HOUR OF CODE 2022:
Escape Estate

EDUCATOR GUIDE

This Educator Guide includes:

- Introduction
- Hour of Code 2022: Theme Overview
- Learning Goals
- Lesson Overview
- Learning Experiences for Students
- Lesson Activities
- Educational Standards
- Visual Glossary

TABLE OF CONTENTS

Introduction

Are you ready to put your computational thinking skills to the test? Welcome to the Hour of Code 2022: Escape Estate!

This Educator Guide is designed to provide you with all the information you need to help your students have an enjoyable and successful Hour of Code.

Hour of Code 2022: Escape Estate has been designed as a fun and creative tutorial introduction to computer science (CS). This one-hour experience will provide your students with the basics of coding (blocks or Python) and demonstrate all the ways they can utilize computational thinking to solve problems.

This year's Hour of Code 2022 also offers additional opportunities to replay and learn more coding skills and connect to other areas.

ALL students can participate in the Hour of Code 2022! Let's go!

How to Get Started

- Install Minecraft: Education Edition by visiting aka.ms/HourofCode2022
- Use this Educator Guide!
- Watch the introductory video to this year's Hour of Code theme.
- Watch the walkthrough video to get a better understanding of the lesson and how to navigate through the Minecraft world.
- Download a copy of the coding solutions.
- Do you have additional questions? Check out the FAQ.

Hour of Code 2022—Theme Overview

How can we solve problems?

Through computational thinking, of course!

Computational thinking is NOT thinking like a computer. In fact, it is much more robust and all-encompassing. Computational thinking is a problem-solving approach consisting of specific skills and practices: decomposition, pattern recognition, abstraction, and algorithms. While these skills are used in computer science, these skills can also be applied in a variety of other disciplines such as mathematics, visual arts, reading, science, and more.

To provide a fun and engaging experience to practice computational thinking, Minecraft: Education Edition to proud to present this year's Hour of Code: Escape Estate. Hour of Code 2022: Escape Estate is a quest-based, coding adventure set as series of enchanted escape rooms, featuring engaging and imaginative storylines connected to computational thinking skills. Each of these escape rooms will require students to search and collect clues, solve coding puzzles, and successfully unlock the doors to lead to the ultimate escape out of the mansion!

Students will feel the excitement and curiosity of the escape room concept in this brand-new digital Minecraft world. Players will select their route and activity, which augments their decision-making process by encouraging them to factor in clues found in the world. With the different pathways, students can play through multiple times to discover a new coding experience!

Just remember—behind every door is a new adventure. Are you up for the challenge? If so, use your computational thinking skills and see if you can make it out before dawn! Perhaps you will be able to claim the one million emerald prize…

Learning Goals

Through the Hour of Code 2022: Escape Estate, students will:

- Explore early coding concepts
- Use computational thinking to solve coding puzzles
- Create code (using blocks or Python) to solve problems
- Discover how computational thinking can be applied to other subject areas (outside of computer science)

This year’s Hour of Code has a strong emphasis and focused learning outcomes on computational thinking. Computational thinking is an approach to problem solving using specific skills and practices.

Decomposition is breaking down a problem or task into smaller, more manageable parts.	Pattern Recognition is the act of finding and understanding similarities in ideas and objects.	Abstraction is the process of simplifying– filtering out extra details about characteristics or patterns we don't need so we can focus on what’s essential to solve the problem.	Algorithms are a set of step-by-step instructions (represented as blocks or lines of code) to solve a problem or complete a task.

Lesson Overview

Beyond every door is another adventure…

What is Escape Estate?

Escape Estate is a unique and exciting Minecraft map that reflects a digital escape room concept. Players will have the opportunity to accept the Escape Estate challenge—if they successfully escape the estate before dawn, there is a prize of one million emeralds!

Escape Estate Game Play

After players complete a Minecraft navigation tutorial, the journey will start right outside of the estate, where players meet Dr. Dodge Breakowski. Dr. Breakowski will introduce himself, explain the objective of the game, and ask players to approach the mansion.

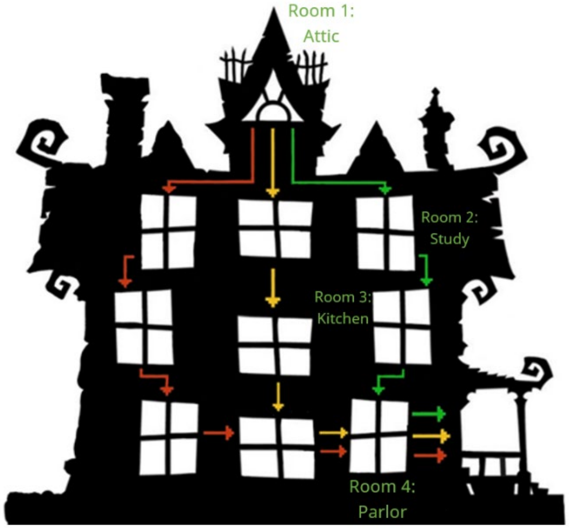
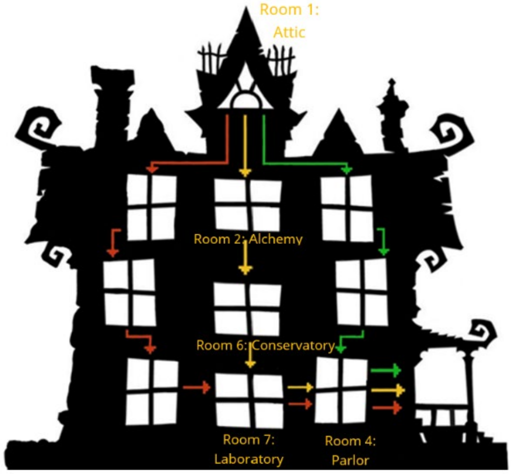
Uh oh! While approaching the front door of the mansion, players will be pulled into the house, and they will find themselves in the attic. At this point, players will complete critical tasks through interactive in-game activities:

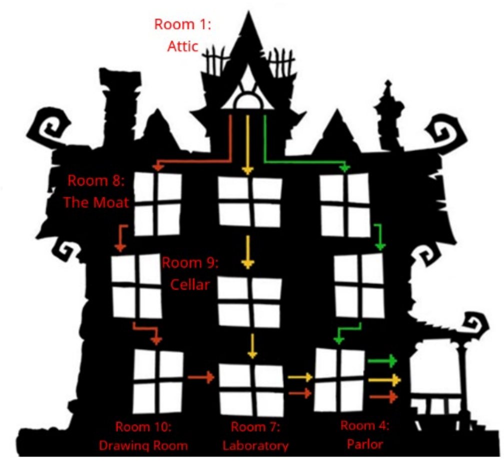
- Locate the Lost Diary: Players will be led to a lost diary. The diary will be placed in their Hotbar. Within the diary, they will also find their first diary entry. The first diary entry is about the attic. Players will also be responsible for trying to find additional lost diary entries throughout game play. There is a total of 10 diary pages.
- Rescue the Agent: Right after finding the lost diary, players will be instructed to investigate a noise coming from the floor! They will discover and meet the Agent. The Agent will need their assistance—players will need to rescue the Agent from beneath the trapdoor. As a reward for helping the Agent, the Agent will unlock the door to open the green pathway. An additional coding activity is required to unlock the yellow pathway; and then another coding activity is required to unlock the red pathway.
- Select their Pathway: Once the doors have been unlocked, players will have the option to select one pathway to help them escape from the mansion. All pathways will have unique rooms; certain rooms will only be

seen/experienced if players select that specific pathway. However, this mystery is part of fun and will provide novelty and engagement for students! Also, they can always return to the game to try the other pathways too!

Pathways within Escape Estate

All players will start in the Attic (Room 1) and end in the Parlour (Room 4).

Green Pathway* (4 Rooms) *This pathway is recommended for younger students and/or students who little to no prior coding experience. Students can complete other paths at any time.	 Room 1: Attic Room 2: Study Room 3: Kitchen Room 4: Parlor The green pathway will have 4 specific escape rooms to complete within the mansion: - Room 1: Attic - Room 2: Study - Room 3: Kitchen - Room 4: Parlour
Yellow Pathway (5 Rooms)	 Room 1: Attic Room 2: Alchemy Room 6: Conservatory Room 7: Laboratory Room 4: Parlor

	The yellow pathway will have 5 specific escape rooms to complete within the mansion: - Room 1: Attic - Room 5: The Alchemy Room - Room 6: Conservatory - Room 7: Laboratory - Room 4: Parlour
Red Pathway (6 Rooms)	 Room 1: Attic Room 8: The Moat Room 9: Cellar Room 10: Drawing Room Room 7: Laboratory Room 4: Parlor The red pathway will have 6 specific escape rooms to complete within the mansion: - Room 1: Attic - Room 8: The Moat - Room 9: Cellar - Room 10: The Drawing Room - Room 7: Laboratory - Room 4: Parlour

Escaping from the Estate

After completing the last coding puzzle in the Parlor (Room 4), the ending sequence will begin. In a sudden turn of events, players will find out the mystery behind the Escape Estate, or should we say who?!

Players will need to complete an additional four coding puzzles in the side rooms to prepare for the epic boss battle to take down the villain (or even villains…).

Finally, players will complete two additional coding puzzles to officially take out the villain, save the day, and…

EMERALDS.

Don’t forget about the one million emeralds!

Learning Experiences for Students

To increase student access to the Hour of Code, there are three different learning experiences for students: in-class with a teacher facilitator, in-class as a self-guided experience, or in a remote (virtual) learning capacity. Each of the learning experiences have different levels of teacher support and modification for student success and participation in the Hour of Code.

Teacher-Facilitating (Face to face)

This type of learning experience is going to provide students with the highest level of teacher support. As the teacher, you will be providing explicit instruction and modelling before releasing students into independently working through the coding activities in the Hour of Code. You will utilize the Hour of Code Presentation to guide you and your students through beginning game play, coding activities, and lesson reflection.

Student Self-Guided

This type of learning experience is going to provide students with the least amount of teacher support. As the teacher, you will make sure your students are able to log into the Minecraft: Education Edition platform. You will provide students with the Hour of Code Presentation to assist them with their self-paced and self-guided experience for game play. The presentation will provide students with explicit directions on how to participate in the Hour of Code. You should also provide students with a copy of the Visual Glossary, which is located at the end of this Educator's Guide. It is recommended that you provide these materials to students in a digital medium of your choice or print off paper copies, so they are readily available for student use.

Remote Learning Experience

This type of learning experience can be facilitated synchronously or asynchronously.

If you are having students participate in a synchronous (live) virtual session, you will facilitate the session just like the in-class experience (with teacher facilitation). As the teacher, you will be providing explicit instruction and modelling before releasing students into independently working through the coding activities in the Hour of Code. You will utilize the Hour of Code Presentation to guide you and your students through beginning game play, coding activities, and lesson reflection. Make sure your virtual communication platform (like Microsoft Teams) is ready to share content and the sound is enabled.

If you are having students participate in an asynchronous (i.e., not live) virtual session, you will facilitate the session just like the self-guided in-class experience. As the teacher, you will make sure your students have access to the Minecraft: Education Edition platform prior to lesson. You will also need to provide students with the Hour of Code Presentation to assist them with their self-paced and self-guided experience for game play. The presentation will provide students with explicit directions on how to participate in the Hour of Code. You should also provide students with a copy of the Visual Glossary, which is located at the end of this Educator's Guide. It is recommended that you provide these materials (presentation and glossary) to students in a digital medium of your choice (examples: Microsoft Teams channel, OneNote notebook, your learning management system, etc.).

If you plan to offer this lesson via remote learning, please consider these tips:

1. This lesson is not designed as a multiplayer experience. Each individual student should work in their own version of the world.
2. Divide students into pairs or small groups via breakout rooms so they can help one another troubleshoot as they solve the coding activities.
3. Given that students are likely to have varying levels of familiarity with Minecraft and coding, it may help to assign student group leaders to help peers who may get stuck.

For more information on remote learning in Minecraft: Education Edition, please refer to https://aka.ms/remote-learning-kit

Lesson Activities

This next section will provide you with an overview of exactly how to facilitate the lesson activities for the Hour of Code.

There will be three different parts:

- Direct Instruction (I Do)
- Guided Instruction (We Do)
- Independent Work (You Do)

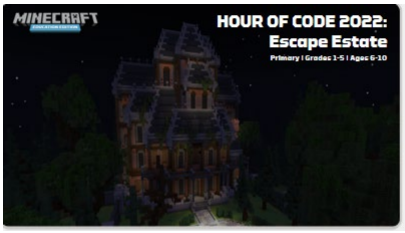
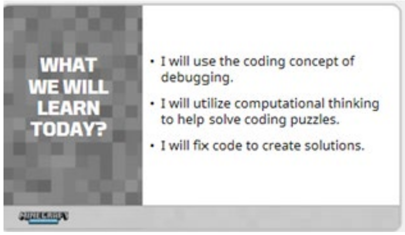
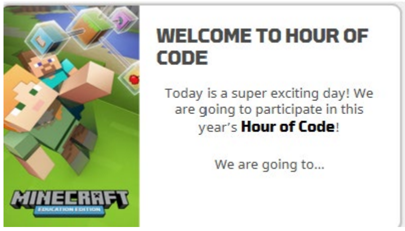
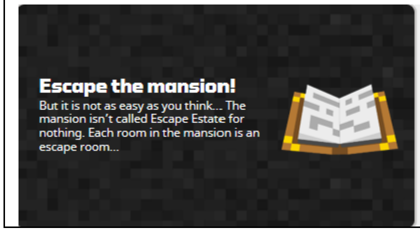
This structure is an intentional lesson design following the gradual release model to ensure student success. Both the Direct Instruction and Guided Instruction parts will utilize the Hour of Code Presentations. This year, there are two options for classroom presentations that should be used with the appropriate age groups:

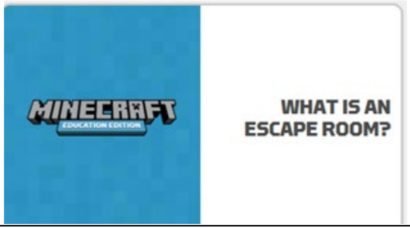
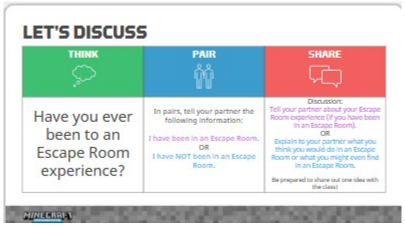
- Primary | Grades 1-5 | Ages 6-10
- Secondary | Grades 6-12 | Ages 11-18

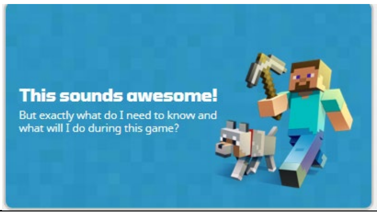
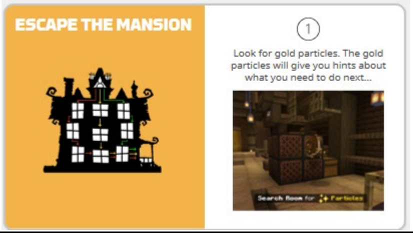
The annotated notes found in this document will directly correlate with the slides within the HOC presentations. You should use them together.

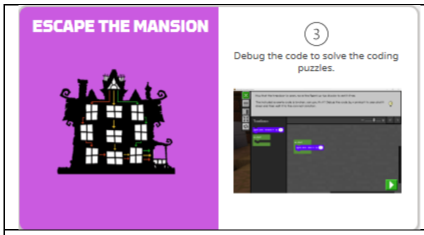
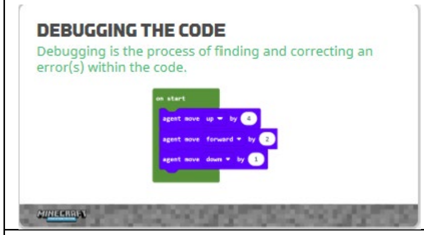
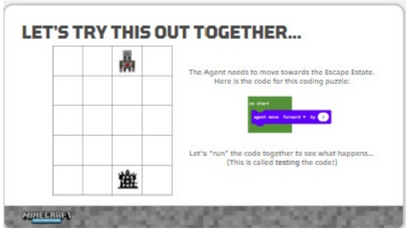
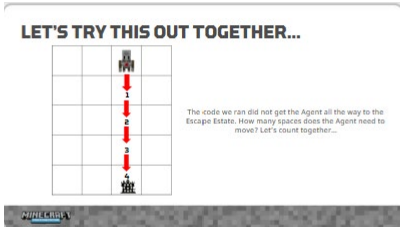
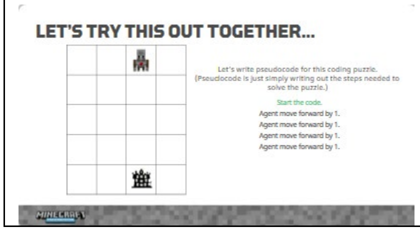
Primary | Grades 1-5 | Ages 6-10

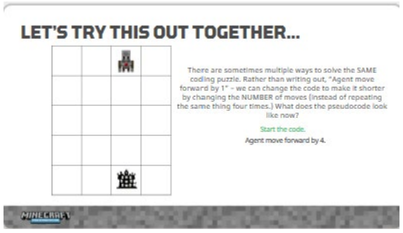
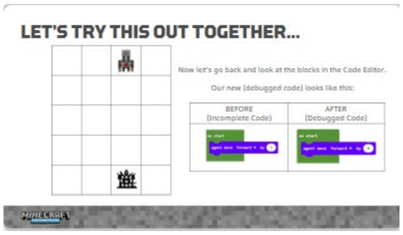
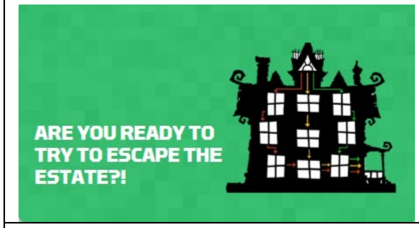
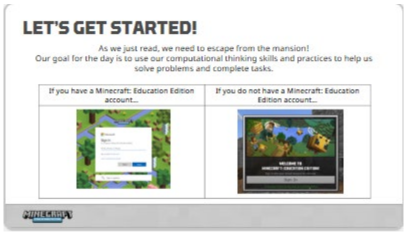
Direct Instruction (I Do…) (teacher led) – PRIMARY |Grades 1-5 | Ages 6-10

Slide 1  HOUR OF CODE 2022: Escape Estate Primary I Grades 1-5 I Ages 6-10	At the time, you should have this presentation on display. The next several slides are intended to set the stage for the HOC game play experience. Students are not logged into the platform yet. However, you should provide the visual glossary (provided at the end of this guide) for reference. **Say**: *"Welcome to Hour of Code 2022! Hour of Code is a global movement that supports computer science and coding for all students. For our Hour of Code, we will be using Minecraft: Education Edition. This year's Hour of Code experience is called Escape Estate. Before we start playing the game, we are going to review some very important information that you will need to know and understand during game play."*
Slide 2  WHAT WE WILL LEARN TODAY? • I will use the coding concept of debugging. • I will utilize computational thinking to help solve coding puzzles. • I will fix code to create solutions.	Review the "I will…" statements with students. (Read slide) **Say**: *These are our learning goals for the day. If we are successful during Escape Estate, we will be able to demonstrate our understanding of each of these statements.*
Slide 3  WELCOME TO HOUR OF CODE Today is a super exciting day! We are going to participate in this year's **Hour of Code**! We are going to…	**Say**: "*Today, you are going to participate in the Hour of Code! We are going to…"*
Slide 4  **Escape the mansion!** But it is not as easy as you think… The mansion isn't called Escape Estate for nothing. Each room in the mansion is an escape room…	**Say:** *"Escape the mansion! BUT… it's not as easy as you think… The mansion isn't called Escape Estate for nothing. Each room in the mansion is an escape room."*

Slide 5  WHAT IS AN ESCAPE ROOM?	**Say:** *"What is an Escape Room?"*
Slide 6  LET'S DISCUSS THINK: Have you ever been to an Escape Room experience? PAIR: In pairs, tell your partner the following information: I have been in an Escape Room. OR I have NOT been in an Escape Room. SHARE: Discussion: Tell your partner about your Escape Room experience (if you have been in an Escape Room). OR Explain to your partner what you think you would do in an Escape Room or what you might even find in an Escape Room. Be prepared to share out one idea with the class!	Facilitate a Think-Pair-Share activity with students. **Step 1: THINK** Begin with the question, "Have you ever been to an Escape Room experience?" Give students time to individually think about an answer and document their responses on their own, either written or in pictures. Students can be given about 30 seconds to 1 minute for this part of the exercise. **Step 2: PAIR** Pair up the students in your classroom. Ask the students to share their responses to this question with their partners. **Step 3: SHARE** Allow the pairs to elaborate on their response. Then, for the next part, come back together as a class and have a whole class discussion. You can either choose to have one person from each pair share with the class, or the discussion can be more open. Students can also share with the class what their partner said.
Slide 7 WHAT IS AN ESCAPE ROOM? An escape room is a game where players must uncover clues, solve puzzles, and complete tasks in one or more rooms with the objective of completing a specified goal in a period of time. The main goal is simple… ESCAPE!	**Say:** *"An escape room is a game where player must uncover clues, solve puzzles, and complete tasks in one or more rooms. The goal is simple—you want to escape the room!*
Slide 8 WELCOME TO HOUR OF CODE 2022 ESCAPE ESTATE Select the image to view video	Press the play button on the video—this video clip (trailer) will provide students with a preview of what they will be doing in Hour of Code 2022.
Slide 9	**Say:** *"Our experience is going to be a little bit different! We are going to go into a single-player, digital journey to escape a mansion in Minecraft!"*

 ESCAPE ROOMS IN MINECRAFT? Our experience is going to be a little bit different ☺ We are going to go into a single-player, digital journey to escape a mansion in Minecraft!	
Slide 10 GOAL FOR THE DAY Today, you will be playing Escape Estate- a super exciting and fun escape room challenge in Minecraft! You will use the coding concept of debugging to solve coding puzzles to help you escape the mansion!	**Say:** *"Today, you will be playing Escape Estate- a super exciting and fun escape room challenge in Minecraft! You will use the coding concept of debugging to solve coding puzzles to help you escape the mansion!"*
Slide 11  This sounds awesome! But exactly what do I need to know and what will I do during this game?	**Say:** *"Wow! This sounds awesome! But exactly what do I need to know and what will I do during this game?"*
Slide 12 DON'T WORRY! There are 3 things you will do during Hour of Code to help you successfully escape the mansion!	**Say:** *"Don't worry! There are 3 things you will do during the Hour of Code to help you successfully escape the mansion."*
Slide 13  ESCAPE THE MANSION 1 Look for gold particles. The gold particles will give you hints about what you need to do next... Search Room for ✦ Particles	**Say:** *"The first important thing you will need to do is look for the gold particles. The gold particles will give you hints about what to look for or what to do next..."*
Slide 14 ESCAPE THE MANSION 2 Look for the secret diary pages.	**Say:** *"Look for the secret diary pages hidden in the rooms."*
Slide 15	**Say:** *"Debug the code to solve the coding puzzles."*

 ESCAPE THE MANSION 3 Debug the code to solve the coding puzzles.	
Slide 16 WAIT! WHAT DOES THAT MEAN?!	**Say:** *"WAIT! What does that mean?!"*
Slide 17  DEBUGGING THE CODE Debugging is the process of finding and correcting an error(s) within the code. on start agent move up by 4 agent move forward by 2 agent move down by 1	**Say:** *"Debugging is the process of finding and correcting an error(s) within the code."*
Slide 18*  LET'S TRY THIS OUT TOGETHER... The Agent needs to move towards the Escape Estate. Here is the code for this coding puzzle: on start agent move forward by 2 Let's "run" the code together to see what happens... (This is called testing the code!) *This slide has animations; you will need to advance the slide for the content to be displayed.	**Say:** *"The Agent needs to move towards the Escape Estate. Here is the code for this coding puzzle."* **Do:** Point to the code on the slide. **Say:** *"Let's 'run' the code together to see what happens. {This is called testing the code.}"* **Do**: Advance the slide to show the animation to "run" the code.
Slide 19  LET'S TRY THIS OUT TOGETHER... The code we ran did not get the Agent all the way to the Escape Estate. How many spaces does the Agent need to move? Let's count together... *This slide has animations; you will need to advance the slide for the content to be displayed.	**Say:** *"The code we ran did not get the Agent all the way to the Escape Estate. How many spaces does the Agent need to move? Let's count together…"* **Do:** Advance the slide to show the arrows {indicates a move forward} and the number to count the block moved.
Slide 20  LET'S TRY THIS OUT TOGETHER... Let's write pseudocode for this coding puzzle. (Pseudocode is just simply writing out the steps needed to solve the puzzle.) Start the code. Agent move forward by 1. Agent move forward by 1. Agent move forward by 1. Agent move forward by 1.	**Say:** *"Awesome job! The Agent made it to the Escape Estate. Now, let's write out pseudocode for this coding puzzle. {Pseudocode is just simply writing out the steps needed to solve the puzzle}"* **Do:** Advance the slide to show each line of pseudocode.

*This slide has animations; you will need to advance the slide for the content to be displayed.	
Slide 21  LET'S TRY THIS OUT TOGETHER... There are sometimes multiple ways to solve the SAME coding puzzle. Rather than writing out, "Agent move forward by 1" – we can change the code to make it shorter by changing the NUMBER of moves (instead of repeating the same thing four times.) What does the pseudocode look like now? Start the code. Agent move forward by 4. *This slide has animations; you will need to advance the slide for the content to be displayed.	**Say:** *"There are sometimes multiple ways to solve the SAME coding puzzle. Rather than writing out, "Agent move forward by 1" – we can change the code to make it shorter by changing the NUMBER of moves (instead of repeating the same thing four times.) What does the pseudocode look like now?"* **Do:** Advance the slide to show each line of pseudocode.
Slide 22  LET'S TRY THIS OUT TOGETHER... Now let's go back and look at the blocks in the Code Editor. Our new (debugged code) looks like this: BEFORE (Incomplete Code) | AFTER (Debugged Code) *This slide has animations; you will need to advance the slide for the content to be displayed.	**Say:** *"Now, let's go back and look at the blocks that will appear in the Code Editor. Our new (debugged) code looks like this."* **Do:** Point out the code on the slide. Then advance the slide to run the code again.
Slide 23  ARE YOU READY TO TRY TO ESCAPE THE ESTATE?!	**Say:** *"Are you ready to try to escape the estate?!"*
Slide 24  LET'S GET STARTED! As we just read, we need to escape from the mansion! Our goal for the day is to use our computational thinking skills and practices to help us solve problems and complete tasks. If you have a Minecraft: Education Edition account... | If you do not have a Minecraft: Education Edition account...	**Do:** Help all students to log into the Minecraft: Education Edition platform. After logging in, they should launch the HOC 2022 world. Explain to students that they should NOT start playing- just simply launch the platform. **NOTE**: This is a single-player game. Although they will all be seeing the same world, each person will be playing their own adventure.

Guided Instruction (We Do…) (Class activity) – PRIMARY |Grades 1-5 | Ages 6-10

Tutorial Sequence Navigation in Minecraft (Slides 25-33)	Students will select their mode of play based on the device they are using (i.e., keyboard or touch). The player will then start the tutorial by completing a series of mini tasks that will help them to get familiar with how to move and utilize different elements in Minecraft.

	Students will press the button to begin the map. Players will meet Dr. Dodge Breakowski. He will introduce himself, extend the prize of one million emeralds, and then ask the players to move towards the mansion. They will also understand how to use the Immersive Reader feature, as reading the dialog in this Minecraft map is critical for understanding.
Start the Minecraft Map Room 1: The Attic (Slides 34-42)	The mansion will pull the player in deposit them into Room 1: The Attic. Players will locate the lost diary and read the attic entry. Then, players will need to investigate the noise! Upon their investigation, players will find the Agent under a trapdoor. Players will now need to select their programming language: blocks or Python. NOTE: We are telling students to select BLOCKS. Once you select a programming language, you cannot change it, or you must restart the game.
Blocks: Coding Demonstration Room 1: The Attic (Slides 43-52)	They will code a solution to rescue the Agent. Upon rescuing the Agent, the player will be rewarded with the door to the green pathway be unlocked for them. NOTE: In this experience, we are telling the player to select the **GREEN pathway**.

There are two options on how to continue Hour of Code 2022: Escape Estate.

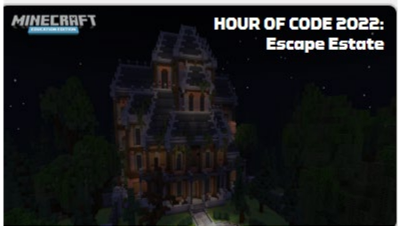
- **Option 1**: You can continue to work through the world as a guided experience (i.e., complete the game as a whole group experience; model/demonstrate each coding puzzle as needed). This option is recommended for younger students, who may need additional support in fully understanding the coding concepts. There are also additional slides (slides 53-79) to provide additional support for the green pathway. These slides will provide additional hints for students!
- **Option 2**: You can release students to work independently on completing their selected pathway. Once they have finished completing their pathway, you can regroup as a class to discuss the experience, address the reflection questions, and consider ways to extend the learning.

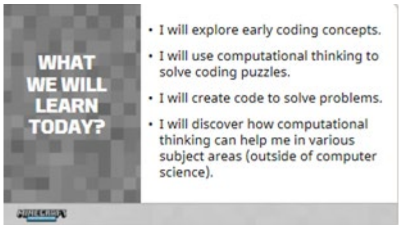
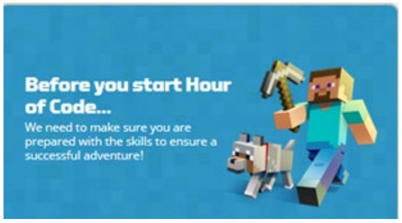
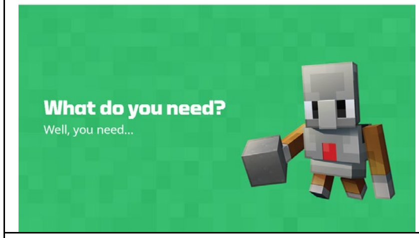
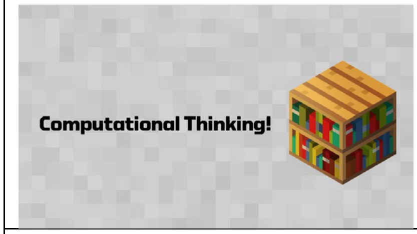
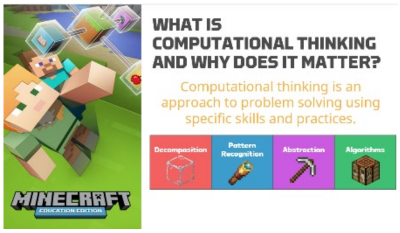
Independent Work (You Do…)

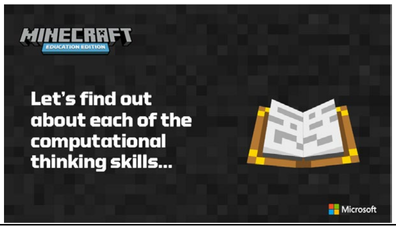
At this point, you are going to allow students to continue through their pathway through the Escape Estate.

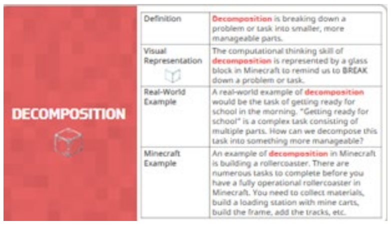
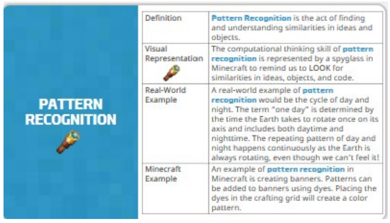
Secondary | Grades 6-12 | Ages 11-18

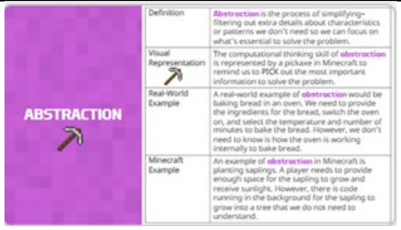
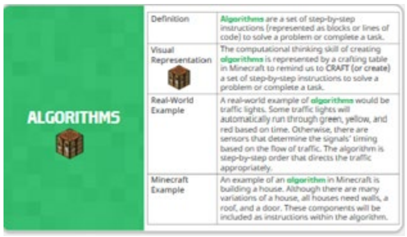
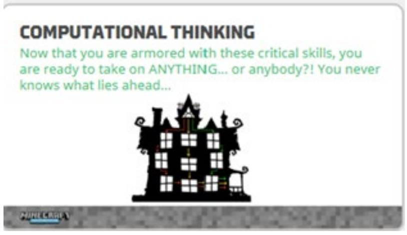
Direct Instruction (I Do…) (teacher led) – SECONDARY | Grades 6-12 | Ages 11-18

Slide 1  MINECRAFT HOUR OF CODE 2022: Escape Estate	At the time, you should have this presentation on display. The next several slides are intended to set the stage for the HOC game play experience. Students are not logged into the platform yet. However, you should provide the visual glossary (provided at the end of this guide) for reference. **Say**: *"Welcome to Hour of Code 2022! Hour of Code is a global movement that supports computer*

	science and coding for all students. For our Hour of Code, we will be using Minecraft: Education Edition. This year's Hour of Code experience is called Escape Estate. Before we start playing the game, we are going to review some very important information that you will need to know and understand during game play."
Slide 2  WHAT WE WILL LEARN TODAY? • I will explore early coding concepts. • I will use computational thinking to solve coding puzzles. • I will create code to solve problems. • I will discover how computational thinking can help me in various subject areas (outside of computer science).	Review the "I will…" statements with students. (Read slide) **Say**: *These are our learning goals for the day. If we are successful during Escape Estate, we will be able to demonstrate our understanding of each of these statements.*
Slide 3  Before you start Hour of Code… We need to make sure you are prepared with the skills to ensure a successful adventure!	**Say**: "*You are going to have a fantastic time during Hour of Code! However, I want to make sure you are fully prepared before you start playing the game."*
Slide 4  What do you need? Well, you need…	**Say:** "What do you need? Well, you need…"
Slide 5  Computational Thinking!	**Say:** "Computational thinking!"
Slide 6*  WHAT IS COMPUTATIONAL THINKING AND WHY DOES IT MATTER? Computational thinking is an approach to problem solving using specific skills and practices. Decomposition | Pattern Recognition | Abstraction | Algorithms	Ask the question to students: **Say**: "*What is computational thinking?"* Allow for students' responses. (Answers will vary) Advance the slide to reveal the definition: **Say:** *"Computational thinking is an approach to problem solving using specific skills and practices."*

<table>
<tr>
<td>*This slide has animations; you will need to advance the slide for the content to be displayed.</td>
<td>Advance the slide to reveal the computational thinking domains.
<b>Say:</b> <i>"There are 4 parts to computational thinking: decomposition, pattern recognition, abstraction, and algorithms. These skills will guide you through the problem-solving process to help to solve a problem or complete a task. Although there are 4 parts to computational thinking, you may not use all of them for every coding solution. Some of these skills may be more useful and/or relevant in certain coding puzzles."</i>
<i>Computational thinking is not just helpful for computer science! Computational thinking can help us solve problems and become critical thinkers and innovators. We use computational thinking all the time in all kinds of way, even though you may not realize it!</i></td>
</tr>
<tr>
<td>Slide 7


Let's find out about each of the computational thinking skills...
</td>
<td><b>Say</b>: <i>"We are going to explore each of the skills within computational thinking. We will review the formal definition, the visual representation, a real-world example, and an example within Minecraft."</i></td>
</tr>
<tr>
<td>Slide 8

DECOMPOSITION
<table>
<tr><td>Definition</td><td>Decomposition is breaking down a problem or task into smaller, more manageable parts.</td></tr>
<tr><td>Visual Representation</td><td>The computational thinking skill of decomposition is represented by a glass block in Minecraft to remind us to BREAK down a problem or task.</td></tr>
<tr><td>Real-World Example</td><td>A real-world example of decomposition would be the task of getting ready for school in the morning. "Getting ready for school" is a complex task consisting of multiple parts. How can we decompose this task into something more manageable?</td></tr>
<tr><td>Minecraft Example</td><td>An example of decomposition in Minecraft is building a rollercoaster. There are numerous tasks to complete before you have a fully operational rollercoaster in Minecraft. You need to collect materials, build a loading station with mine carts, build the frame, add the tracks, etc.</td></tr>
</table>
*This slide has animation
As you advance the slide, each answer will appear on the slide.</td>
<td>Read and discuss this slide on decomposition.
<ul>
<li>Ask students for additional real-world examples of decomposition.</li>
<li>Ask students how they use/might use decomposition in other school subjects.</li>
</ul></td>
</tr>
<tr>
<td>Slide 9

PATTERN RECOGNITION
<table>
<tr><td>Definition</td><td>Pattern Recognition is the act of finding and understanding similarities in ideas and objects.</td></tr>
<tr><td>Visual Representation</td><td>The computational thinking skill of pattern recognition is represented by a spyglass in Minecraft to remind us to LOOK for similarities in ideas, objects, and code.</td></tr>
<tr><td>Real-World Example</td><td>A real-world example of pattern recognition would be the cycle of day and night. The term "one day" is determined by the time the Earth takes to rotate once on its axis and includes both daytime and nighttime. The repeating pattern of day and night happens continuously as the Earth is always rotating, even though we can't feel it!</td></tr>
<tr><td>Minecraft Example</td><td>An example of pattern recognition in Minecraft is creating banners. Patterns can be added to banners using dyes. Placing the dyes in the crafting grid will create a color pattern.</td></tr>
</table>
*This slide has animation
As you advance the slide, each answer will appear on the slide.</td>
<td>Read and discuss this slide on pattern recognition.
<ul>
<li>Ask students for additional real-world examples of pattern recognition.</li>
<li>Ask students how they use/might use pattern recognition in other school subjects.</li>
</ul></td>
</tr>
<tr>
<td>Slide 10</td>
<td>Read and discuss this slide on abstraction.
<ul>
<li>Ask students for additional real-world examples of abstraction.</li>
<li>Ask students how they use/might use abstraction in other school subjects.</li>
</ul></td>
</tr>
</table>

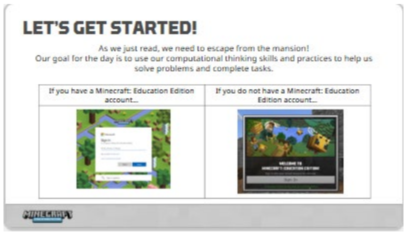
<table>
<tr>
<td>
ABSTRACTION
<table><tr><td>Definition</td><td>Abstraction is the process of simplifying–filtering out extra details about characteristics or patterns we don't need so we can focus on what's essential to solve the problem.</td></tr><tr><td>Visual Representation</td><td>The computational thinking skill of abstraction is represented by a pickaxe in Minecraft to remind us to PICK out the most important information to solve the problem.</td></tr><tr><td>Real-World Example</td><td>A real-world example of abstraction would be baking bread in an oven. We need to provide the ingredients for the bread, switch the oven on, and select the temperature and number of minutes to bake the bread. However, we don't need to know is how the oven is working internally to bake bread.</td></tr><tr><td>Minecraft Example</td><td>An example of abstraction in Minecraft is planting saplings. A player needs to provide enough space for the sapling to grow and receive sunlight. However, there is code running in the background for the sapling to grow into a tree that we do not need to understand.</td></tr></table>
*This slide has animation
As you advance the slide, each answer will appear on the slide.</td>
<td></td>
</tr>
<tr>
<td>Slide 11

ALGORITHMS
<table><tr><td>Definition</td><td>Algorithms are a set of step-by-step instructions (represented as blocks or lines of code) to solve a problem or complete a task.</td></tr><tr><td>Visual Representation</td><td>The computational thinking skill of creating algorithms is represented by a crafting table in Minecraft to remind us to CRAFT (or create) a set of step-by-step instructions to solve a problem or complete a task.</td></tr><tr><td>Real-World Example</td><td>A real-world example of algorithms would be traffic lights. Some traffic lights will automatically run through green, yellow, and red based on time. Otherwise, there are sensors that determine the signals' timing based on the flow of traffic. The algorithm is step-by-step order that directs the traffic appropriately.</td></tr><tr><td>Minecraft Example</td><td>An example of an algorithm in Minecraft is building a house. Although there are many variations of a house, all houses need walls, a roof, and a door. These components will be included as instructions within the algorithm.</td></tr></table>
*This slide has animation
As you advance the slide, each answer will appear on the slide.</td>
<td>Read and discuss this slide on algorithms.<ul><li>Ask students for additional real-world examples of algorithms.</li><li>Ask students how they use/might use algorithms in other school subjects.</li></ul></td>
</tr>
<tr>
<td>Slide 12

COMPUTATIONAL THINKING
Now that you are armored with these critical skills, you are ready to take on ANYTHING... or anybody?! You never knows what lies ahead...</td>
<td><b>Say</b>: <i>"Now that you are armed with these critical skills, you are ready to take on anything... or anybody?! You never know what lies ahead."</i></td>
</tr>
<tr>
<td>Slide 13

WELCOME TO HOUR OF CODE 2022
ESCAPE ESTATE
Select the image to view video</td>
<td>Press the play button on the video—this video clip (trailer) will provide students with a preview of what they will be doing in Hour of Code 2022.</td>
</tr>
<tr>
<td>Slide 14

MINECRAFT EDUCATION EDITION
WHAT IS AN ESCAPE ROOM?</td>
<td><b>Say:</b> <i>"This sounds amazing! But exactly what is an escape room?"</i>

(Allow for student responses)</td>
</tr>
<tr>
<td>Slide 15

WHAT IS AN ESCAPE ROOM?
An escape room is a game where players must uncover clues, solve puzzles, and complete tasks in one or more rooms with the objective of completing a specified goal in a period of time. The main goal is simple... ESCAPE!</td>
<td><b>Say:</b> <i>"An escape room is a game where player must uncover clues, solve puzzles, and complete tasks in one or more rooms. The goal is simple—you want to escape the room!</i><ul><li><i>Has anyone ever been in an escape room (in real life) or online?</i></li><li><i>What was the experience like?</i></li><li><i>Can you describe it for us?</i></li></ul></td>
</tr>
</table>

Slide 16 GOAL FOR THE DAY Today, you will be playing Escape Estate– a super exciting and fun escape room challenge in Minecraft! You will use computational thinking to solve coding puzzles to try to escape the mansion!	**Say:** *"Now that we understand how computational thinking can help us solve problems and complete tasks—we are ready to try out Escape Estate. In this Minecraft map, you will be challenged to escape from the mansion before dawn. Are you ready?"*
Slide 17  LET'S GET STARTED! As we just read, we need to escape from the mansion! Our goal for the day is to use our computational thinking skills and practices to help us solve problems and complete tasks. If you have a Minecraft: Education Edition account… If you do not have a Minecraft: Education Edition account…	Help all students to log into the Minecraft: Education Edition platform. After logging in, they should launch the HOC 2022 world. Explain to students that they should NOT start playing- just simply launch the platform. **NOTE**: This is a single-player game. Although they will all be seeing the same world, each person will be playing their own adventure.

Guided Instruction (We Do…) (Class activity) – SECONDARY | Grades 6-12 | Ages 11-18

Tutorial Sequence Navigation in Minecraft (Slides 18-26)	Students will select their mode of play based on the device they are using (i.e., keyboard or touch). The player will then start the tutorial by completing a series of mini tasks that will help them to get familiar with how to move and utilize different elements in Minecraft. They will also understand how to use the Immersive Reader feature, as reading the dialog in this Minecraft map is critical for understanding.
Start the Minecraft Map Room 1: The Attic (Slides 27-35)	Students will press the button to begin the map. Players will meet Dr. Dodge Breakowski. He will introduce himself, extend the prize of one million emeralds, and then ask the players to move towards the mansion. The mansion will pull the player in deposit them into Room 1: The Attic. Players will locate the lost diary and read the attic entry. Then, players will need to investigate the noise! Upon their investigation, players will find the Agent under a trapdoor. Players will now need to select their programming language: blocks or Python.

	NOTE: We recommend younger students or students with little to no coding background to select **blocks** for this experience. Once you select a programming language, you cannot change it, or you must restart the game.
Blocks: Coding Demonstration Room 1: The Attic (Slides 36-48)	They will code a solution to rescue the Agent. Upon rescuing the Agent, the player will be rewarded with the door to the green pathway be unlocked for them. The player will also be able to choose to talk to the Agent again to open the yellow door and the red door. To leave Room 1: The Attic, players must select a pathway: Green Pathway, Yellow Pathway, or the Red Pathway.
Python: Coding Demonstration Room 1: The Attic (Slides 49-60)	They will code a solution to rescue the Agent. Upon rescuing the Agent, the player will be rewarded with the door to the green pathway be unlocked for them. The player will also be able to choose to talk to the Agent again to open the yellow door and the red door. To leave Room 1: The Attic, players must select a pathway: Green Pathway, Yellow Pathway, or the Red Pathway.
There are two options on how to continue Hour of Code 2022: Escape Estate. • **Option 1**: You can continue to work through the world as a guided experience (i.e., complete the game as a whole group experience; model/demonstrate each coding puzzle as needed). This option is recommended for younger students, who may need additional support in fully understanding the coding concepts. There are also additional slides (slides 61-85) to provide additional support for the green pathway. These slides will provide additional hints for students! • **Option 2**: You can release students to work independently on completing their selected pathway. Once they have finished completing their pathway, you can regroup as a class to discuss the experience, address the reflection questions, and consider ways to extend the learning.	

Independent Work (You Do…)

At this point, you are going to allow students to continue through their pathway through the Escape Estate.

Overview of the Coding Puzzles

START: All players will start in Room 1: The Attic. Then players will select one of the color pathways. ALL pathways end in the Wrong Birthdays Puzzle.

Green Pathway	
Room 2: Study	Move the Clay Ball Puzzle Players will need to create a code to move the clay ball to the gold block.
	Mirror, Mirror Puzzle A mirror has suddenly appeared in the room! Players need to make the current room match the room in the mirror.
	Speedy Knight Puzzle We need to flip all four levers to open the door, but we are not quick enough to do it manually. Players need to figure out the order the levers need to be flipped to program the Speedy Knight to do it for us!
Room 3: Kitchen	Making Bread Recipe Puzzle The Zombie Chef is making bread. Use the recipe on the wall to guide the creation of your algorithm to make the bread.
	Hungry House Puzzle The House is hungry! What could be better than apples, salmon, and mushroom stew?! Look at the moving conveyor belt behind the table to figure out how many ingredients to feed the house.
	Mixed-Up Meals Puzzle The Zombie Chef mixed up all the meals. Search the kitchen for the menu and make sure everyone ends up with the correct meals.

Yellow Pathway	
Room 5: Alchemy Room	Move the Clay Ball Puzzle Players will need to code the clay ball to move to the emerald block.
	Clay Golem Puzzle The clay golem needs to move forward 6 blocks; When the golem steps on a pressure plate, it will open a pit for 1 second so the golem will need to wait for the block to come back before moving.
	Potion Puzzle The player needs to create a potion—the ingredient list can be found on the wall.
Room 6: Conservatory	Breaking Blocks Puzzle Players need to break blocks to allow for the water to start flowing from the fountain.
	Plugging Holes Puzzle Players need to use the cursor to place gold blocks to plug the holes in the fountain.
	Flower Patterns Puzzle Players need to use the cursor to create a code to fill in the middle flower bed based on the colored flowers (blue, yellow, red) displayed on the side flower beds.
Room 7: Laboratory	The Hopper Puzzle Players need to drop additional color blocks to have the hopper start.
	Animal Scale Puzzle Players will need to place the appropriate combination of animals on the scale to total 23 pounds.
	Mob Passageway Puzzle The mobs need to be sorted into the right passageways. The mobs are already queued up in the mine carts—they just need to be sorted into their passageways!

Red Pathway

Room 8: The Moat	Crossing the River Puzzle Can you help the Enderman Butler cross the river? Players will need to teleport the Butler and then have the Butler switch the lever.
	Enderman Maze Puzzle When one door closes, another door opens… Use the light switches to open or close doors in the maze. Then, help the Enderman Butler navigate to the emerald blocks.
	Flashing Lights Puzzle These colors and sounds seem to be repeating… can you repeat the pattern for the Enderman Butler to follow?
Room 9: The Cellar	Staircase Puzzle Build a staircase to reach the top of the platform in the cellar.
	Blocking the Wind Puzzle It’s super windy in here! Can you place blocks over those open holes to control the heavy winds?
	Build a Bridge Puzzle Build a bridge for the Crate Minion to move across safely. Then, break down the wall blocking the next red door!
	Block Breaker Puzzle Use the cursor to break the blocks in front of the door to escape.
Room 10: Drawing Room	Lights On Puzzle Write some code to turn on all the lights in the drawing room.
	Red Curtain Puzzle Fill all the holes in the floor to draw up the red curtain in the room.
	Laser Show Puzzle Can you figure out which blocks to shift around? The lasers will only shine through a glass block!
Room 7: Laboratory	The Hopper Puzzle

	Players need to drop additional color blocks to have the hopper start.
	Animal Scale Puzzle Players will need to place the appropriate combination of animals on the scale to total 23 pounds.
	Mob Passageway Puzzle The mobs need to be sorted into the right passageways. The mobs are already queued up in the mine carts—they just need to be sorted into their passageways!

END: All students will finish in Room 4: The Parlor.

Room 4: Parlour	**Scrambled Picture Puzzle** Oh no! The picture has gotten all scrambled. Can you push some blocks around to fix the picture?
	Save the Queen The Queen is on top of the platform. Can you build something to help you get up there?
	Wrong Birthdays Puzzle The King and Queen are freed now, but they are not on their right spot on the chessboard. Can you move the pieces to their right birthdays?

Once students complete the coding puzzles in the Parlor, a mystery will unfold! Students will finally find out what is really going on in this Escape Estate… players will then have to finish the Hour of Code with the most epic boss battle!

Turns out that Dr. Dodge Breakowski is NOT who we think…

THE TIME AGENTS STRIKE AGAIN!!!
(Not sure who the Time Agents are? No worries… Check out Hour of Code 2021: Time Craft to find out about these pesky little time agents!)

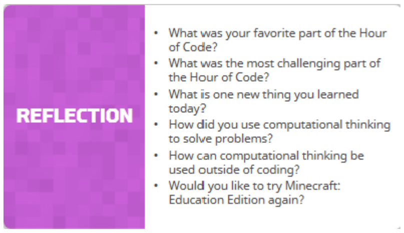
The players will need to defeat the time agents once and for all! Players will have to solve 4 coding puzzles to solve to set the chrono cannons, 1 coding puzzle to activate the Chrono Cannons, and then 1 coding puzzle to trap the time agents in the Time Orb chamber.

Completing the Hour of Code

After students have completed the Boss Battle, they will experience the ending sequence for the Hour of Code 2022: Escape Estate.

Reflection & Celebration

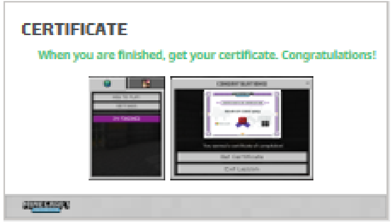
After game play has finished, gather students back together to recap their learning and to discuss the **reflection questions.**



REFLECTION

- What was your favorite part of the Hour of Code?
- What was the most challenging part of the Hour of Code?
- What is one new thing you learned today?
- How did you use computational thinking to solve problems?
- How can computational thinking be used outside of coding?
- Would you like to try Minecraft: Education Edition again?

Then provide students with their **certificate of completion.**



CERTIFICATE

When you are finished, get your certificate. Congratulations!

Continue the Learning

Did you enjoy coding in Minecraft today? Would you like to keep using Minecraft: Education Edition? Well, you are in luck! There is a full computer science progression for all students. You should check out more of the learning content in the computer science subject kit.

Educational Standards

Standards		
United States CTSA	Australia	UK - England
I can model daily processes by creating and following algorithms to complete tasks **1A-AP-08**	I can follow, describe and represent a sequence of steps and decisions (algorithms) needed to solve simple problems **ACTDIP004**	I can create and debug simple programs **Key Stage 1**
I can develop programs with sequences and simple loops, to express ideas or address a problem **1A-AP-10**	I can define simple problems, and describe and follow a sequence of steps and decisions (algorithms) needed to solve them **ACTDIP010**	I can understand what algorithms are, how they are implemented as programs on digital devices, and that programs execute by following precise and unambiguous instructions **Key Stage 1**
Debug (identify and fix) errors in an algorithm or program that includes sequences and simple loops. **1A-AP-14**	Design, modify and follow simple algorithms involving sequences of steps, branching, and iteration (repetition) **ACTDIP019**	I can use logical reasoning to predict the behaviour of simple programs **Key Stage 1**
I can compare how people live and work before and after the implementation or adoption of new computing technology **1A-IC-16**	Design algorithms represented diagrammatically and in English, and trace algorithms to predict output for a given input and to identify errors **ACTDIP029**	I can use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs **Key Stage 2**
I can create programs that include sequences, events, loops, and conditionals. **1B-AP-10**	I can define simple problems and describe and follow a sequence of steps and decisions (algorithms) needed to solve them. **ACTDIP01**	I can use sequence, selection, and repetition in programs; work with variables and various forms of input and output **Key Stage 2**
I can compare and refine multiple algorithms for the same task and determine which is the most appropriate **1B-AP-08**	I can implement simple digital solutions as visual programs with algorithms involving branching (decisions) and user input **ACTDIP011**	I can design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts **Key Stage 2**
Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended.	I can implement digital solutions as simple visual programs involving branching, iteration (repetition), and user input	I can understand several key algorithms that reflect computational thinking; use logical reasoning to compare the utility of

1B-AP-15	**ACTDIP020**	alternative algorithms for the same problem **Key Stage 3**
I can discuss computing technologies that have changed the world, and express how those technologies, and are influenced by, cultural practices **1B-IC-20**	I can implement and modify programs with user interfaces involving branching, iteration and functions in a general-purpose programming language **ACTDIP030**	I can use 2 or more programming languages, at least one of which is textual, to solve a variety of computational problems; make appropriate use of data structures; design and develop modular programs that use procedures or functions **Key Stage 3**
I can use flowcharts and/or pseudocode to address complex problems as algorithms **2-AP-10**	I can collect, access and present different types of data using simple software to create information and solve problems **ACTDIP009**	
I can design and iteratively develop programs that combine control structures, including nested loops and compound conditionals **2-AP-12**		
I can decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs **2-AP-13**		

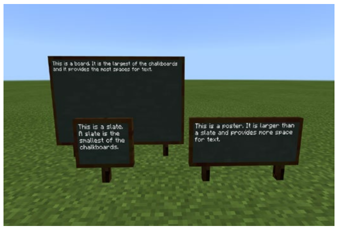
MINECRAFT VISUAL GLOSSARY

Agent

personal robot who can complete tasks for you

Chalkboards

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



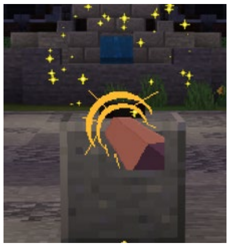
This is a board. It is the largest of the chalkboards and it provides the most spaces for text.

This is a slate. A slate is the smallest of the chalkboards.

This is a poster. It is larger than a slate and provides more space for text.

Button

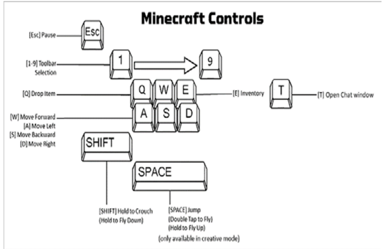
a block that can be pressed to activate several things



Controls

(keyboard)

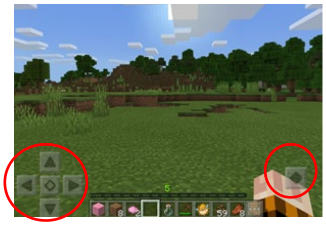
keyboard buttons that help you move around and complete tasks



Controls

(touch)

the touch pad that helps you move around and complete tasks

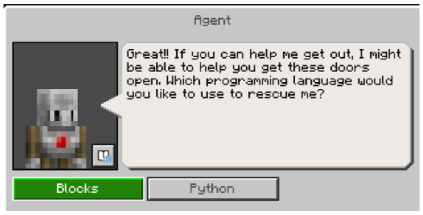


Diary

book containing entries about the rooms in the Escape Estate

Dialog

a written conversational exchange between the player and NPC

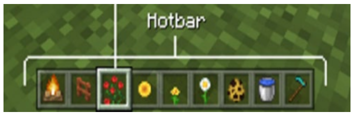


Agent

Great!! If you can help me get out, I might be able to help you get these doors open. Which programming language would you like to use to rescue me?

Hotbar

selection bar that appears on the bottom of the screen

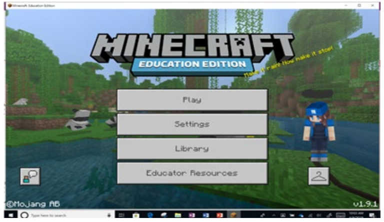


Lever

a non-solid block that can provide switchable power

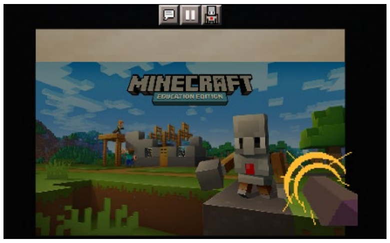
Minecraft: Education Edition

a game-based learning platform



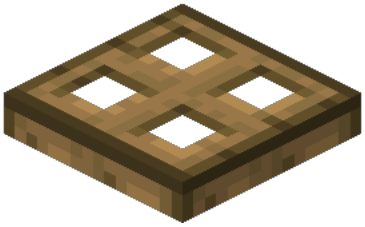
Spawn Point

the location where a player begins game play



Trapdoor

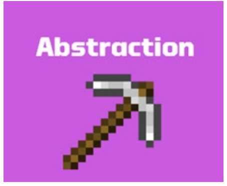
a solid, transparent block that can be used as an openable barrier



COMPUTER SCIENCE GLOSSARY

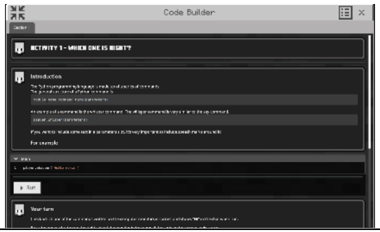
Abstraction

process of simplifying– filtering out extra details we don't need so we can focus on what’s essential to solve the problem



Azure Notebooks

user interface for a document that includes executable code, graphics, and narrative text



Code Builder

ACTIVITY 1 - WHICH ONE IS RIGHT?

Introduction

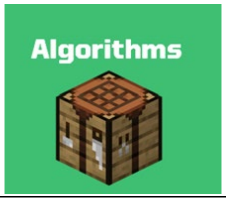
For example

Run

Your turn

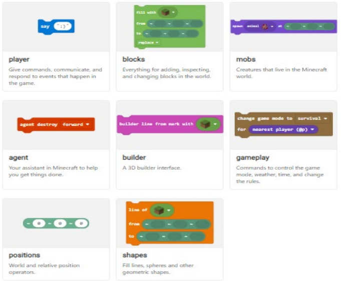
Algorithm

a set of step-by-step instructions (represented as blocks or lines of code) to solve a problem or complete a task



Blocks

blocks used to create code



player
Give commands, communicate, and respond to events that happen in the game.

blocks
Everything for adding, inspecting, and changing blocks in the world.

mobs
Creatures that live in the Minecraft world.

agent
Your assistant in Minecraft to help you get things done.

builder
A 3D builder interface.

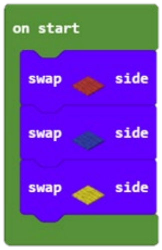
gameplay
Commands to control the game mode, weather, time, and change the rules.

positions
World and relative position operators.

shapes
Fill lines, spheres and other geometric shapes.

Code (Coding)

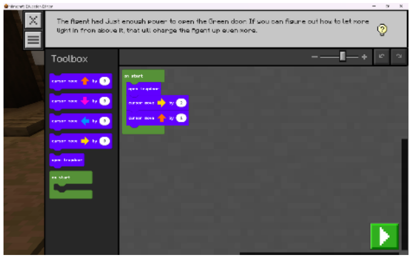
the method of giving a computer specific instructions to perform a task



```
on start
    swap side
    swap side
    swap side
```

Code Builder

coding editor used to create programs in Minecraft



The Agent had just enough power to open the Green door If you can figure out how to let more light in from above it, that will charge the Agent up even more.

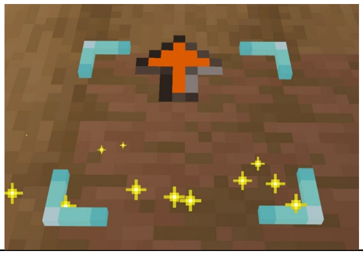
```
on start
    open trapdoor
    cursor move by 2
    cursor move by 1
```

Computational Thinking

an approach to problem solving using specific skills and practices

Cursor

the box with the orange arrow on the floor; helps you to track your code



Debugging

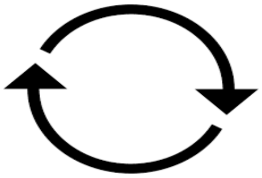
the process of finding (and correcting) errors in a computer program

Decomposition

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

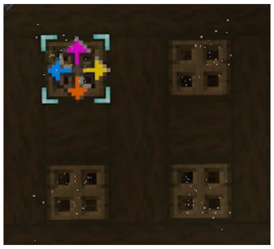
Loop

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



Orientation

the determination of the relative position of something or someone



Pattern Recognition

act of finding and understanding similarities in ideas and objects

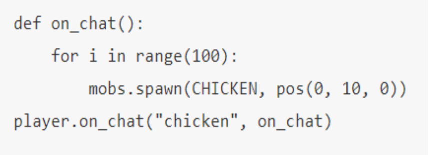
Pseudocode

writing out an algorithm in plain language, instead of in code

```
Press Start
Agent Move ↑ by 2
Agent Turn →
```

Python

a computer programming language that uses a mixture of recognizable words and characters



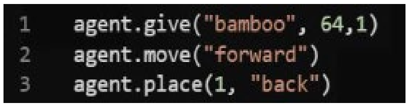
```
def on_chat():
    for i in range(100):
        mobs.spawn(CHICKEN, pos(0, 10, 0))
player.on_chat("chicken", on_chat)
```

Sequence

a programmed (coded) algorithm

String

a linear sequence of characters, words, or other data types



```
agent.give("bamboo", 64,1)
agent.move("forward")
agent.place(1, "back")
```

Syntax

the rules that control the structure of the symbols, punctuation, and words of a programming language

```
player.say(“2”)
```

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