Unit 4, Lesson 1

DECOMPOSITION

3RD & 4TH GRADE

CS FOR ALL

Award #1837380

Lesson created by the GMU-ODU CSforAll Team. For more information about
this lesson and our CSforAll initiative, contact Dr. Amy Hutchison at ahutchison1@ua.edu

REMINDER FOR TEACHERS

Prior to beginning this Unit, be sure to **assign your students a story in CoCo**, using **Level 5.**

Please use the following naming strategy for assigning the story in CoCo:
“Unit # + Descriptor”, for example, “Unit 4 Summary”

Students should use the same naming strategy for their final Scratch Project:
“Student Name + Unit # + Descriptor”, for example, “John’s Unit 4 Summary”

SUMMARY AND STANDARDS

Summary:

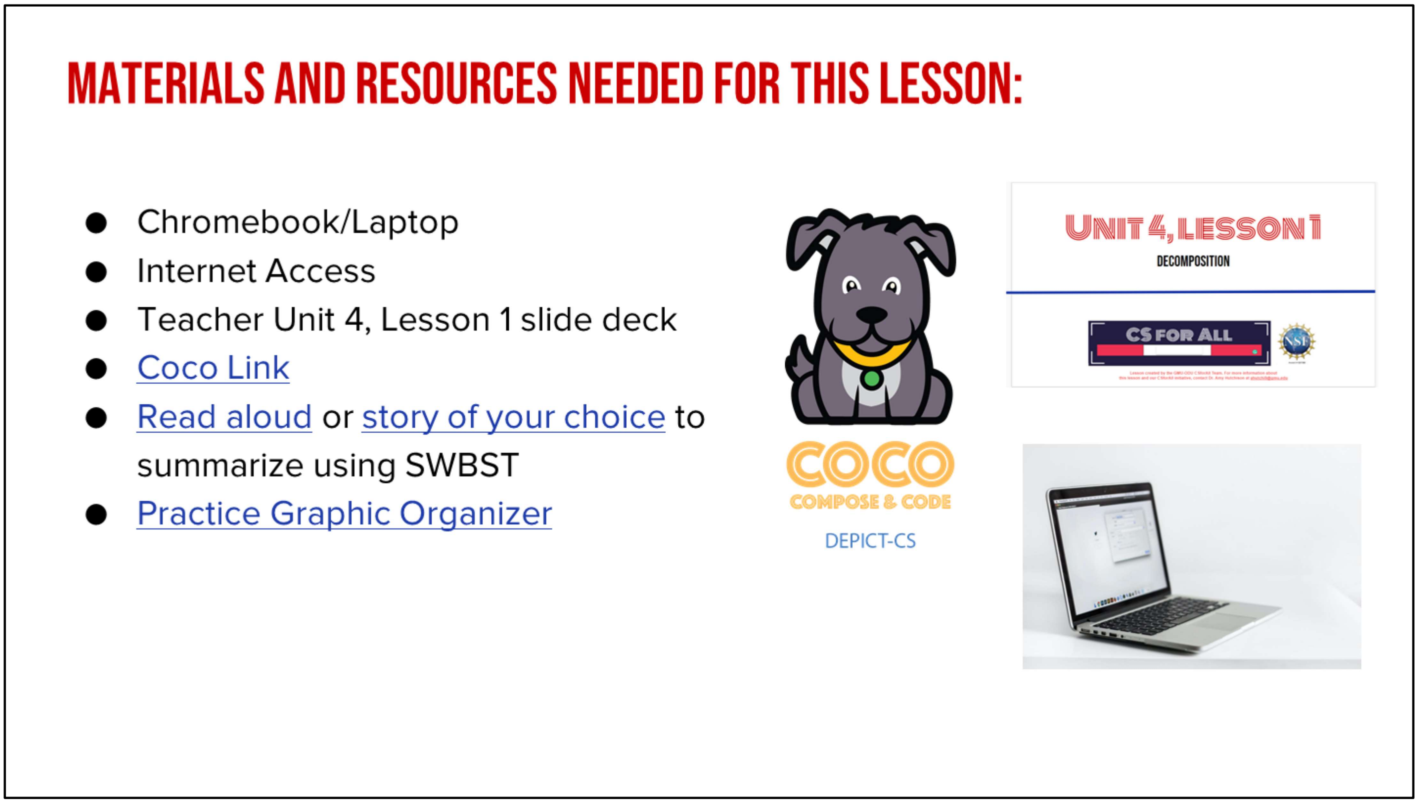
In this lesson, students will use a summary strategy to decompose a story in Coco.

ELA Standards

The student will write in a variety of forms to include narrative, descriptive, opinion, and expository.
a) Engage in writing as a process.
b) Identify audience and purpose.
c) Use a variety of prewriting strategies.
d) Use organizational strategies to structure writing according to type.
e) Use transition words to vary sentence structure.

CS Standards:

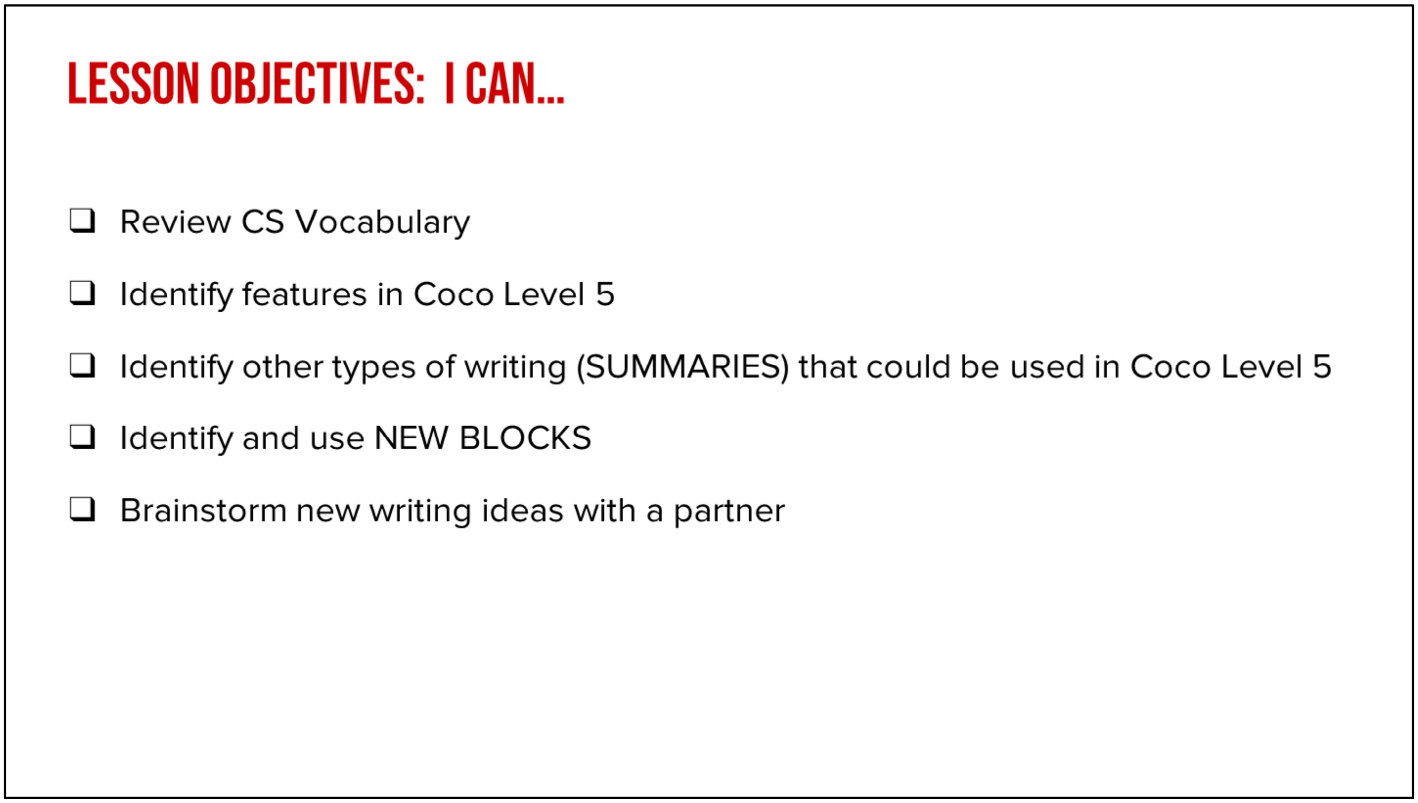
The student will break down (decompose) a larger problem into smaller sub-problems, independently or collaboratively.



MATERIALS AND RESOURCES NEEDED FOR THIS LESSON:

- Chromebook/Laptop
- Internet Access
- Teacher Unit 4, Lesson 1 slide deck
- Coco Link
- Read aloud or story of your choice to summarize using SWBST
- Practice Graphic Organizer

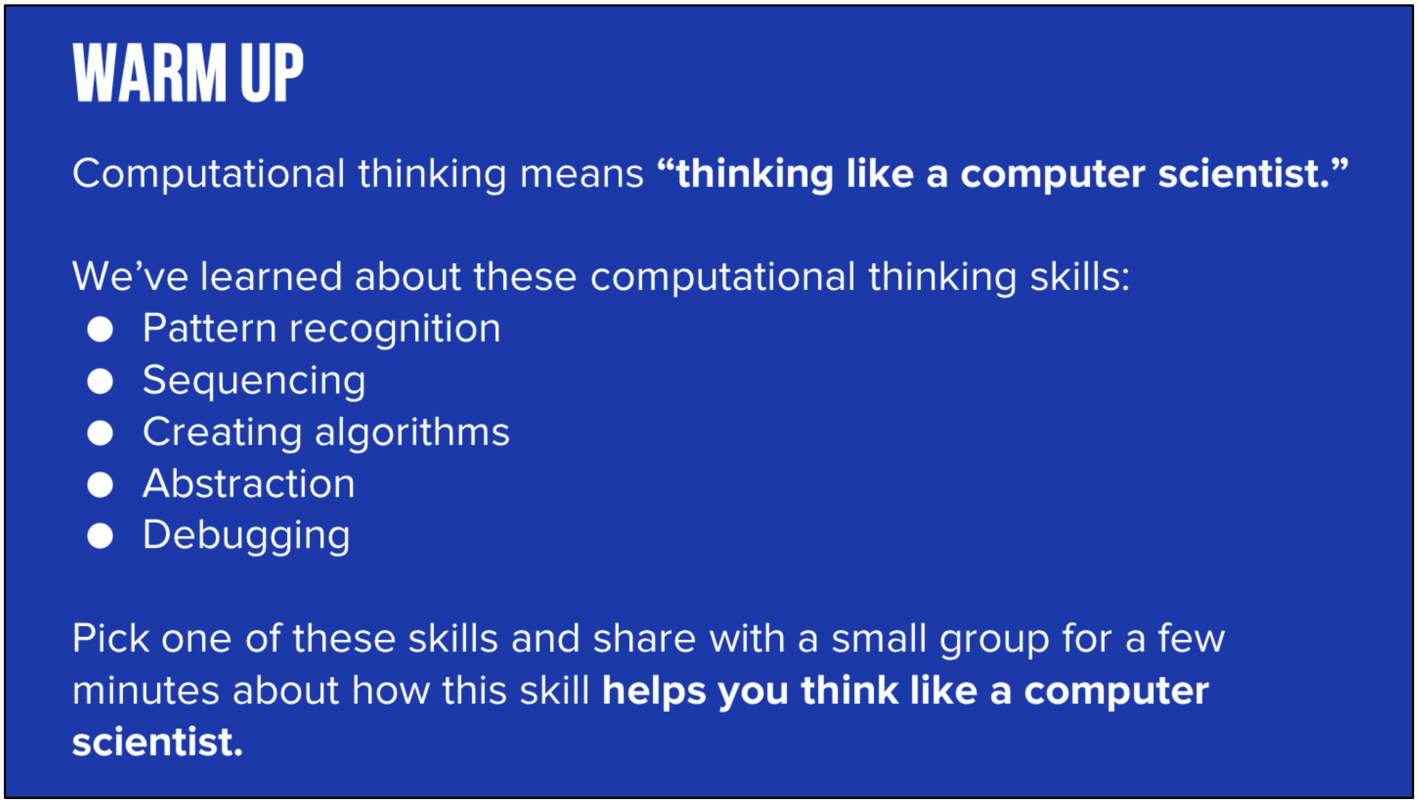
You will need….[read slide]



LESSON OBJECTIVES: I CAN...

- □ Review CS Vocabulary
- □ Identify features in Coco Level 5
- □ Identify other types of writing (SUMMARIES) that could be used in Coco Level 5
- □ Identify and use NEW BLOCKS
- □ Brainstorm new writing ideas with a partner

Let’s go over today’s lesson’s objectives: [read slide]



WARM UP

Computational thinking means **“thinking like a computer scientist.”**

We’ve learned about these computational thinking skills:

- Pattern recognition
- Sequencing
- Creating algorithms
- Abstraction
- Debugging

Pick one of these skills and share with a small group for a few minutes about how this skill **helps you think like a computer scientist.**

INTRODUCTION

Today we are going to start with a new vocabulary word!

DECOMPOSE

Have you ever heard the word decompose? Or even decomposition? In science you may have learned about decomposers, those organisms that break things down. Well, today we are going to learn about how we can decompose in both computer science and in our writing and literacy!

DECOMPOSE: TO BREAK A PROBLEM DOWN INTO SMALLER PIECES

DECOMPOSITION (DEE-COMP–UH-ZI–SHUN): BREAKING A LARGE PROBLEM INTO SMALLER PARTS

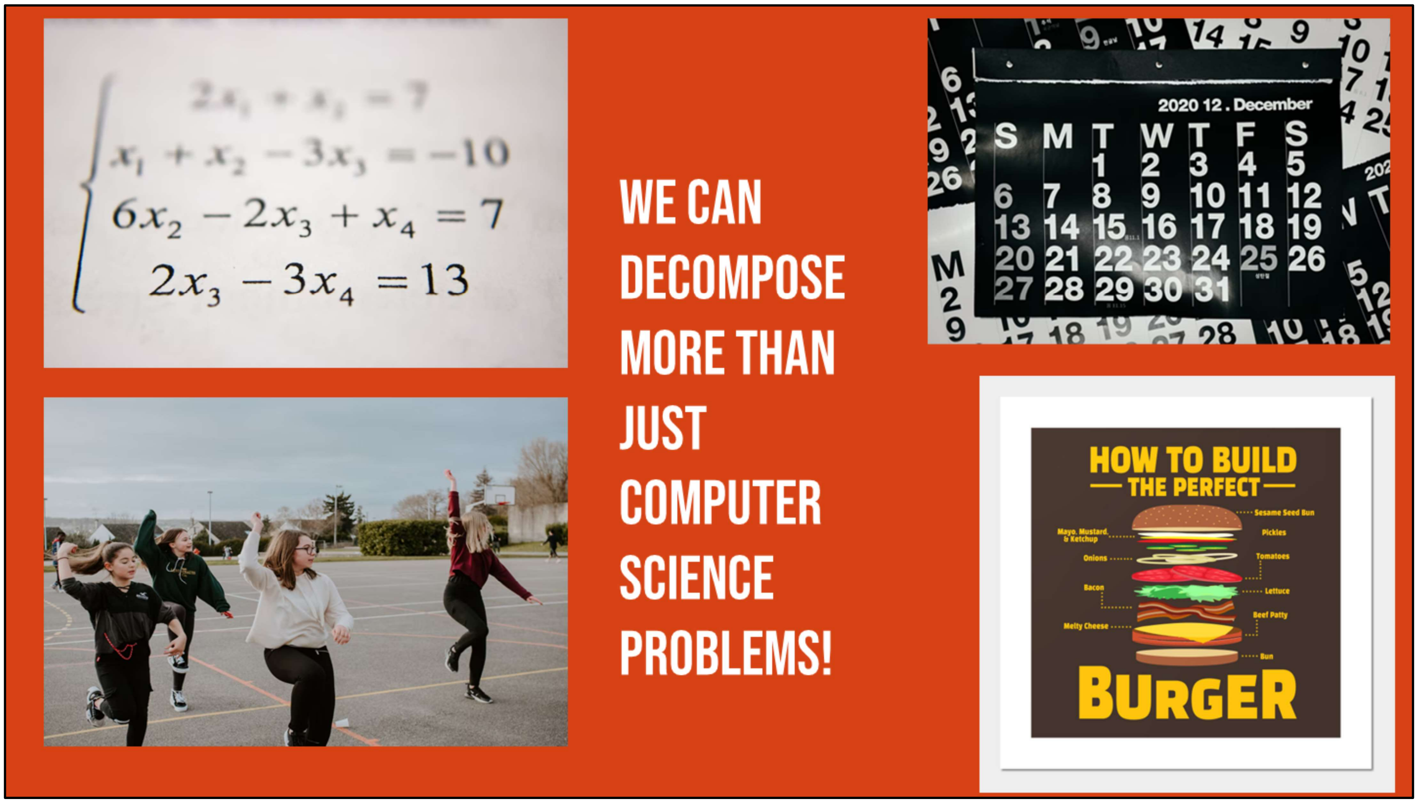
Computer scientists have to solve coding problems and need to make think about what steps they need to take and what order to do them in.

Sometimes the problem is so big or complex, it can be hard to know where to start.

Decomposition is when we break a problem down into smaller parts to make it easier to tackle.

It has many advantages. It helps us manage large projects and makes the process of solving a complex problem less scary and much easier to take on.

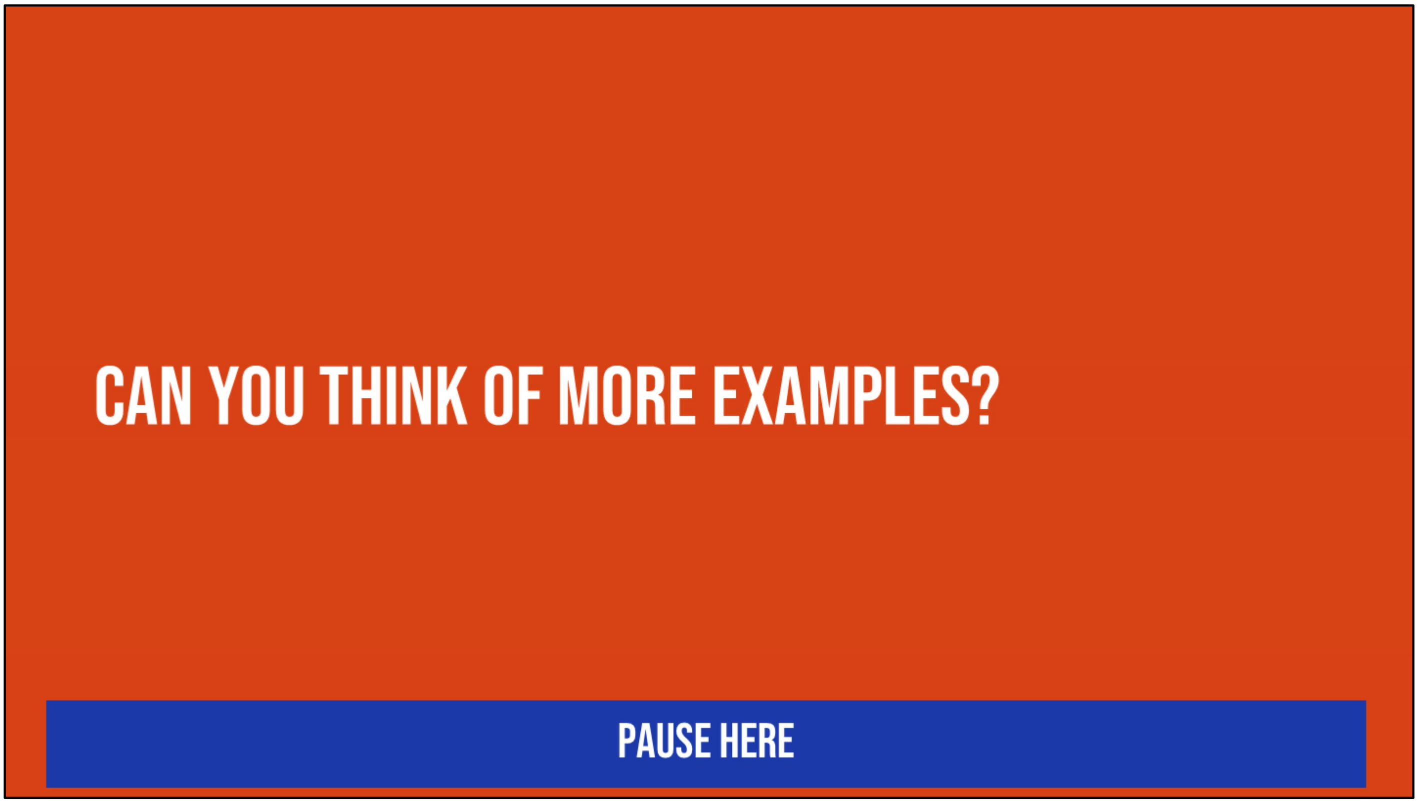
This Unit, we’re going to learn more about decomposition, our final computational thinking concept. Decomposition is going to be key in helping us to animate our new stories in Scratch with more Scratch blocks.”



WE CAN DECOMPOSE MORE THAN JUST COMPUTER SCIENCE PROBLEMS!

According to code.org: “Decompose—To break a hard problem up into smaller, easier ones”

- *This Unit, we’re going to learn more about decomposition, our final computational thinking concept. Decomposition is going to be key in helping us to animate our new stories in Scratch with more Scratch blocks.”*

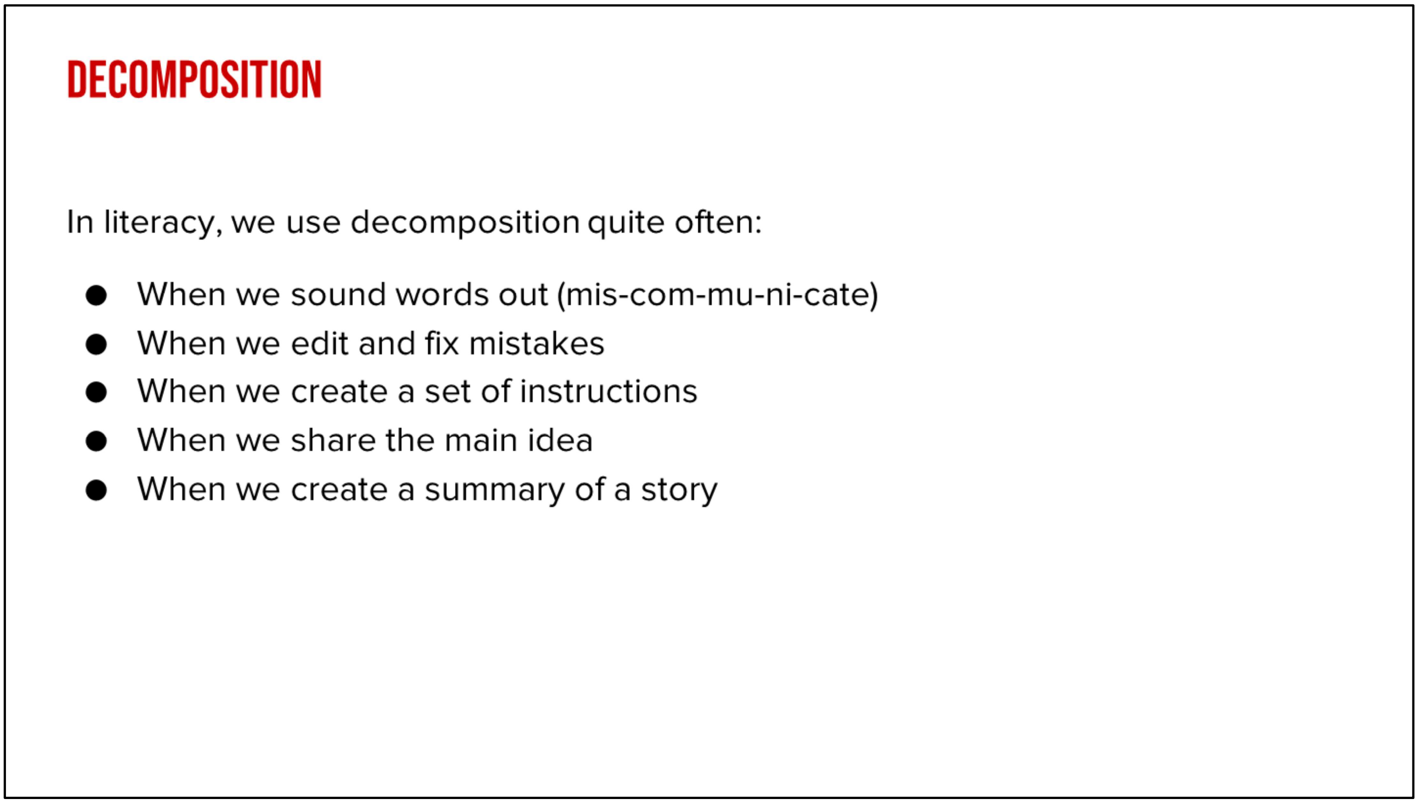


CAN YOU THINK OF MORE EXAMPLES?

PAUSE HERE

When do you break things down?

Break down a recipe, a workout, picking out your outfit, ex

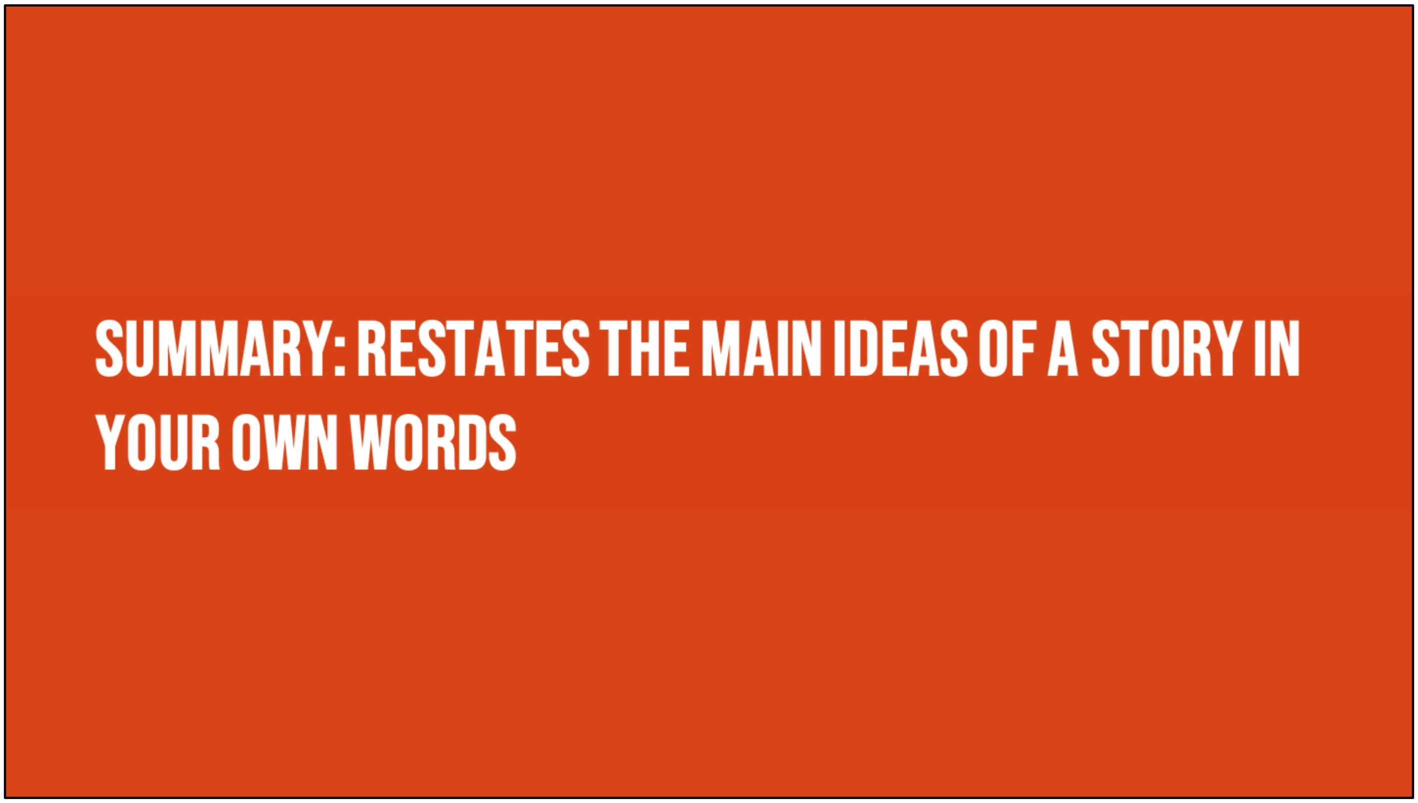


DECOMPOSITION

In literacy, we use decomposition quite often:

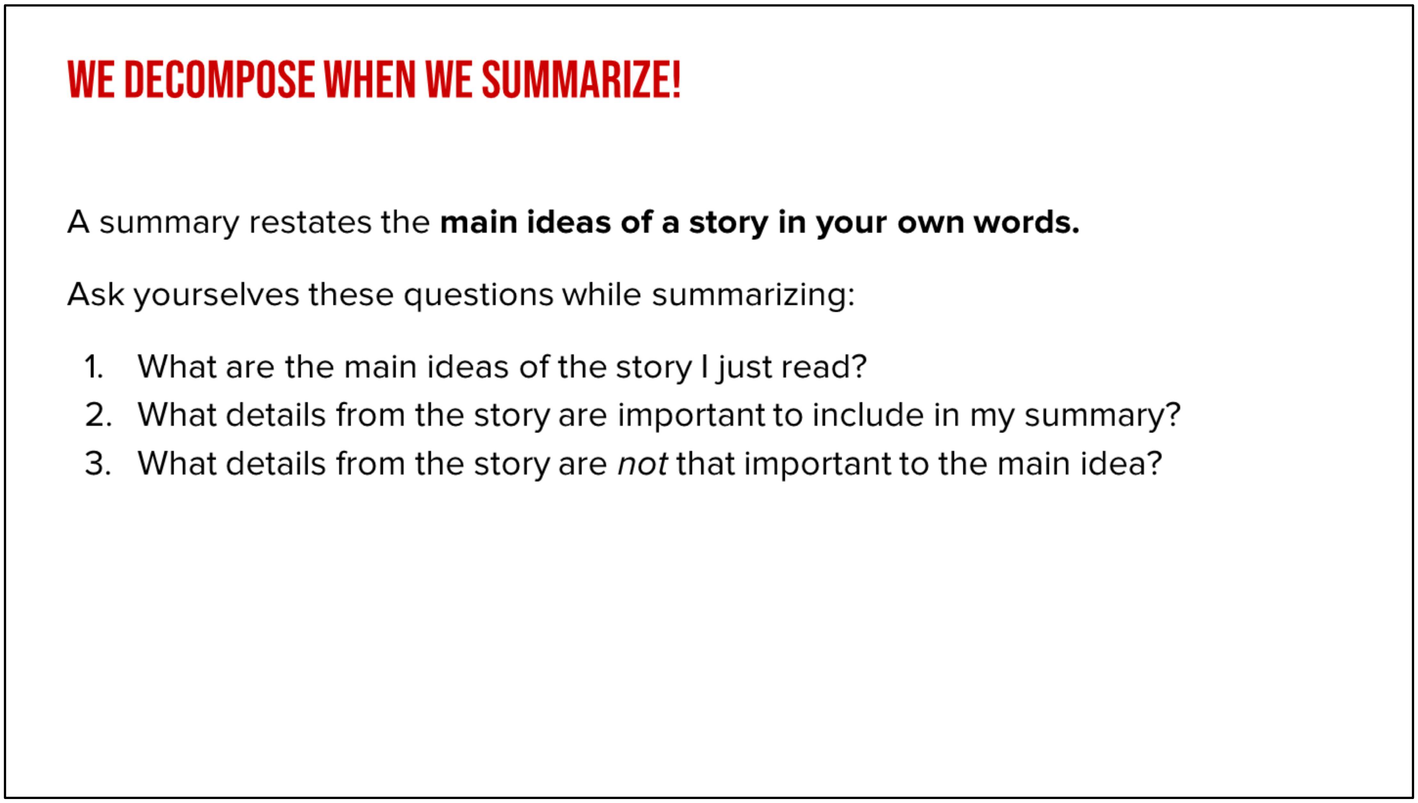
- When we sound words out (mis-com-mu-ni-cate)
- When we edit and fix mistakes
- When we create a set of instructions
- When we share the main idea
- When we create a summary of a story

We actually use the skill of decomposition quite often in literacy. We decompose when we break down a word or sentence into small parts, like sounding a word we don’t know in a sentence. We decompose when we break down our writing to look for errors, like debugging. We decompose and abstract when we break down a series of steps, like a set of instructions, into the most important parts. We also decompose when we break a story down into the main idea or try to provide a summary. Today, we are going to talk more about summarizing!



SUMMARY: RESTATES THE MAIN IDEAS OF A STORY IN YOUR OWN WORDS

A summary restates the main ideas of a story in your own words. You have probably done this quite often after you have read or listened to a story. We don’t want to retell everything that happens in the story, we just want to share the main idea



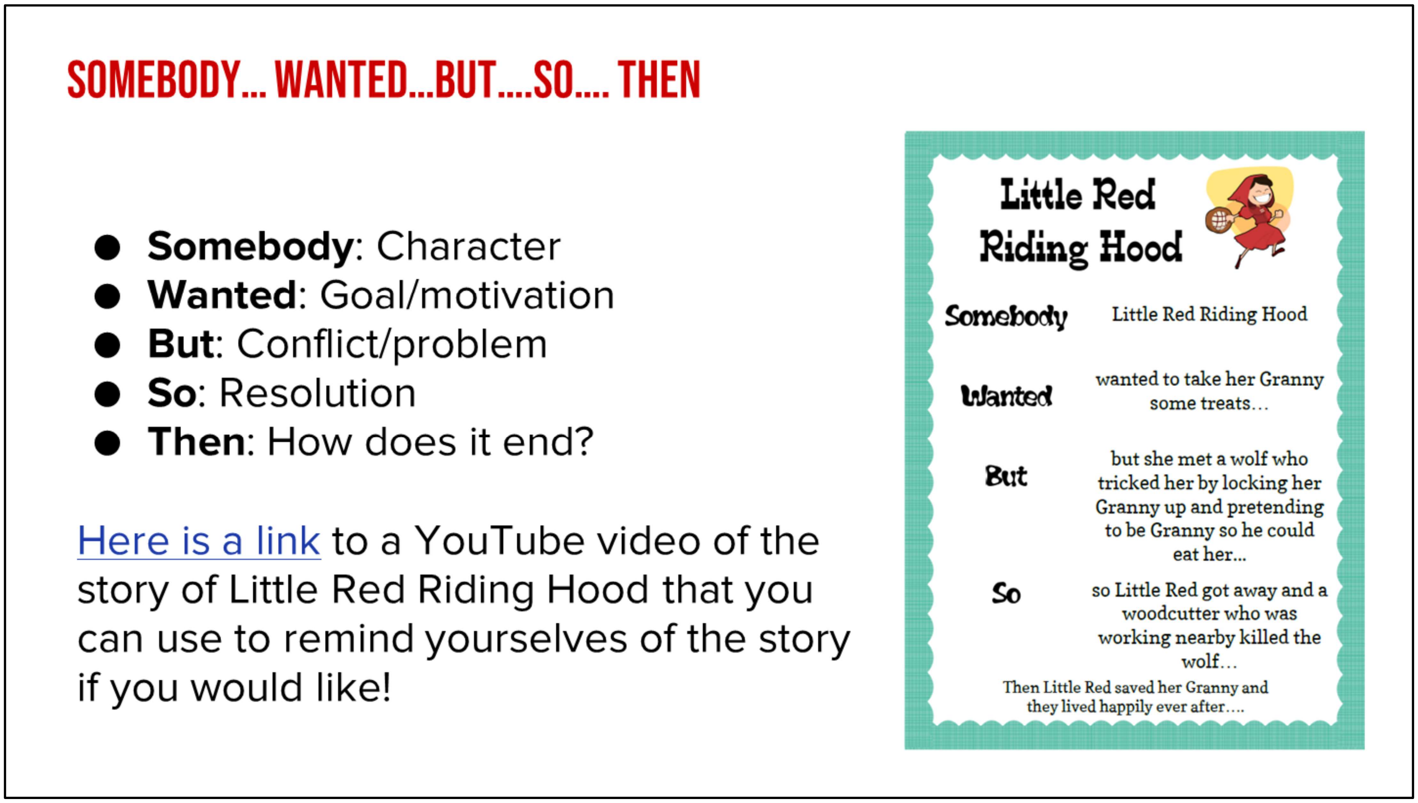
WE DECOMPOSE WHEN WE SUMMARIZE!

A summary restates the **main ideas of a story in your own words.**

Ask yourselves these questions while summarizing:

1. What are the main ideas of the story I just read?
2. What details from the story are important to include in my summary?
3. What details from the story are *not* that important to the main idea?

{Read questions}
You're decomposing because you're breaking down the entire story and identifying its most important parts.



SOMEBODY… WANTED…BUT….SO…. THEN

- **Somebody**: Character
- **Wanted**: Goal/motivation
- **But**: Conflict/problem
- **So**: Resolution
- **Then**: How does it end?

Here is a link to a YouTube video of the story of Little Red Riding Hood that you can use to remind yourselves of the story if you would like!

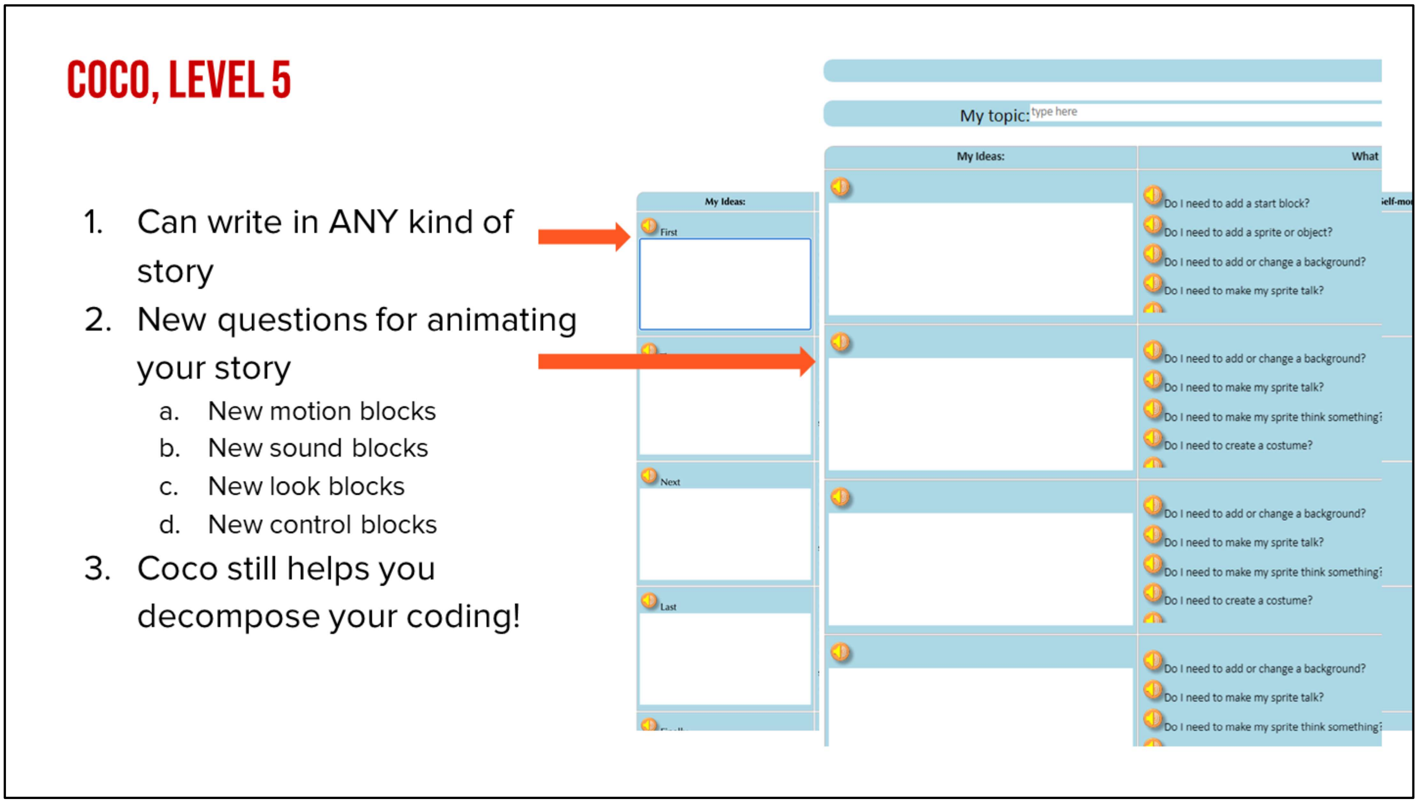
Little Red Riding Hood

Somebody	Little Red Riding Hood
Wanted	wanted to take her Granny some treats…
But	but she met a wolf who tricked her by locking her Granny up and pretending to be Granny so he could eat her…
So	so Little Red got away and a woodcutter who was working nearby killed the wolf…

Then Little Red saved her Granny and they lived happily ever after….

There are several ways that we can decompose a story, but Today, we're going to review one great strategy for summarizing. This strategy uses these anchor words to help us create a summary: "Somebody Wanted But So Then" {Read Slide}

Let's see how this strategy can help use decompose the familiar story of Little Red Riding Hood. To do this, we can use Coco, our favorite graphic organizer, to help us. Today, we are briefly going to be introduced to Level 5 of Coco, where we can record our summaries. Let me show you how.



COCO, LEVEL 5

1. Can write in ANY kind of story
2. New questions for animating your story
   a. New motion blocks
   b. New sound blocks
   c. New look blocks
   d. New control blocks
3. Coco still helps you decompose your coding!

Do I need to add a start block?
Do I need to add a sprite or object?
Do I need to add or change a background?
Do I need to make my sprite talk?

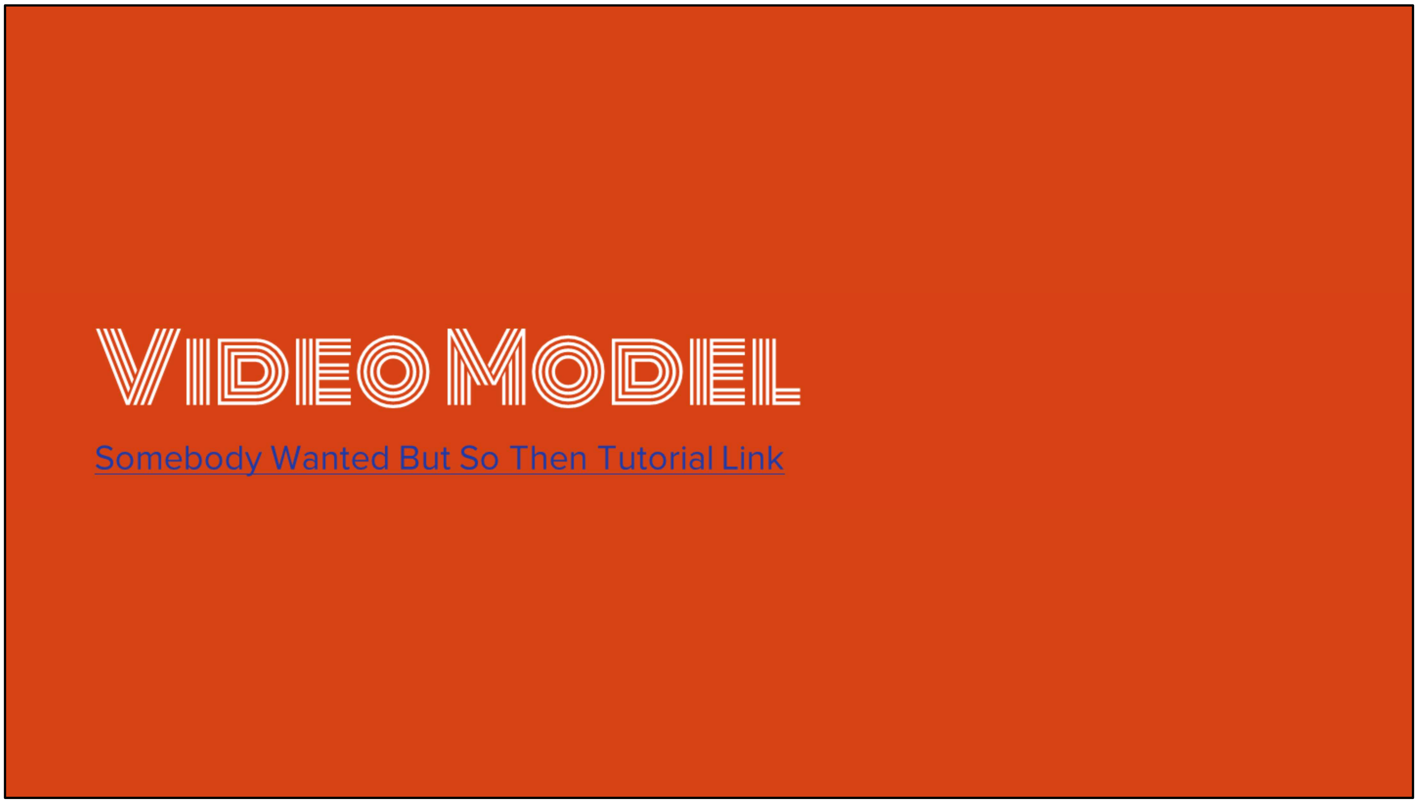
Do I need to add or change a background?
Do I need to make my sprite talk?
Do I need to make my sprite think something?
Do I need to create a costume?

Do I need to add or change a background?
Do I need to make my sprite talk?
Do I need to make my sprite think something?
Do I need to create a costume?

Do I need to add or change a background?
Do I need to make my sprite talk?
Do I need to make my sprite think something?

First, let’s take a quick look at Coco, Level 5. In this level:

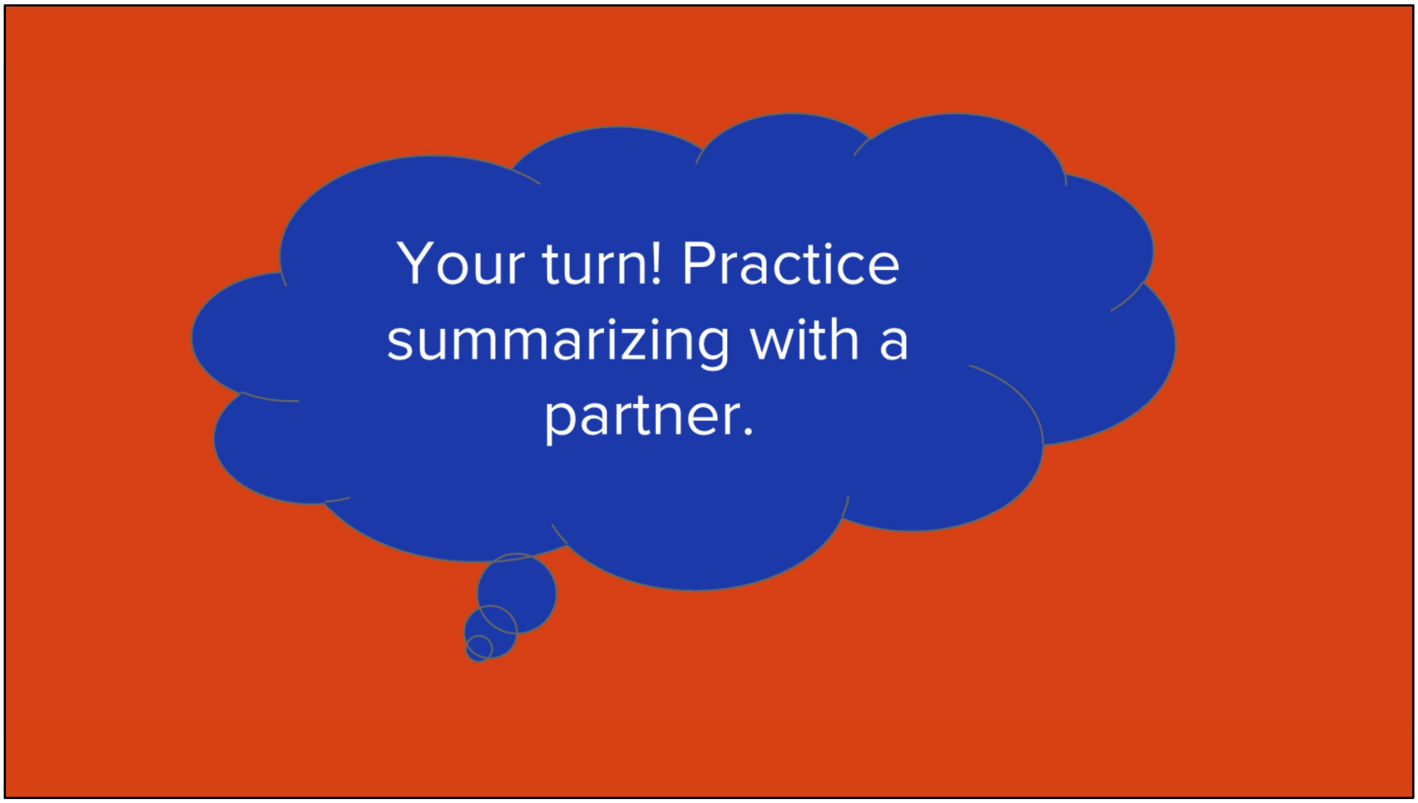
1. They are completely empty, no suggested text. To write a story, to summarize, to write a set of instructions, etc.
2. You will also see many new questions in Column 2. These questions will prompt you to use new blocks in Scratch. There are blocks that move your sprite, create sound, change looks and also blocks to control your code and create loops.
3. Even though there are some new features in this level, Coco is still a great way for you to break down each step of your code and prepare for creating a scratch animation
4. The other two columns remain the same, when you say yes to any of the questions, the block you need will appear in the 3rd column. Remember, if you click on the block a video will appear to show you how the block works.



Video Model

Somebody Wanted But So Then Tutorial Link

TEACHERS: You may model how to use CoCo Level 5 to write a summary OR play this video to model how. NOTE that the video says “Level 4,” but students should work in Level 5 of CoCo for this activity.

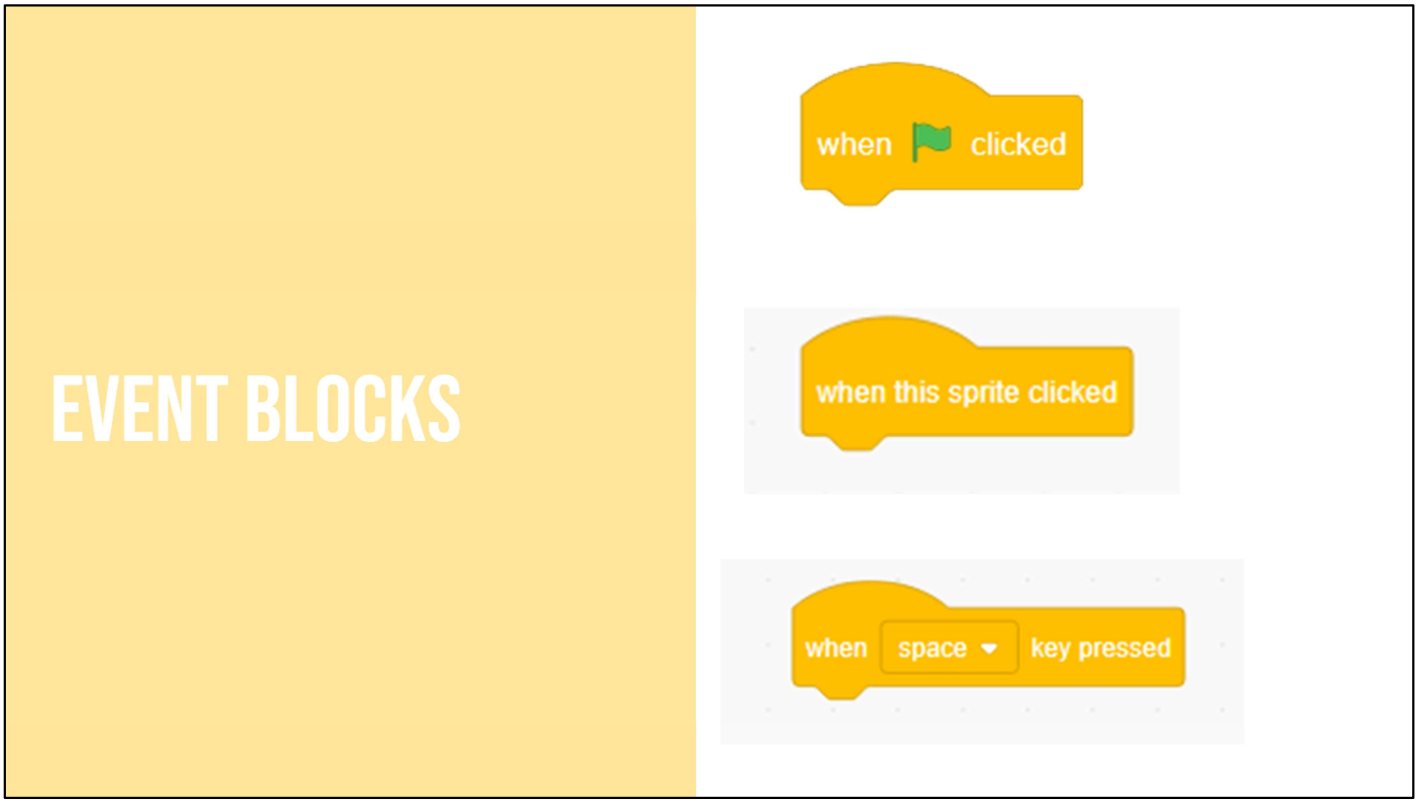


Your turn! Practice summarizing with a partner.

With a partner, practice summarizing a familiar story using the graphic organizer provided by your teacher.

https://www.dropbox.com/scl/fi/64of0bqyi4wub8xm9toxd/SWBST_Practice-graphic-organizer.docx?dl=0&rlkey=y7wyfpzguw46wnt301wq9nx20

New Scratch Blocks



EVENT BLOCKS

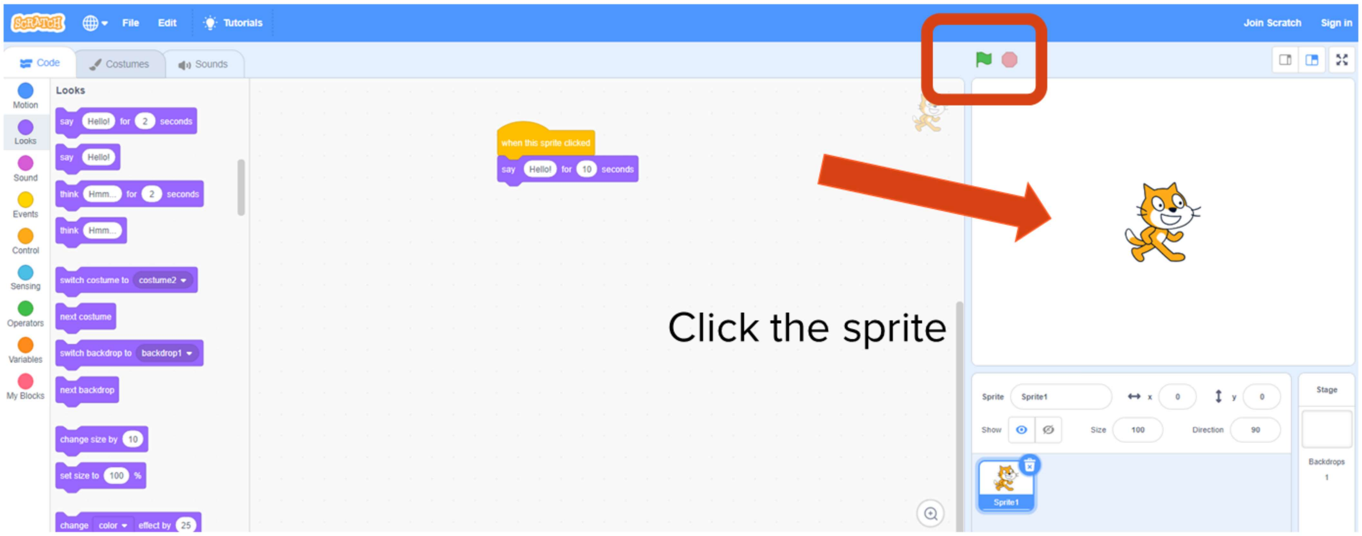
Another category of Scratch block is the ‘Event blocks.” These blocks are yellow where control blocks are more orange in color.

Events in computer science are the triggers for making action happen, like selecting the play button on any screen.

Events in Scratch are represented by the yellow codes including: when flag clicked, when sprite clicked, when key pressed. You’ve already seen the “when flag clicked” event block so today we will review the “when sprite clicked” and “when key pressed” blocks.

WHEN THIS SPRITE CLICKED BLOCK (VIDEO)

Instead of the green flag.

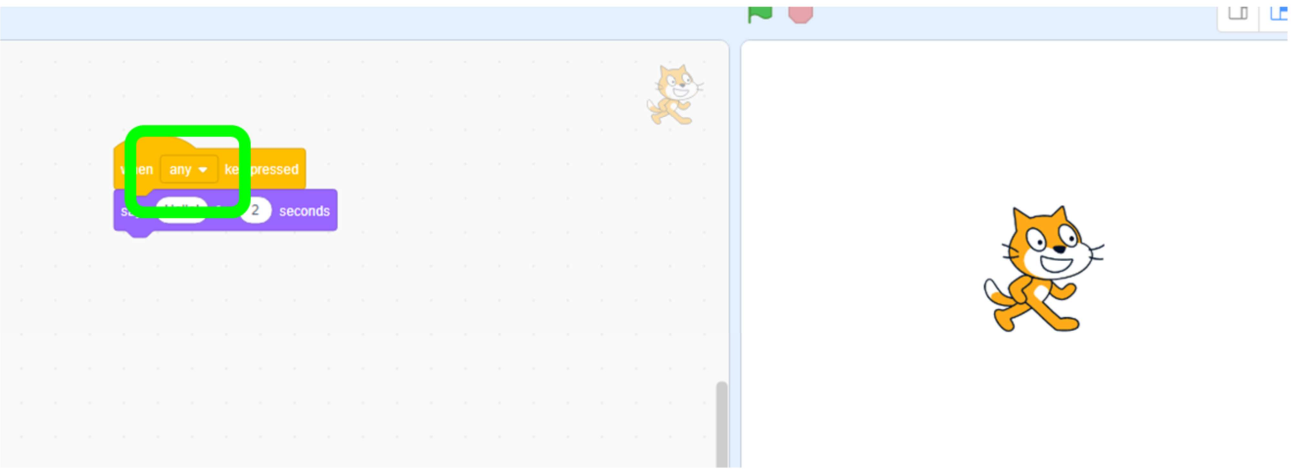


```
when this sprite clicked
say Hello! for 10 seconds
```

Click the sprite

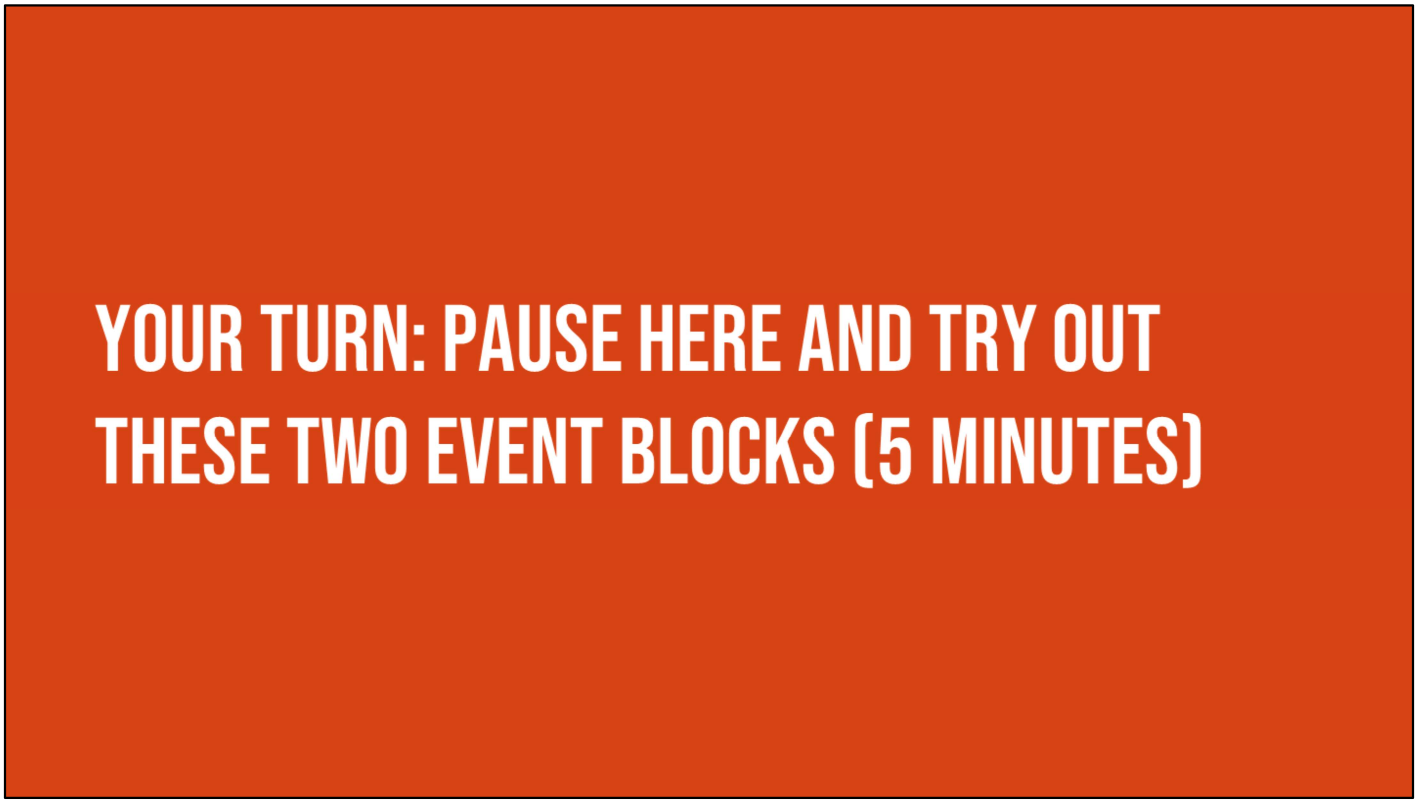
Starts the animation by clicking the sprite on the screen.

WHEN ANY KEY PRESSED BLOCK (VIDEO)



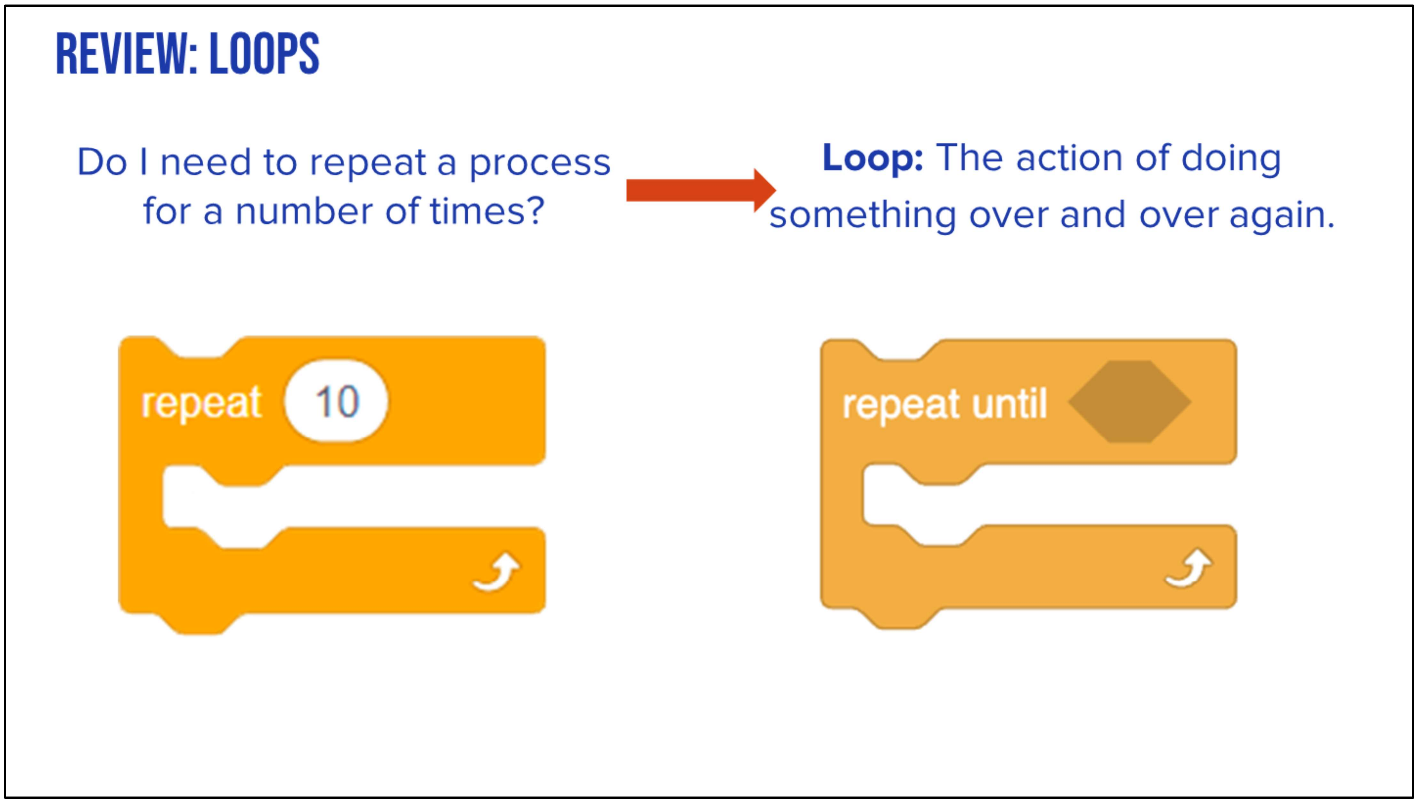
Starts the animation by pressing any key on your keyboard.

But be careful! The block’s default is set to “when space key pressed” which would only start the animation by pressing the space key. (review space key if students are unfamiliar)



YOUR TURN: PAUSE HERE AND TRY OUT THESE TWO EVENT BLOCKS (5 MINUTES)

Instruct students to create an animation of a sport or outside activity they like to do with two Sprites. Use one of the new event blocks for each Sprite. Share with a partner!



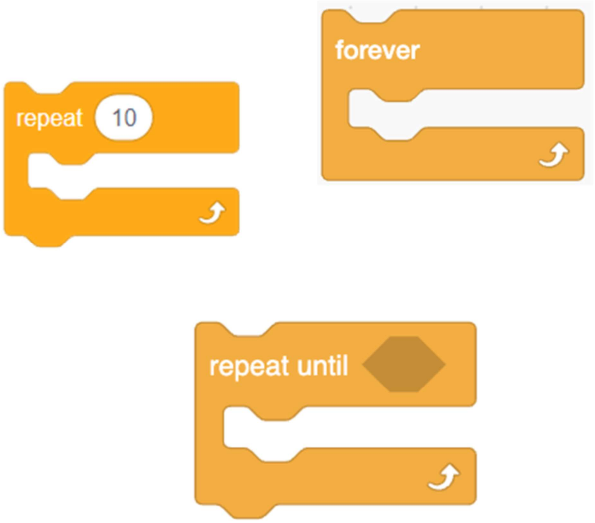
REVIEW: LOOPS

Do I need to repeat a process for a number of times? → **Loop:** The action of doing something over and over again.

{Read question again} What does this mean? Well, if you find that you need to repeat an action or a pattern over and over again, you can use what we call loops. A loop is simply the action of doing something over and over again.You can use loops to repeat steps when a task requires a repeated action or actions. Loops are efficient and easier to use than dragging the same blocks over and over.
In Scratch, there is a really nifty block that will let you loop an action or set of commands. This is the repeat block.

This block will repeat whatever code or other blocks you place inside for a certain amount of time. This is really helpful for lots of things in Scratch, but especially if we find a pattern we like and want to repeat.

CONTROL BLOCKS & LOOPS

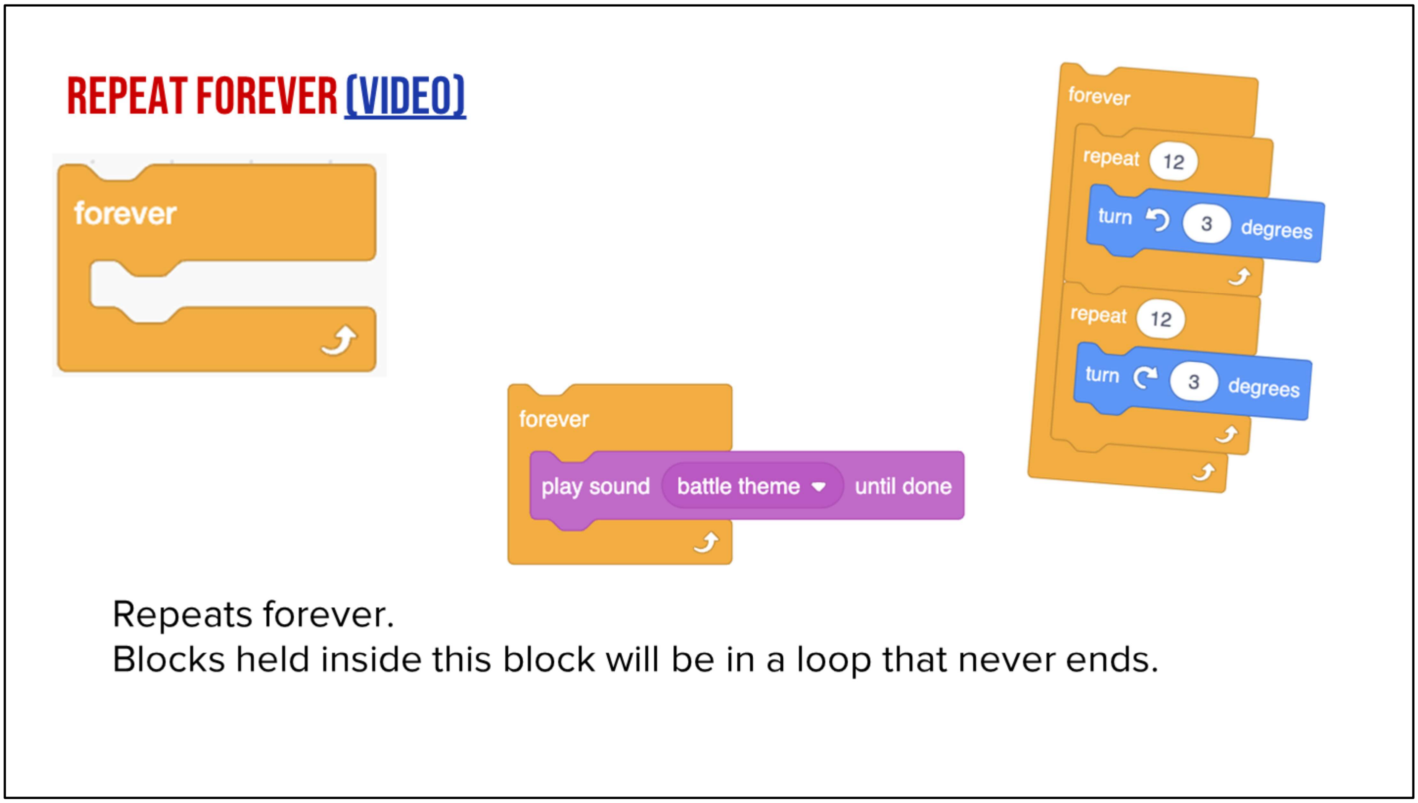


You'll notice that the last question you see in Coco asks {read question} This question prompts you to use a control block.

Control blocks are one of the nine categories of Scratch blocks. They are color-coded amber or yellow. We have already learned about three of these, the wait block, repeat X number of times, and repeat until blocks. Remember, all of the algorithms we code using these blocks are considered loops. Loops are great tools to use within code and projects to repeat an action multiple times.

Today, we will learn about the "forever block."

- Forever: Blocks held inside this block will be in a loop — just like the Repeat () block and the Repeat Until () block, except that the loop never ends (unless the stop sign is clicked, the Stop All block is activated, or the stop script block is activated within the loop). Due to this infinite loop, the block has no bump at the bottom; having a bump would be pointless, as the blocks below it would never be activated.

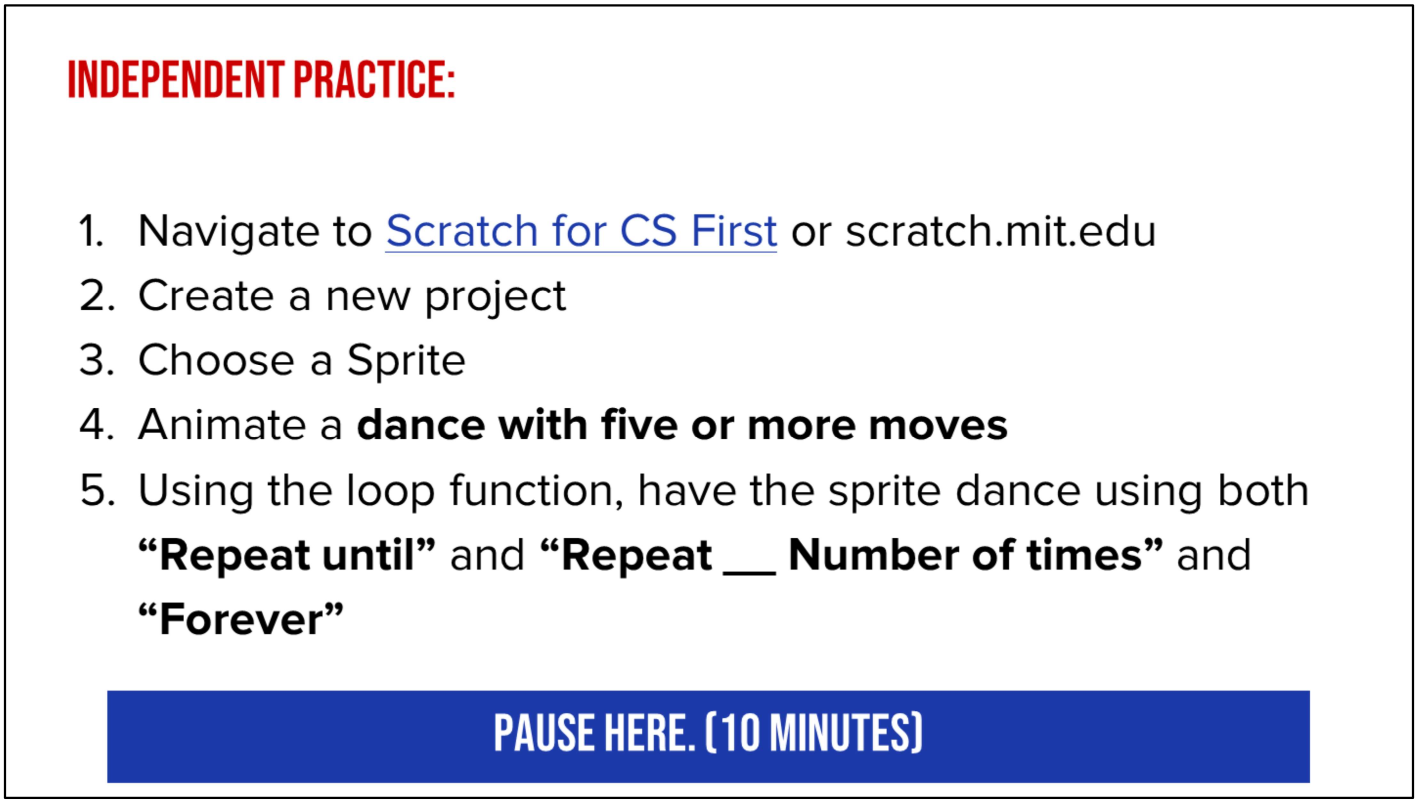


REPEAT FOREVER (VIDEO)

Repeats forever.
Blocks held inside this block will be in a loop that never ends.

Watch Video Here: https://www.dropbox.com/s/jjd1b22i1permk0/forever.mp4?dl=0

Blocks placed inside the forever block will be in a loop — just like the Repeat () block and the Repeat Until () block, except that the loop never ends (unless the stop sign is clicked, the Stop All block is activated, or the stop script block is activated within the loop). Due to this infinite loop, the block has no bump at the bottom; having a bump would be pointless, as the blocks below it would never be activated.



INDEPENDENT PRACTICE:

1. Navigate to Scratch for CS First or scratch.mit.edu
2. Create a new project
3. Choose a Sprite
4. Animate a **dance with five or more moves**
5. Using the loop function, have the sprite dance using both **“Repeat until”** and **“Repeat __ Number of times”** and **“Forever”**

PAUSE HERE. (10 MINUTES)

We did this in U3L2 but this time we’re going to expand our dance to include 5 moves and we’re going to use the “forever” loop as well.

TURN & TALK

- Have students get with a partner or small group and discuss when they may want to use a Forever block in their coding.

NEXT TIME...

In our next lesson, we will be summarizing and planning how we will animate our summaries in Scratch!

REMINDER FOR TEACHERS

Prior to beginning this Unit, be sure to **assign your students a story in CoCo**, using **Level 5.**

Please use the following naming strategy for assigning the story in CoCo: “Unit # + Descriptor”, for example, “Unit 4 Summary”

Students should use the same naming strategy for their final Scratch Project: “Student Name + Unit # + Descriptor”, for example, “John’s Unit 4 Summary”