UNITY CREATOR KIT: BEGINNER CODE

https://learn.unity.com/project/creator-kit-beginner-code

Channel students' enthusiasm for *playing* games into a passion for *making* games!

Computer coding and game development are intertwined. The developers who make popular games like Mario Run, Pokemon Go, and Monument Valley, work with code everyday. In the Unity Creator Kit: Beginner Code activity, students get exposure to the coding skills and the professional technology platform (Unity) used to make games.

In this activity, students customize their own 3D roleplay video games using Unity technology and C# computer coding. All activity assets and step-by-step tutorials are included, helping learners to get started quickly. By the end of the project, students will have customized their own playable 3D games and have gained a better understanding of scripting, variables, functions, and classes.

Learn More about Unity

ACTIVITY OVERVIEW

In this activity, students explore the basics of C# code for digital interactive development using Unity, in the context of an action Role Play Game (RPG). This lesson is designed for students aged 13 +, and works best for students who have some basic experience of coding.

Before starting this project, it is recommended that students complete **one** of the three no-code introductory Creator Kits to get familiar with the Unity platform. We find that students who have some experience with the Unity interface can move more efficiently through the Creator Kit: Beginner Code activity.

Choose one of the following no-code projects here to help students get familiar with the Unity interface:

Creator Kit: FPS
Creator Kit: Puzzle
Creator Kit: RPG

Time

This project takes 3.5 hours to complete. This likely translates to 4 - 6 lessons to complete in a classroom setting. Please see our suggestions below for ways to approach breaking the activity into classroom/lesson timeframes.

ACTIVITY OBJECTIVES

Students will:

- Identify and modify key information in C# coding scripts
- Understand the role of variables, functions and classes — and how they can be used to write efficient code
- Use variables to give the computer more efficient commands
- Edit simple scripts and test the impact of code on the player experience
- Write their own code in C# programming language
- See their code come to life in a game that they create

TEACHING SUMMARY

SECTION 1	MATERIALS AND PREPARATION	90 minutes
SECTION 2	YOUR FIRST LESSON	60 minutes
SECTION 3	LESSONS 2-4	60 minutes per lesson
SECTION 4	EXTENDED LEARNING	Varies

SECTION 1: MATERIALS AND PREP - Complete before your Hour of Code

Complete Section 1 one or two weeks before your Hour of Code, or the day prior to your Hour of Code

SECTION 2: YOUR FIRST LESSON

GETTING STARTED (5 MINUTES)

Setting the Stage

ACTIVITY (30-45 MINUTES)

WRAP UP (5 MINUTES)

Debrief

SECTION 3: LESSONS 2 - 4

Next Steps

ASSESSMENT (2 MINUTES)

SECTION 4: EXTENDED LEARNING

Beyond an Hour of Code

Celebrate

SECTION 1: TEACHING GUIDE

MATERIALS AND PREP

Two Weeks Before Your Hour of Code

- Review the “Hour of Code Educator Guide” and “Best Practices from Successful Educators” in order to begin planning your Hour of Code.

- Register your Hour of Code event if you’d like to receive swag or classroom support.

- Watch Unity Hour of Code Teacher Walkthrough video

One Week Before Your Hour of Code

- Review and familiarise yourself with the tutorials and videos.

- Be sure to test the project before asking your students to complete it. Check your technology and decide if you need to troubleshoot anything in advance of your Hour of Code.

- Recommended: Have students complete the first tutorial of the activity in advance Get started with Creator Kit: Beginner Code

  This tutorial will guide students on how to download Unity, how to download the project and assets needed, and will walk them through the activities they need to complete before Hour of Code. They will need to allocate 90 minutes for the completion of the Get started with Creator Kit: Beginner Code tutorial and the prerequisite code-free introductions to digital making in Unity (Creator Kit: FPS, Creator Kit: Puzzle or Creator Kit: RPG). These activities could also be conducted during class time the week before Hour of Code.

One Day Before Your Hour of Code

- In preparation for the conclusion of your Hour of Code, print an Hour of Code Certificate for each of your students. Each student should receive a certificate to acknowledge their efforts during the Hour of Code.

SECTION 2: YOUR FIRST LESSON

GETTING STARTED (5 - 10 MINUTES)

Setting the Stage

Welcome students to class and very briefly introduce the day's activity.

To set the scene: Show one of the Hour of Code inspirational videos to frame the discussion:

- For middle school students, we recommend "The Hour of Code is Here."
- For older students, we recommend "Anybody Can Learn."

Explain that students will spend one hour on coding today in celebration of CS Education Week. They join millions of other students across the globe who are also learning programming skills during Hour of Code this week. Congratulate students on being part of the movement.

Confirm with all students that they have completed Get started with Creator Kit: Beginner Code, one of the recommended Creator Kits, and have downloaded Unity version 2019.2.

Check in to ensure students have accessed the materials and assets for the Creator Kit: Beginner Code.

Set the scene for the project by showing students the Unity Hour of Code Trailer and Getting Started Video with the students.

Explain that this project will take longer 3.5 hours to complete and that in this session they will complete the following tutorials:

Get started with code in Unity (10 minutes)
Introduction to variables (10 minutes)
Write your own instruction (15 minutes)

ACTIVITY (30-45 MINUTES)

During this first session, we recommend that students complete the following tutorials:

Get started with code in Unity (10 minutes)
Introduction to variables (10 minutes)
Write your own instruction (15 minutes)

Depending on the age and ability of your students, you might consider:

- For learners in the middle grades, we find that working independently on tutorials works well; however, we recommend students work in pairs to problem solve and troubleshoot together.
- For more experienced learners, this project works well as an independent challenge or a partner/group programming activity.
- If a group or individual finishes early, they can move onto the next tutorial Introduction to

functions (30 minutes)

Teacher Tip:
As much as possible get students to work together to solve problems. This is how coders work in the workplace. It also encourages independent, critical and creative thinking.

Teacher Tip:
Students work at different paces. After students watch the variables video, which precedes the tutorial, “Introduction to Variables”, ask them to raise their hand or otherwise let you know they have completed the video. Check for understanding by asking students to share what they’ve learned or ask them to describe the coding concept.

WRAP UP (5 MINUTES)

Debrief:

Ask students to reflect on the day’s activities. What problems did they encounter and how did they solve them and what was the outcome? What were the highlights? What was challenging?

Ask students to reflect on the concepts and skills they practiced. How will they approach the rest of the project based on their learnings from the day?

Share photos and videos of your Hour of Code event on social media. Use #HourOfCode and @codeorg so we can highlight your success, too!

SECTION 3: LESSONS 2-6

Next Steps:

This will depend on how you wish the students to complete the remainder of the project - either during class time or a combination of the students' own time and class time.

If you ask students to complete sections in their own time, at the beginning of each subsequent class, spend 10 minutes talking about what they learned and problems they encountered.

Below is a breakdown of the remaining tutorials, time it will take to complete them and recommended groupings of tutorials.

Lesson 2:
VIDEO: Functions
Introduction to functions (30 minutes)
Write your own spawn function (30 minutes)

Lesson 3:
VIDEO: Classes
Introduction to classes (25 minutes)
Write your own spawner class (30 minutes)

Lesson 4:
VIDEO: Keywords
Introduction to public keyword and inheritance (20 minutes)
Customise the health potions (20 minutes)

Recommendations for Lesson Starters and Wrap Ups

- Start the lesson with questions that ask students to share previous learnings and their progress with the tutorials. What's working? What do they need to spend more time completing? As a class, watch the video indicated in the groupings above. Have students pair up to discuss what they've learned with their partner about that coding concept.

- Wrap up the lesson with one of the **assessment activities** below.

- Share photos and videos of your Hour of Code event on social media. Use #HourOfCode and @codeorg so they can highlight your success, too!

ASSESSMENT (5 MINUTES)

Consider implementing one or more of the assessment activities below at the end of the lesson.

5x5 Journaling

Have students journal about the five most interesting ideas or concepts they discovered during the lesson. Next, they identify five things that resonate with them about each one and explain why or if their learning relate to problems they solved, explain how they solved them.

3-2-1

Have students write or talk about 3 things they learned, 2 things they still want to learn, and 1 question they have. These values are interchangeable and can be used with different questions or combinations.

Twitter Board

Have students summarise what they learned using 280 characters. To make it a little harder, ask them to do it in 140 characters. Pin these strips of paper or post it notes to resemble a Twitter feed.

SECTION 4: EXTENDED LEARNING

Beyond Hour of Code

After Hour of Code ends, there's still so much coding fun and learning to explore! Here are some extension ideas:

- Direct students to Customize their game tutorial, which is a reference guide for further customization of the Creator Kit: Beginner Code game. It provides guidance to help students use scripts templates to create:
    - Usable items with Usage Effects (like the health potion in the final tutorial)
    - Equipment items with Equipped Effects
    - Weapons with Weapon Attack Effects

- Direct students to Unity Learn, for Educators or Code.org to try other projects.

- Invite a game designer to talk to your class about his or her work. Don't know any local game designers?

Celebrate!

Once students have completed their projects, give each student a certificate with his or her name on it.