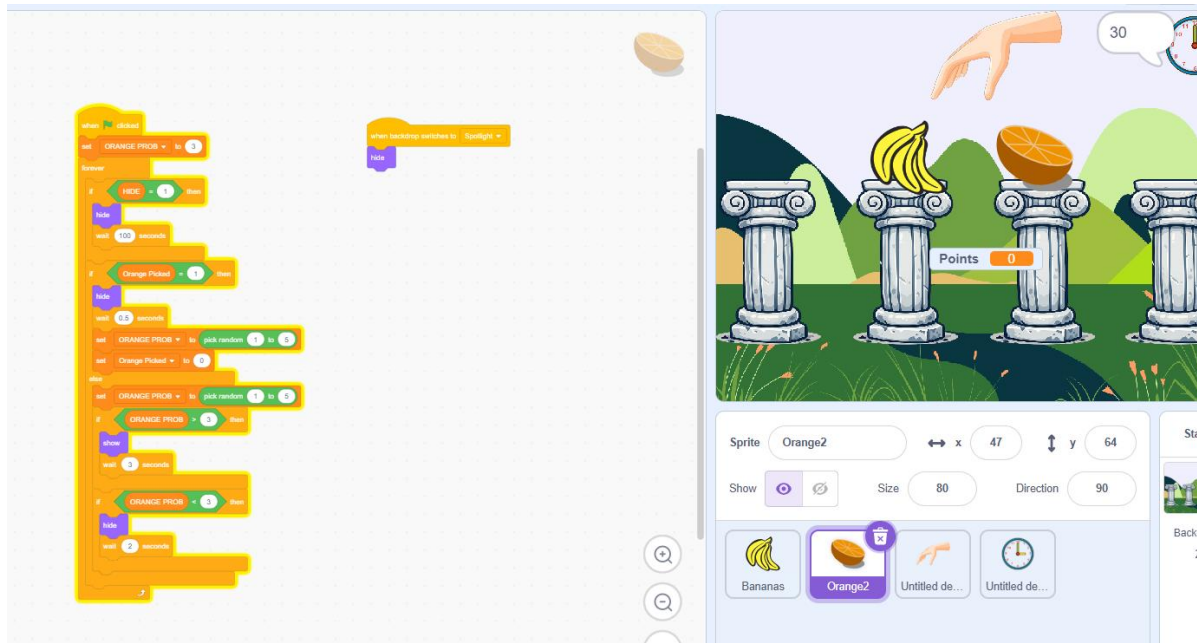
- **Setup:** Pencil drawings of a **banana (up arrow)** and an **orange (down arrow)** are connected to the Makey Makey using alligator clips.

- **Interaction:** In this game, fruits appear on the screen as Scratch sprites at random times. Players interact by touching the matching pencil drawing to trigger the correct arrow key and catch the fruit with a virtual hand before it disappears. This combines drawing, physical touch, and immediate on-screen feedback.
- **Why This Is Engaging:** The game is engaging because it turns players' own drawings into functional controllers, blending creativity with technology. The **50-second** time limit adds excitement, while instant visual feedback reinforces cause-and-effect learning and encourages replay, collaboration, and discussion.

## Coding Overview

- **Each key trigger:** Each pencil-drawn fruit is connected to a specific Makey Makey key. Touching the **banana** triggers the **up arrow**, and touching the **orange** triggers the **down arrow**. The Scratch game responds by moving the virtual hand to catch the fruit. Because the game depends on precise timing, sprite behavior, and input responses, even small changes to the key mappings can be challenging.
- **The code structure:** The Scratch project uses multiple **when key pressed**, **if-else**, and **timing** blocks to control fruit spawning, movement, scoring, and the **50-second** game timer. Since many elements are connected and interact with one another, modifying or expanding the game requires careful adjustments, making it a more complex project to change or extend.



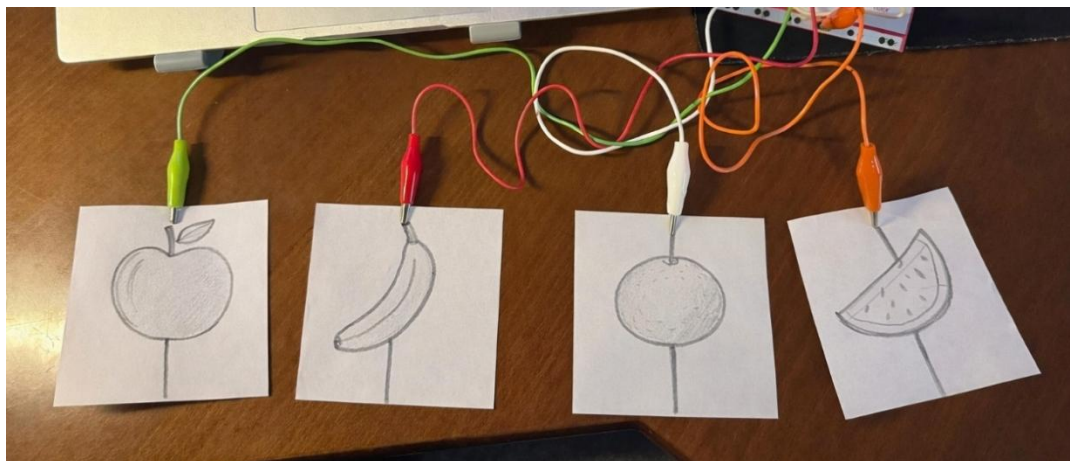
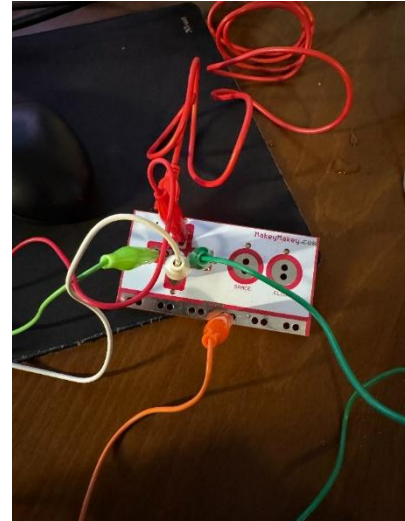


## Outcome

- Players interact physically with **pencil-drawn banana and orange controllers** to catch fruits, making gameplay active and engaging.
- Players receive **instant visual feedback** as the virtual hand catches fruits on screen and the score updates in real time.
- The hands-on Makey Makey setup encourages **learning through play** and keeps players focused and motivated.
- The game promotes **discussion and reflection** as players compare scores, strategies, and reaction times after each **50-second** round.
- Players learn core coding concepts such as **key detection (up/down arrows), if-else logic, timing, and variables** through an interactive Scratch project.
- While the game can be expanded, adding new fruits or features requires **careful planning and adjustments** due to the game's interconnected logic and timing.

## Level 2

### Setup Makey Makey

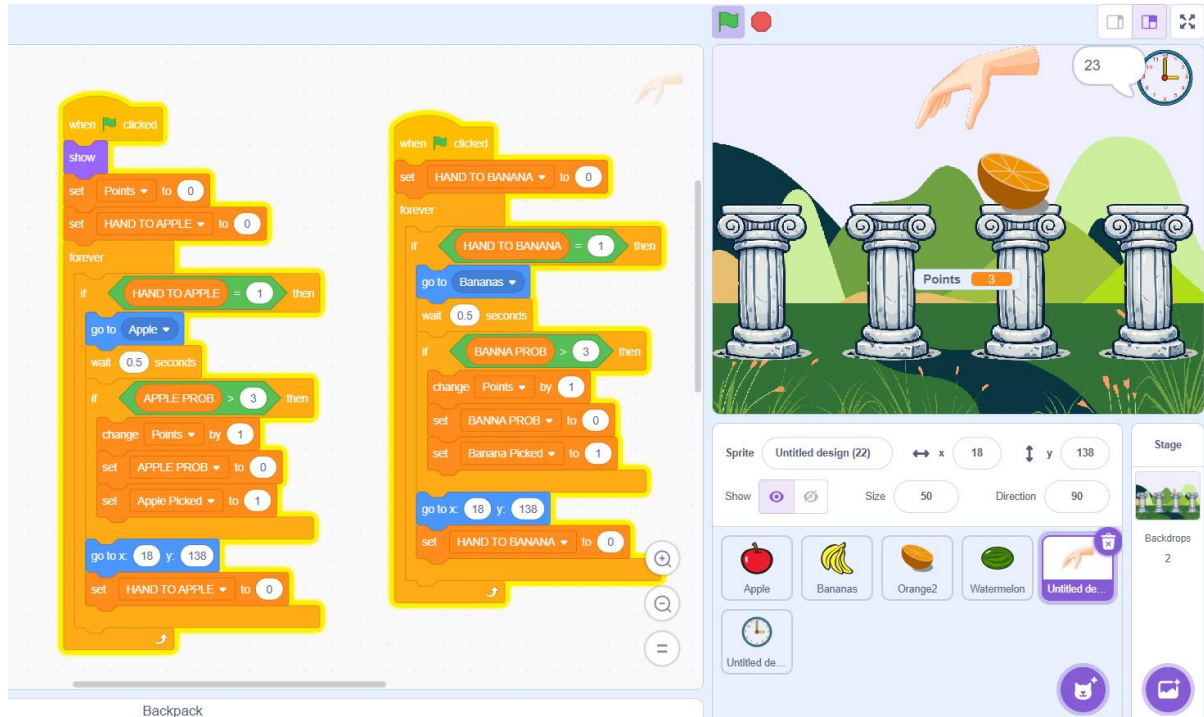


To set up the Makey Makey for this game:

1. Plug the Makey Makey into the computer using a USB cable.
2. Connect one alligator clip to the **ground (Earth/GND)** on the Makey Makey; the player must hold or touch the ground clip at all times for the system to work.
3. Connect four alligator clips to the **left, up, down, and right arrow** inputs on the Makey Makey.
4. Attach the other end of each clip to a **pencil drawing**, making sure the pencil line touches the clip so it is conductive: **apple (left arrow)**, **banana (up arrow)**, **orange (down arrow)**, and **watermelon (right arrow)**.
5. Open the Scratch project on the computer. Touching a drawing now acts like pressing a keyboard key, allowing players to catch the matching fruits on screen during the 30-second game.

## Experiment 4: Play a video game with a drawing LEVEL 2

### Experiment Explanation



- **Goal:** Challenge players to catch as many fruits as possible in a fun, interactive game using Makey Makey and Scratch. Players use their own pencil drawings as controllers while practicing quick reaction skills.

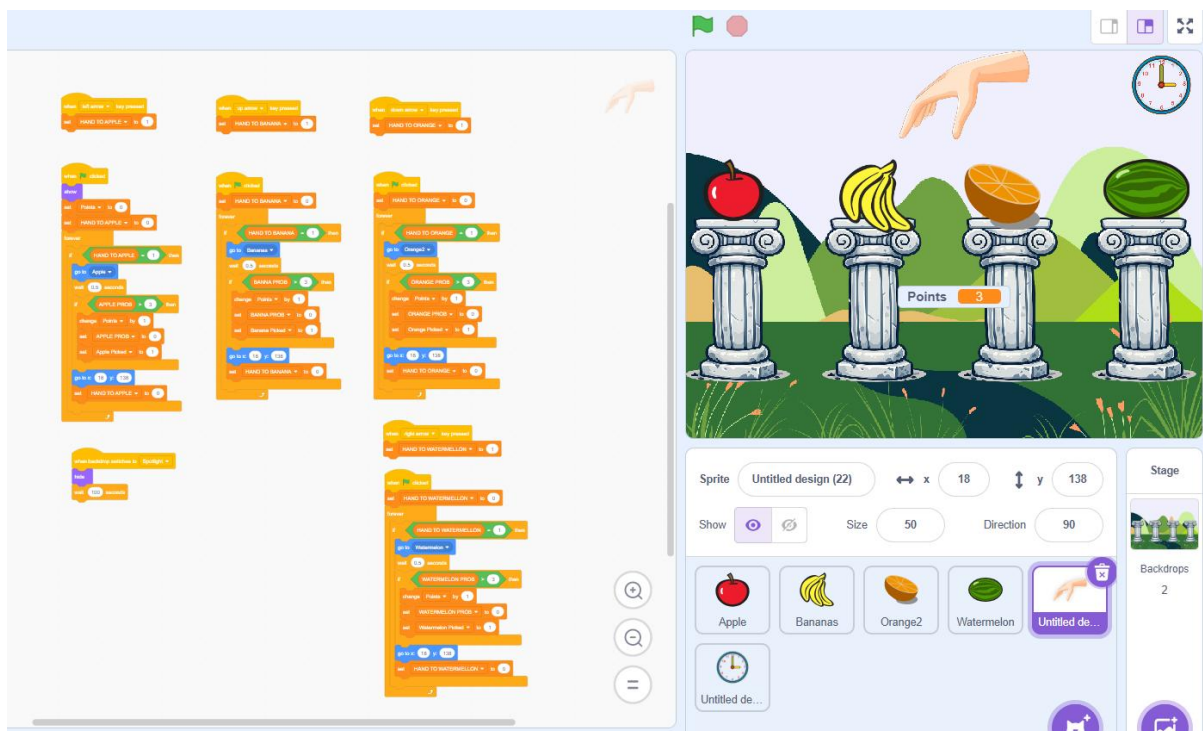
- **Setup:** Pencil drawings of **apple (left arrow)**, **banana (up arrow)**, **orange (down arrow)**, and **watermelon (right arrow)** are connected to the Makey Makey using alligator clips.

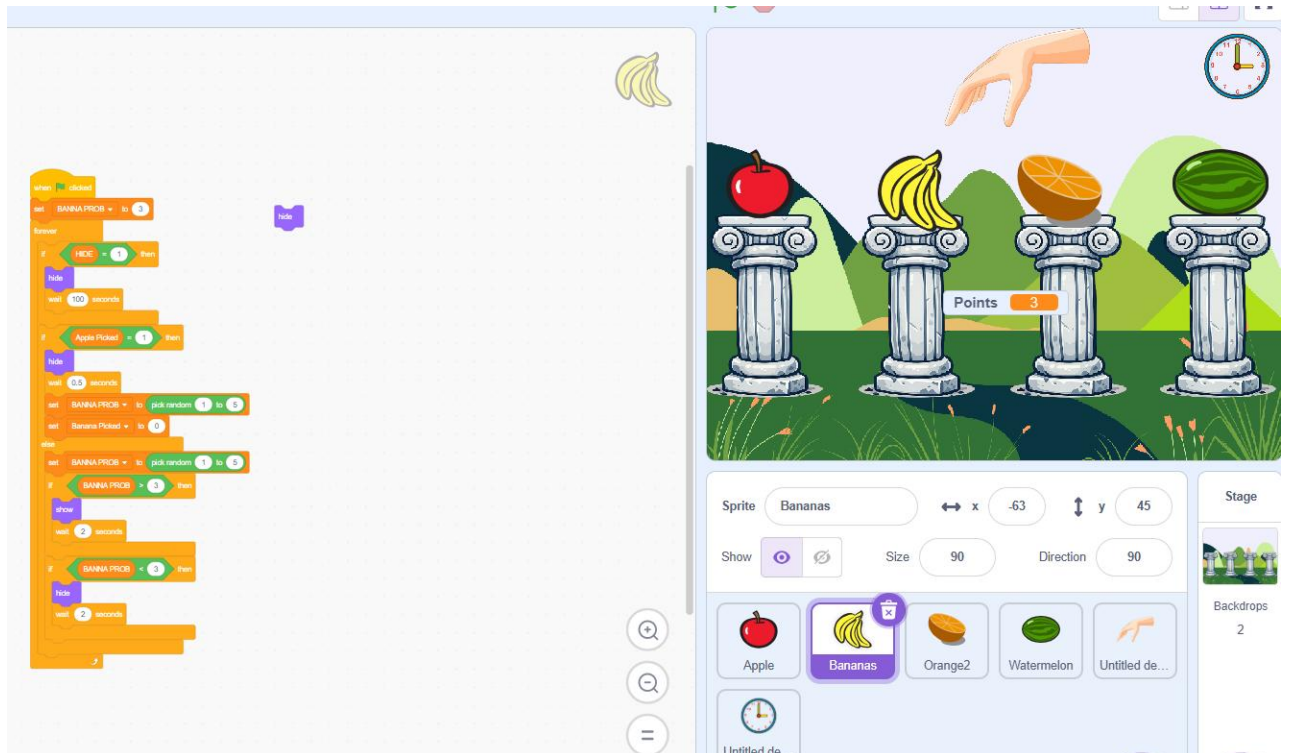
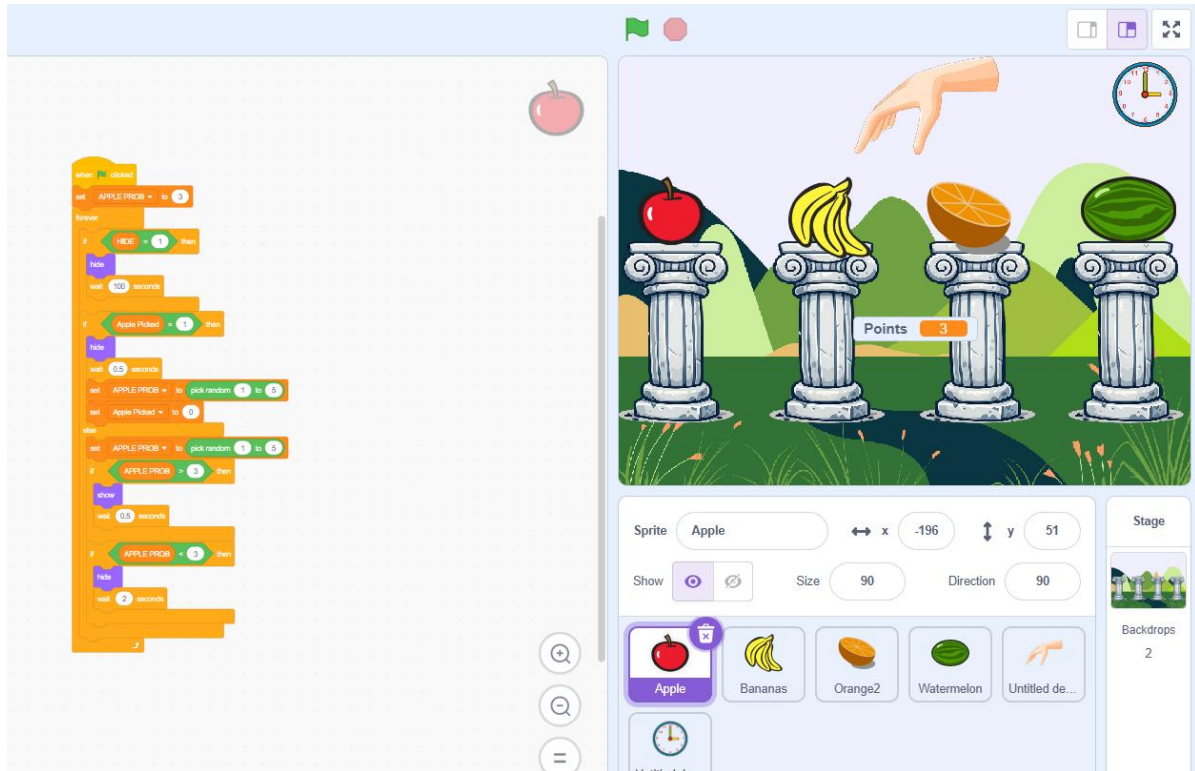
- **Interaction:** In this game, fruits appear on the screen as Scratch sprites at random times. Players interact by touching the matching pencil drawing to trigger the correct arrow key and catch the fruit with a virtual hand before it disappears. This combines drawing, physical touch, and immediate on-screen feedback.

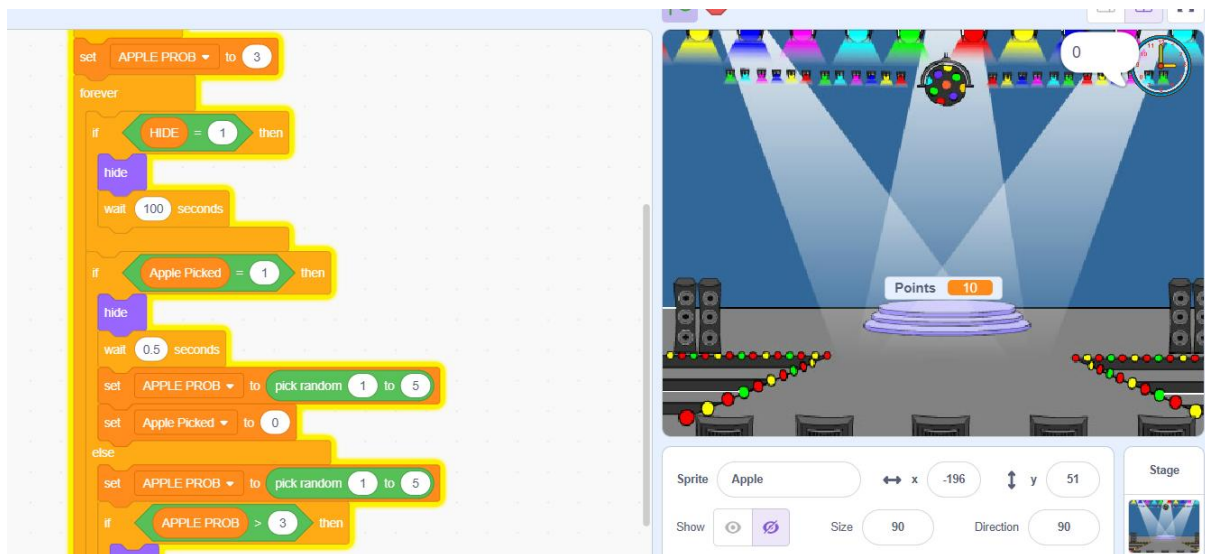
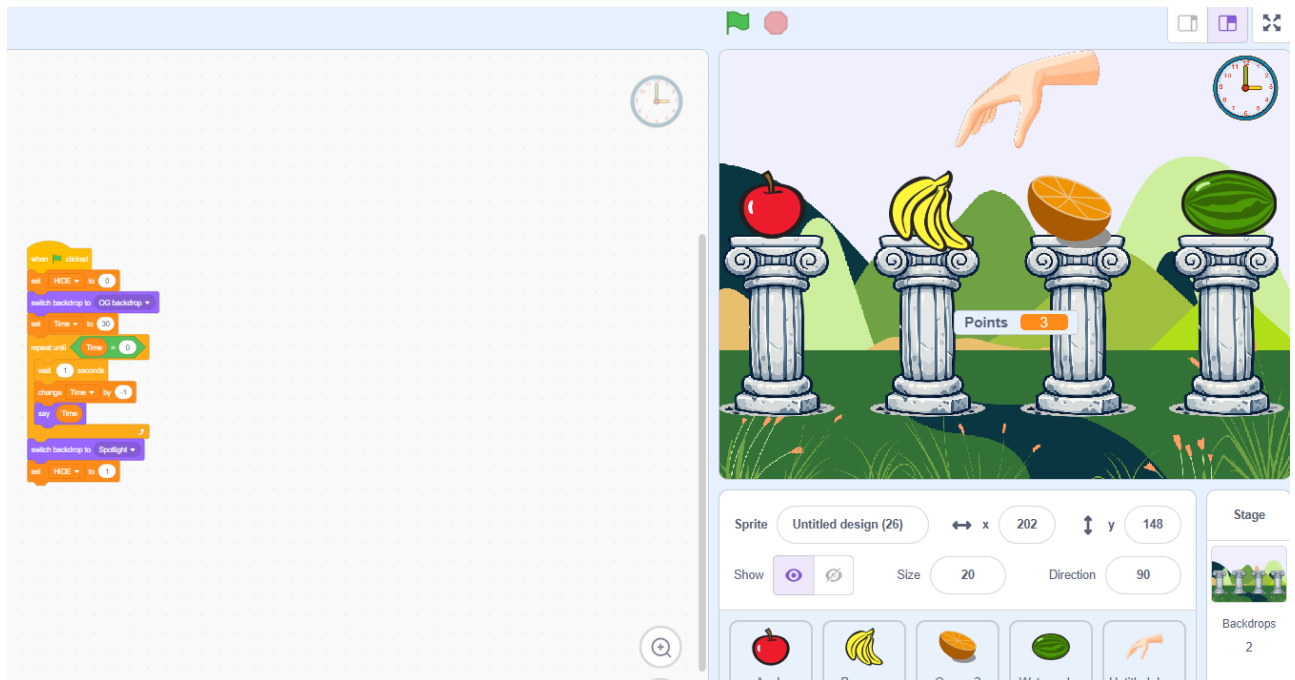
- **Why This Is Engaging:** The game is engaging because it turns players' own drawings into functional controllers, blending creativity with technology. The 30-second time limit adds excitement, while instant visual feedback reinforces cause-and-effect learning and encourages replay, collaboration, and discussion.

## Coding Overview

- **Each key trigger:** Each pencil-drawn fruit is connected to a specific Makey Makey key. When the player touches a drawing, the corresponding key is triggered and the Scratch game responds by moving the virtual hand to catch the fruit. Because multiple timings, sprites, and controls are involved, even small changes to the key mappings can be complicated.
- **The code structure:** The Scratch project relies on multiple **when key pressed**, **if-else**, and timing blocks to manage fruit spawning, movement, scoring, and the 30-second timer. Due to the number of connected elements and interactions, modifying or expanding the game requires careful adjustments, making it a more complex project to change or extend.







## Outcome

- Players interact physically with pencil-drawn controllers to catch fruits, making gameplay active and engaging.
- Players receive instant visual feedback as the virtual hand catches fruits on screen and scores update in real time.
- The hands-on setup encourages learning through play and keeps players focused and motivated.
- The game promotes discussion and reflection as players compare scores, strategies, and reaction times.
- Players learn core coding concepts such as key detection, **if-else** logic, timing, and variables through an interactive Scratch project.
- While the game can be expanded, adding new fruits or features requires careful changes due to the game's complexity.