1.3 Build Your First Game

Duration: 60-65 minutes

Students will be able to:

1. Use code blocks to control a sprite's movement in a game.
2. Use loops, conditions, and sensing blocks to make a game interactive.
3. Create a simple game that tracks score and responds when sprites touch.

Learning Activity Summary

- Warm-up: introduce programming and the game goal (5 minutes).
- Guided build: reopen project, set start points, add dog collision logic, add a forever loop, build apple reset logic, and create a score variable (45-50 minutes).
- Clean-up, save/share, student feedback, assessment, and Q&A (10 minutes).

Student Materials

- Computer with Internet access
- Saved project from the previous lesson
- Lesson notes or workbook

Teacher Preparation

- 1.3 lesson slide deck projected
- Prepare a teacher reference project
- Prepare submission instructions and rubric.

Teacher Notes / Differentiation

- **Support:** Provide starting blocks, pair students for peer walkthroughs, and remind students to start scripts with the green flag.
- **Extension:** Advanced students can use different sprites, resize the apple or dog during play, create a snake-game effect, or add a victory sound when the score reaches a target.
- **Teaching tip:** Guide students step by step; do not show the full code at once. Emphasize testing after adding each block and using errors as learning moments.

Key for this Plan:

 = explicitly explain

 = student action

 = teacher tip

Engage & Explore (5 minutes)

Slide 1

Introduction to Programming

Welcome students and connect the lesson to the project they will build.

"Programming is giving instructions to a computer to make it do what you want."

"Today we are going to make a game where you use the keyboard to move the dog sprite to eat the apple."

+ Student action

Students open CreatiCode, identify the dog and apple sprites, and describe the goal of the game in one sentence.

! Checkpoint

Students can state the game objective: move the dog, touch the apple, and make the game respond.

Slide 2

Overview Agenda

Review the build sequence and explain that students should test after each step.

"First, we will reopen the project from last lesson. Then we will set starting positions for the dog and apple. After that, we will make the dog detect the apple, add a forever loop, move the apple when it is eaten, and create a score variable."

"Each sprite has its own code area, so make sure you select the correct sprite before adding blocks."

+ Student action

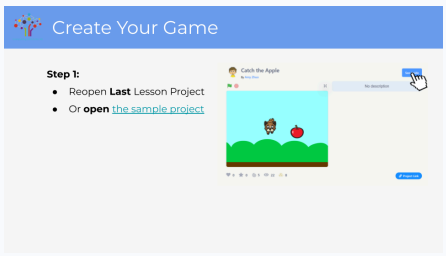
Students follow the agenda and prepare to build one feature at a time.

! Checkpoint

Students understand that the code should be tested after each new block group is added.

EXPLORE / GUIDED BUILD (45-50 minutes)

Slide 3



Step 1 - Reopen Last Lesson's Project

"Open your saved project from last lesson. Make sure the dog sprite and apple sprite are both visible."

"Name or rename the project clearly so it is easy to find and submit later."

+ Student action

Students sign in, reopen the previous project, confirm the dog and apple sprites are present, and save the project with a clear name.

! Checkpoint

Every student has the correct project open with the dog and apple visible.

Slide 4

Step 2 - Set the Start Point

Step 2 - Set the Start Point

"When the green flag is clicked, the sprite should go to a starting place. For the apple, use go to random position so it appears somewhere new when the game starts."

"Add starting blocks to the correct sprite. Blocks placed on one sprite do not automatically control the other sprite."

+ Student action

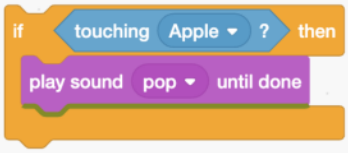
Students add green flag starting blocks and set initial positions for the dog and/or apple. They click the green flag to test the start state.

! Checkpoint

Clicking the green flag places the sprites as expected before the game begins.

Slide 5

Step 3 - Dog Collision Logic



Step 3 - Build the Core Game for the Dog

"Now we need the dog to know when it touches the apple. Use an if then block with the touching Apple sensing block. Inside the if then block, play the pop sound."

"Run it once. If you do not hear the sound right away, that is useful information: the check only happens one time. We will fix that by adding a loop next."

+ Student action

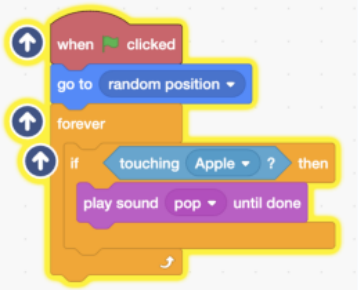
Students select the dog sprite and add if touching Apple then play sound pop until done.

! Checkpoint

Students can point to the condition and the action inside the if then block.

Slide 6

Step 4 - Add a Forever Loop



Step 4 - Add a Forever Loop

"A loop means the code runs again and again. Put the if touching Apple check inside a forever loop so the dog keeps checking for contact."

"Click the green flag, move the dog, and test what happens when the dog reaches the apple."

+ Student action

Students wrap the dog collision check inside a forever loop and test by moving the dog to the apple.

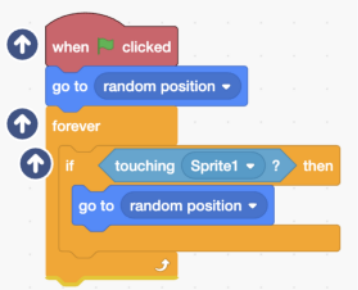
! Checkpoint

The sound or feedback happens when the dog touches the apple because the code keeps checking continuously.

Slide 7

Step 5 - Build the Core Game for the Apple

Step 5 - Apple Reset Logic



"The apple also needs its own code. Select the apple sprite. When the green flag is clicked, place the apple at a random position."

"Then, forever check whether the apple is touching the dog. If it is touching the dog, send the apple to a random position."

+ Student action

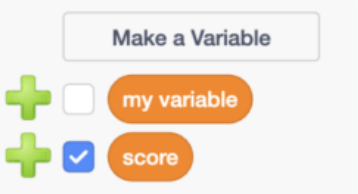
Students select the apple sprite and build the apple script: start at a random position, forever check touching the dog, and move to a random position when touched.

! Checkpoint

The apple jumps to a new random position whenever the dog reaches it.

Slide 8

Step 6 - Add a Score



Step 6 - Add a Score

"Now we will create a variable. A variable is a value that can change while the game is running, like a score or timer."

"Make a score variable, set it to 0 when the game begins, and change it when the dog touches the apple."

"Optional: students can change the variable name or change how much it increases, such as change BOOM by 200."

+ Student action

Students create a score variable, reset it at the start, and add a change score block in the collision logic.

! Checkpoint

The score increases reliably each time the dog eats the apple.

PRACTICE / CLEAN-UP / FEEDBACK (10 minutes)

Slide 9

Clean-up and Feedback

Clean-up and Feedback

"Before you submit, name, save, and share your project. Test the full game from the green flag."

"The order of independent scripts may not matter, but each sprite needs the correct blocks in its own code area."

"Who can explain one block you added and what it does in the game?"

+ Student action

	Students run one final playtest, save/share the project, and 1-2 students explain what they added. **! Checkpoint** Students have a saved project link and can describe at least one block group: motion, condition, loop, sensing, or variable.
Slide 10 **Extensions & Differentiation**	**Optional Extension Challenge** Students who finish early can polish the game or add a creative feature. ● Use different sprites or backgrounds. ● Change the size of the apple or dog during play. ● Create a snake-game style growth effect. ● Set a victory sound or message when the score reaches a target. **+ Student action** Advanced students choose one extension. Students needing support use starter blocks, pair programming, or peer walkthroughs. **! Checkpoint** Students can explain how their extension changes gameplay or how scaffolding helped them complete the core game.

WRAP-UP / Q&A (5 minutes)

Summary

"Let us recap the steps we used to create a simple interactive game. We used code blocks to control sprites, conditions and sensing blocks to detect when sprites touch, a forever loop to keep checking, and a variable to track the score."

Invite students to name one block category they used and explain why it was important.

Q&A

Open the floor for questions. Address common challenges such as selecting the wrong sprite, forgetting the forever loop, or placing the score change in the wrong script.

Encourage students to use errors as learning moments by asking questions like: Why did the sound not play? Why did the score not increase? Why did the apple not move?