



Computer Science

16 Lessons

Grades K-2/Lower Primary/Ages 5-7

Computing with Minecraft Curriculum Overview

[EDUCATION.MINECRAFT.NET](https://education.minecraft.net)

Table of Contents

<u>GETTING STARTED WITH MINECRAFT</u>	<u>3</u>
<u>COMPUTER SCIENCE PATHWAY PROGRESSION</u>	<u>4</u>
<u>CURRICULUM SUMMARY</u>	<u>5</u>
<u>INTRODUCTION</u>	<u>7</u>
<u>MINECRAFT: EDUCATION EDITION- TEACHING & LEARNING FRAMEWORK</u>	<u>8</u>
<u>INSTRUCTIONAL SEQUENCE</u>	<u>9</u>
<u>COMPUTER SCIENCCE CONCEPTS</u>	<u>12</u>
<u>VISUAL GLOSSARY</u>	<u>13</u>

Getting Started with Minecraft

Install Minecraft: Education Edition

Minecraft: Education Edition can be installed on Chromebook, iPad, Mac, and PC. To ensure your experience with Minecraft: Education Edition is top-notch, make sure your devices meet the minimum system requirements. To check if your device supports Minecraft: Education Edition, see [System Requirements](#).

If you have questions about setting up Minecraft: Education Edition, the following link will provide you with some [frequently asked questions and additional information](#) about set-up. On this page, you will find assistance for:

- Get Started
- Purchase Licenses
- Administration and License Management
- Installation
- Troubleshooting

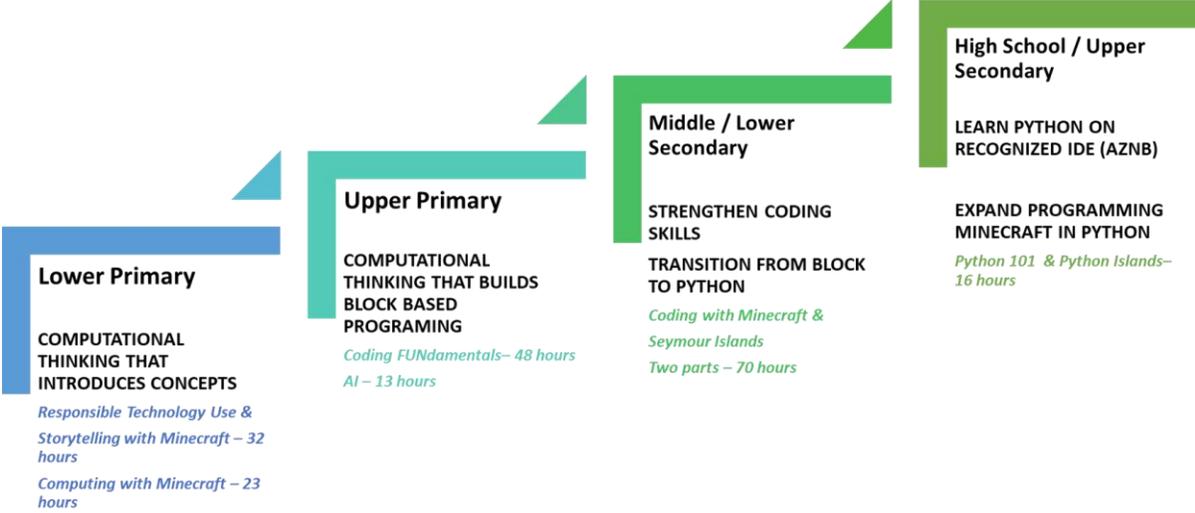
Preparing to Teach with Minecraft: Education Edition

Teachers do not need to have any prior computer science experience; however, they should familiarize themselves with a basic understanding of what is Minecraft: Education Edition. Support for building out teacher knowledge can be found here:

[Minecraft: Education Edition: Teacher Academy](#)

[Minecraft: Education Edition Webinar Series](#)

Computer Science Pathway Progression



Computing with Minecraft is a part of the Lower Primary (Grades K-2, Ages 5-7) computer science progression. In this part of the progression, students are learning computational thinking and beginning computer science concepts. Students will use Code Builder to create code using MakeCode word blocks. In this unit, students will be able to build out their content knowledge and develop their skills as they create their Minecraft City with code!

Take your time with these activities and coding experiences as they are critical for building out a strong computer science foundation for our youngest learners.

Curriculum Summary

Computing with Minecraft is specially designed computer science content for students in Grades K-2/Lower Primary/Ages 5-7. Students will learn, practice, and apply relevant computer science skills and concepts in both unplugged and digital experiences. The lessons are designed to provide students with an opportunity to build knowledge of the coding concept(s) in an unplugged version (i.e., demonstrate the coding concept on paper), practice the concept in the Minecraft world by manually completing the task, and then finally complete the coding task to grasp how coding can support efficiency and creativity.

Lesson Design

Each EDU guide contains multiple activities that are intended to be taught over the duration of several days; however, you should use discretion and modify/adapt the lesson activities based on your students’ needs and abilities. Within the lessons, the instructional sequence will contain three parts correlating with the gradual release model:

Direct Instruction—Teacher-Directed, “I Do”	In the first step, the teacher introduces and models the appropriate way of performing the skills included in the new concept being taught.
Guided Instruction— Teacher Modeling, “We Do”	After the teacher models the correct way to understand or perform the new concept being taught, teacher will guide the students as they work through some examples together.
Independent Practice—Teacher Support, “You Do”	This step is where students demonstrate their initial level of understanding of the new concept being taught through independent practice.

Instructional Materials

Curriculum Overview	That is this document you are reading now! This will provide you with insight about the curriculum and what is taught within the curriculum.
Educator’s Guides (EDU Guides)	An educator’s guide is provided for each of the lessons. The guide provides a high-level overview of the lesson, learning goals, standards addressed, required preparation for the activities, the lesson plans for the activities, and any additional materials needed.
Classroom Presentations	Each unit is supported by its own PowerPoint presentation to provide structure and guide the educator through the activities for the lesson.

Formative Assessments	After each lesson in the EDU guide, there is an opportunity to check for student understanding of the concept taught within the lesson. These formative assessments are typically comprised of 2-4 questions directly related to the learning that just took place.
Summative Assessment	At the end of the entire lesson sequence, students will be provided with a performance-based task to demonstrate their new knowledge and skills learned throughout the computer science unit, Computing with Minecraft. This performance-based task can be assessed using the provided rubric.
Minecraft World Files	The specific world files needed to experience the instructional activities have been linked directly within the Educator’s Guides. For Computing with Minecraft, there is one comprehensive world file that will support all of the lessons within the EDU guides.

Introduction

Level: Lower Primary/Grades K-2/Ages 5-7

Essential Question: How can computational thinking help us to solve problems?

Overview

Computing with Minecraft will provide students with an entry point into coding and an overview of Minecraft: Education Edition and Code Builder. First, students will “meet” their Agent and learn how to move their Agent in different directions, use their inventory, and try out their first building challenge. Next, students will design and code a road network, buildings, and houses in their Minecraft City. Then, students will create a park with water features, paths, and flower. After that, students will create a zoo with an entrance, paths, and animal enclosures. Finally, students will code and animate a wind farm along with a lighting system to provide wind power for Minecraft City.

As an integral part of this unit, students will learn about having a coding mindset and how to use computational thinking to solve tasks and problems. Once they have familiarized themselves with their Agent, they will complete their coding adventure—planning and building out the rest of their Minecraft City!

Pacing: 21 hours (could be more if tutorials and/or supplemental lessons are included)

Materials

Hardware	<ul style="list-style-type: none">• The teacher will need a laptop or tablet with a projector for the plugged-in coding lessons.• Internet access will be required for a portion of the lessons and activities.• Each student will need a device to complete the plugged-in coding activities.
Software	<ul style="list-style-type: none">• Minecraft: Education Edition needs to be deployed on the devices utilized within these lessons. Use this link to find information about Deploying Minecraft: Education Edition.• The teacher and students will need the relevant MCworld files downloaded for each lesson.
Other Materials	<ul style="list-style-type: none">• M:EE Tutorials (if needed)• M:EE Supplemental Lessons (if needed)• Microsoft MakeCode References for Minecraft

Minecraft: Education Edition Teaching and Learning Framework

GOAL: Provide students with the necessary skills to become creative coders and content creators	
<p style="text-align: center;">Coding Mindset</p> <p>Although Minecraft: Education Edition provides rigorous and engaging academics, we also deliver a holistic education- providing instruction for the social, emotional, and physical needs for our students. We want to foster a distinctive set of attributes. These qualities prepare our students to make exceptional contributions both in school and outside of school.</p> <p style="text-align: center; color: #76b82a;"> CURIOUS COMMITTED COOPERATIVE CONSIDERATE CONFIDENT </p>	<p style="text-align: center;">Computational Thinking Skills</p> <p>Computational thinking provides a vital skill set in which students must possess in order to fulfill the industry’s needs in the jobs of tomorrow. Our ever-changing workforce creates a critical need for innovation. Our students need computational thinking skills not just to solve the problems within their educational journey, but to also meet the challenges of adapting to our constantly changing workforce.</p> <p style="text-align: center; color: #76b82a;"> DECOMPOSITION PATTERN RECOGNITION ABSTRACTION ALGORITHMS </p>
<p style="text-align: center;">Computer Science Units of Study</p> <p>Minecraft: Education Edition provides meaningful, relevant, and engaging units of study. The units of study will possess a conceptual lens to allow for depth and complexity to develop conceptual understanding—knowledge which transfers through time, across cultures, and across situations.</p> <p style="text-align: center; color: #76b82a;"> DIGITAL CITIZENSHIP PROGRAMMING CYBERSECURITY IMPACTS OF COMPUTING </p>	<p style="text-align: center;">Community</p> <p>As students gain and possess new knowledge and skills, we strive for them to find a greater purpose of “why do I need to know this” or more importantly, “how can I use this information?”. We aim to empower students, develop confidence and self-efficacy into a commitment to serve the community in which we live in and beyond.</p> <div style="text-align: center;"> <pre> graph TD Gather --> Analyze Analyze --> Create Create --> Share Share --> Reflect Reflect --> Gather </pre> </div>

This unit will focus on **Programming**, as students will learn all about how to use Code Builder in Minecraft. Students will explore the various MakeCode blocks they can use to create programs.

This unit will focus on the coding mindset of being **curious**, as they create, test, and debug various programs throughout their coding journey to build Minecraft City.

This unit will focus on **algorithms** as students create programs that utilize step-by-step instructions to solve problems, complete tasks, and be creative!

This unit will provide students with the opportunity to consider how they can utilize their new knowledge of artificial intelligence to bring awareness to the **community**. Students can explore how programming is presently being used in their communities and reflect on programming can enhance their communities or even solve a present problem within their community.

Instructional Sequence

This next section will provide you with an overview of the activities included in this lesson sequence. The lesson sequence is presented in chronological order—we suggest working in order, as the content will build upon skills presented in the previous session. A session is equivalent to one class period, or a 45-60 minute session. However, educators should feel empowered to modify and adapt the lesson sequence to best meet the needs of their students.

Lesson Sequence Overview

Day	Objectives	Teacher will	Students will	Resources
1*	<i>Students will learn about what is coding and the game play of Minecraft: Education Edition</i>	<i>Introduce Minecraft: Education Edition to students and demonstrate how to use Code Builder with the Agent</i>	<i>Students will learn how to utilize Code Builder to program their Agent</i>	<i>The Agency: Get Familiar with Your Agent</i>
2*	<i>Students will learn how to use the word blocks to navigate the Agent in Minecraft: Education Edition.</i>	<i>Lead the Code Your Agent to Move activity</i>	<i>Students will program the Agent to destroy tree and collect wood to be placed in their inventory</i>	<i>The Agency: Code Your Agent to Move and Destroy</i>
3*	<i>Students will demonstrate the effective use of Player and Agent blocks.</i>	<i>Explain the concept of variables and how they can be used in coding to be more efficient</i>	<i>Students will practice interacting with the Agent's inventory (add, remove, move items) and then code the Agent to place and build structures</i>	<i>The Agency: Code Your Agent to Build</i>
4*	<i>Students will create multiple lines of code with an intended outcome.</i>	<i>Introduce unit summary and lead the coding activity</i>	<i>Students will learn how creating structures with code can take seconds that would normally take hours to build (if done manually)</i>	<i>City Planner: Code a Road Network</i>
5*	<i>Students will be introduced to the concepts of loops.</i>	<i>Model and explain what are loops and the purpose of loops in coding</i>	<i>Students will program the Agent to build a house using functions in their code</i>	<i>City Planner: Code a Building</i>
6*	<i>Students will be introduced to a new and helpful coding tool called "clone".</i>	<i>Explain and model how to use the code tool, "clone" to copy and repeat structures.</i>	<i>Students will design a small house and then use the clone tool to code a row of houses.</i>	<i>City Planner: Code a Row of Houses</i>

7*	<i>Students will describe how loops affect code and in-game actions.</i>	<i>Explain how loops can help you could more efficiently</i>	<i>Students will use loops to code a park fence.</i>	<i>Parks and Recreation: Code a Park Fence</i>
8*	<i>Students will code and test programs to ensure they run as intended.</i>	<i>Support student through the building and testing process</i>	<i>Students will build and test a program to build a water feature in Minecraft City</i>	<i>Parks and Recreation: Code a Water Feature</i>
9*	<i>Students will be introduced to the concept of variables.</i>	<i>Introduce students to the math blocks in Code Builder to use in their code.</i>	<i>Students will merge math with code to create calculated in-game actions</i>	<i>Parks and Recreation: Plant Some Flowers</i>
10*	<i>Students will review the loops drawer and how they can help with certain tasks.</i>	<i>Introduce the coding activity of creating a zoo entrance with a gate and a sign</i>	<i>Students will create a program using loops to build a gate for the zoo</i>	<i>A Zoo: Code a Zoo Entrance</i>
11*	<i>Students will utilize positions blocks to manipulate the Minecraft world through coordinates.</i>	<i>Explain the difference between absolute world position and relative player position</i>	<i>Students will create color-coded zoo path by coding with coordinates to complete the task.</i>	<i>A Zoo: Code a Zoo Path</i>
12*	<i>Students will be able to explain what a builder function is and utilize the builder function in a program.</i>	<i>Introduce the builder functions and how to utilize these coding blocks to build things much faster than the player or the Agent</i>	<i>Students will code a wall using the builder functions to create animal enclosures for the zoo.</i>	<i>A Zoo: Code Animal Enclosures</i>
13*	<i>Students will use code to design a moving system.</i>	<i>Review the concept of the clone command and how it's beneficial for coding.</i>	<i>Students will build a wind turbine by selecting a method of their choice and then clone the wind turbines.</i>	<i>Wind Power: Code a Wind Farm</i>
14*	<i>Students will input, test, and adjust cloning coordinates to animate their wind farm.</i>	<i>Model and guide students through the six phases to animate the wind farm.</i>	<i>Students will animate the wind farm to simulate the movement of the wind blades generating power when the wind turns them.</i>	<i>Wind Power: Animate the Wind Farm</i>
15*	<i>Students will create a model for wind power.</i>	<i>Model and guide students through the six phases to code a lighting system.</i>	<i>Students will code wiring the redstone, code a wind indicator, and code to generate power to create a model of wind power.</i>	<i>Wind Power: Code a Lighting System</i>

16- 20	<i>Students will collaborate with peers to create a computational artifact.</i>	<i>Explain the summative task and provide success criteria and guidance to collaborative groups.</i>	<i>Students will work together to plan, design, test, and present their Minecraft City Improvement.</i>	<i>Computing with Minecraft: Minecraft City Improvement</i>
-----------	---	--	---	---

Computer Science Concepts: Computing with Minecraft

Lesson	Concept(s)
The Agency Get Familiar with Your Agent	Sequencing
The Agency Code Your Agent to Move and Destroy	Sequencing
The Agency Code Your Agent to Build	Variables
City Planner Code a Road Network	Coordinates/Positions
City Planner Code a Building	Loops
City Planner Code a Row of Houses	Loops
Parks & Recreation Code a Park Fence	Sequencing Loops
Parks & Recreation Code a Water Feature	Sequencing Loops
Parks & Recreation Plant Some Flowers	Variables
A Zoo Code a Zoo Entrance	Loops
A Zoo Code Zoo Paths	Functions
A Zoo Code Animal Enclosures	Functions
Wind Power Code a Wind Farm	Functions
Wind Power Animate the Wind Farm	Abstraction
Wind Power Code a Lighting System	Abstraction
Summative Task Improve Minecraft City Performance Task	All concepts from the Computing with Minecraft lessons

MINECRAFT VISUAL GLOSSARY

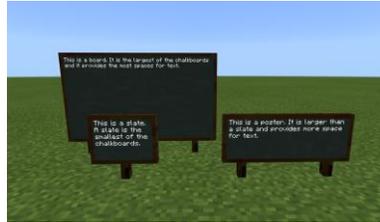
Agent

personal robot in Minecraft



Chalkboards

special blocks that allow you to write and display text in M:EE worlds



Chest

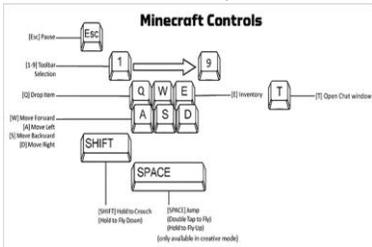
a place to store items outside of your inventory



Controls

(keyboard)

keyboard buttons that help you move around and complete tasks



Controls

(touch)

the touch pad that helps you move around and complete tasks



Hotbar

selection bar that appears on the bottom of the screen



Inventory

pop-up menu the player (or Agent) uses to manage the items they carry



Minecraft: Education Edition

a game-based learning platform



Mob

game character resembling a living creature



NPC

non-player character



Redstone

mined from redstone ore; its dust is used to power circuits and machinery in Minecraft



Spawn Point

the location where a player begins game play



COMPUTER SCIENCE GLOSSARY

Absolute World

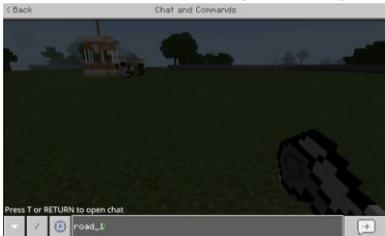
Position

position or coordinate (X, Y, Z) that represents the distance from the Minecraft world origin of (0, 0, 0)



Chat Window

field to enter commands; can be displayed by pressing the T key (default), enter key, or / key



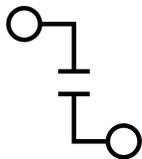
Code Builder

coding editor used to create programs



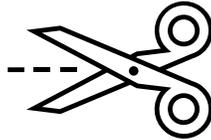
Decompose

breaking down a problem or task into smaller, more manageable parts



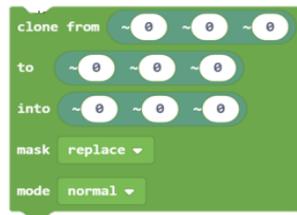
Abstraction

process that instructs us to remove non-relevant details and patterns that will not help us solve the problem



Clone

the clone command copies the blocks in a certain area and places them in another area



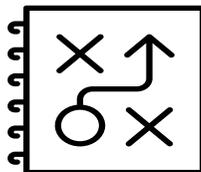
Coordinates

a coordinate represents a position or location



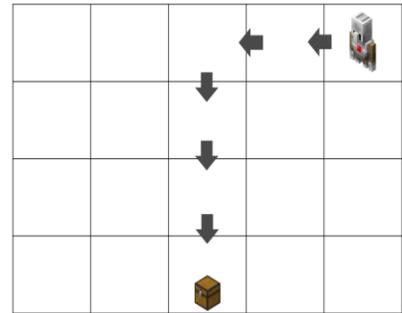
Function

a block of organized, reusable code that is used to perform a single, related action



Algorithm

a sequence of defined steps



Code (Coding)

the method of giving a computer instructions to perform a specific task



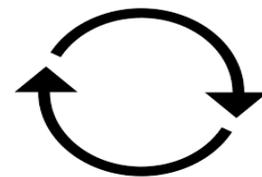
Debug

the process of finding (and correcting) errors in a computer program



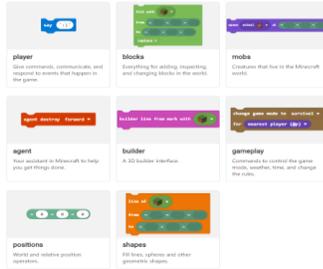
Loop

sequence that repeats a portion of code a set number of times until the desired task is complete



MakeCode Blocks

blocks used to create code



Sequence

a programmed (coded) algorithm



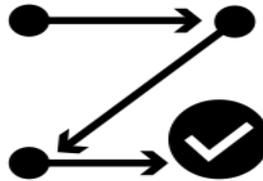
Relative World Position

A position or coordinate ($\sim X, \sim Y, \sim Z$) that represents the distance from the player's current position as the origin of ($\sim 0, \sim 0, \sim 0$)



Test

the process of running a program to see if it is correct



Search Feature

allows you to search for specific items in the inventory by typing in the keyword



Variable

names you give to computer memory locations which are used to store values in a computer program

