Title of Lesson: CHOMP...Eat 'em Up!	
Overview/Purpose: Using the > (greater than) and < (less than) symbols in mathematics can be confusing. Teachers often rely on the 'alligator' analogy to explain that the mouth of the alligator eats the larger number. While this may be true, an understanding of 'greater than' and 'less than' a number is essential to understanding the progressional relationships between numbers. Keep in mind that the alligator analogy only supports the understanding of 'greater than.' There may be an implied understanding of 'less than.'	
Suggested Grade Level: Grades 2 - 3 Original Author: Kai's Edu	**Standards Alignment:** *Click Here*
Learning Target/s:	• I understand the value of written numbers based on the hundreds, tens, and ones digits. • I can compare the value of three-digit numbers. • I understand the meaning of 'greater than.'

	• I understand the meaning of 'less than.' • I understand the meaning of 'equal to.' • I understand the symbolic representation of 'greater than.' • I understand the symbolic representation of 'less than.' • I understand the symbolic representation of 'equal to.'
Pre-assessment:	Students must understand: • the value of written numbers based on the hundreds, tens, and ones digits • given two number values, which written value is more and which is less

Lesson:	

Engage:

To review the concept of comparing numbers watch Comparing Numbers by Jack Hartmann.

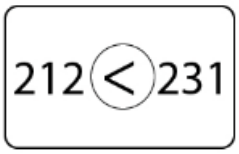
Key talking points:

- In order to compare numbers, you must first determine the value of each one.
- Once you have determined the value of each one, then compare to decide which number has a greater value and which number has a lesser value.

Show the Mrs. Alligator can Chomp video by Jack Hartmann.

Key talking points:

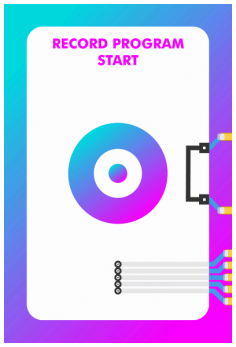
- When using the 'alligator method,' this helps us to identify the 'greater' number.
- Can a computer use the 'alligator method' to solve the following problem?

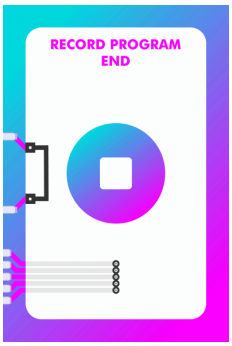


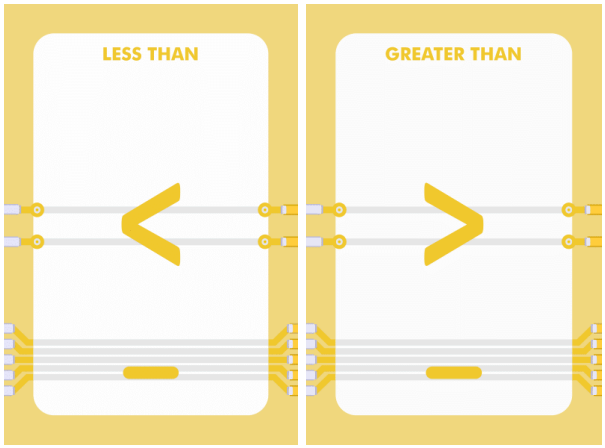
**Allow students to have their own opinions on the answer to this question. Ask them to justify their reasoning, but DO NOT correct it. Misconceptions will be cleared up as they learn more.*

Explore:

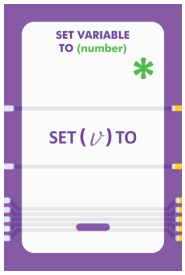
Review how a computer follows directions by showing the 'start' and 'end' cards:

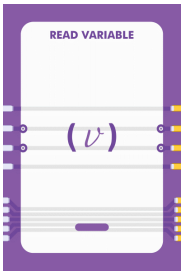




	Key talking points: • The icons in the center of each circle are symbols (icons) that tell the computer to start and stop following the direction. • The black [ ] holds all the other directions inside it. Next, introduce (or review if you did the 'Mystery Number' lesson) the 'less than' and 'greater than' cards.  **We will not be using the 'equal' cards'. The 'equal to' comparator is also referred to in coding as two equal signs back-to-back '=='. It is not setting two things equal but is returning 'true' if two things are equal. You will often want to compare two things and do something if they are equal, like checking if the current time is equal to the alarm you have set.* Key talking points: • When comparing number values, the computer reads the first number and determines if it is 'greater than' or 'less than' the variable (second number). • Computers read 'code' in a step-by-step sequence. In order for us to program the computer to evaluate two numbers, we must first teach the computer what our numbers are doing to be. This concept will be new to those who have not done the 'Mystery Number' lesson. We will have the computer read the first number (472) and then use a 'variable' container' for our second number. Along with the 'set variable' we also need a coding card to 'read variable,' once

it is set.

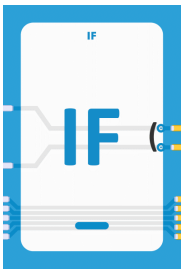


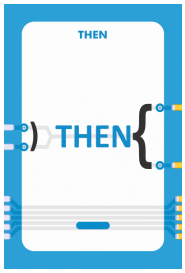


Key talking points:

- A variable is a container that holds a value, such as a piece of text or a number. The value can change, which is why it's stored in a variable.
- This card is used to set v (the variable) to a * number card, placed directly afterwards.

Next, introduce the 'IF' and 'THEN' conditional cards.

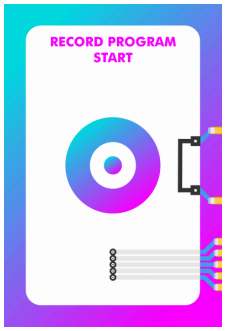


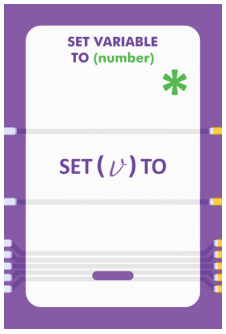


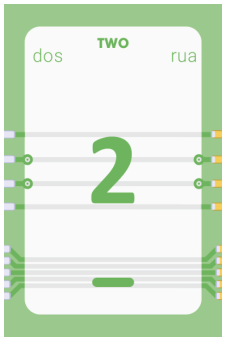
Key talking points:

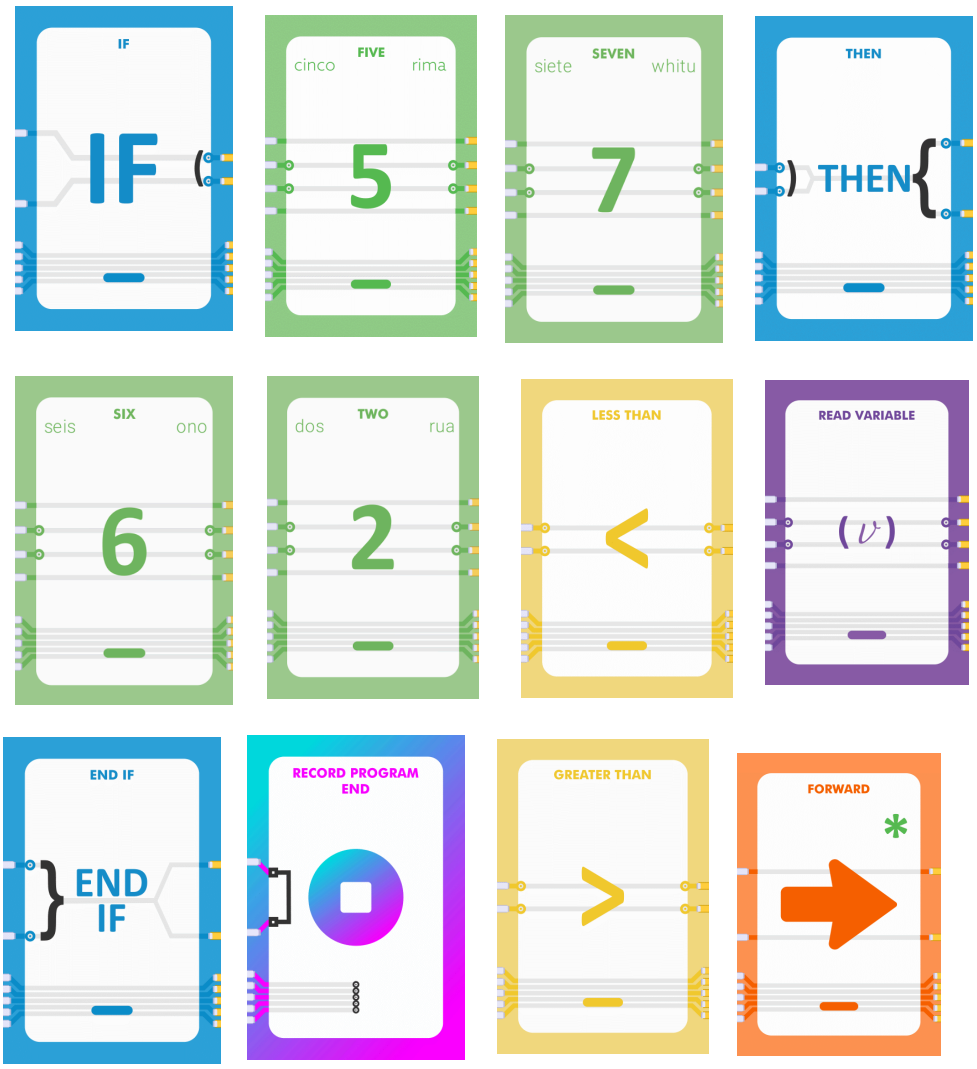
- These cards are used to determine if the code, as it is read by the computer, returns a 'true' or a 'false.'
- If the condition is 'true', then the computer continues to read the rest of the code.
- If the condition is 'false' the computer cannot continue reading the code.

Give students the following cards and ask them to try and put the cards into an algorithm (step-by-step) that the computer can read. After appropriate 'struggle', move on to the 'Explain' section.









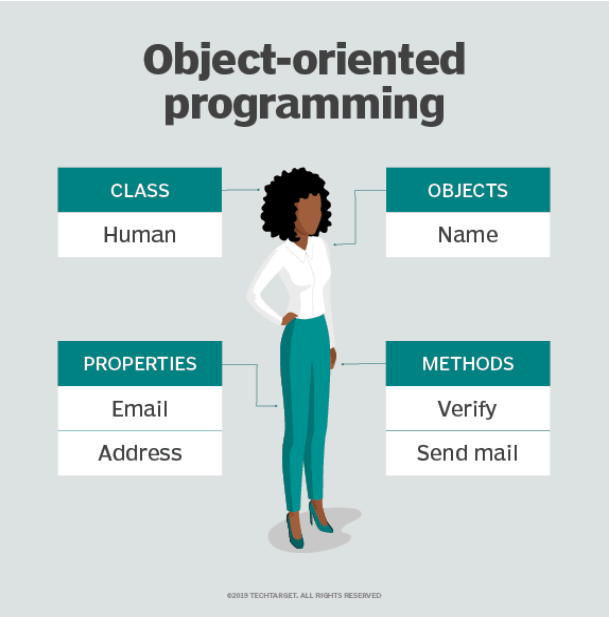
* Note: The 'END IF' card and a 'FORWARD' card have been added to the card deck you are giving students. Students have not yet been introduced to the 'END IF' card, so let them play and see if they can figure out where it might go.

Students should have seen and used the 'FORWARD' card in the previous lessons. The program needs something to 'do' once a true statement is achieved. This will be explained further in the next section.

Teachers should not expect students to correctly place the cards yet. This is about exploration...

Here is a google sheet template where students can copy and paste their coding cards in a google sheet. This may make it easier if they want to do this online.

*Teacher

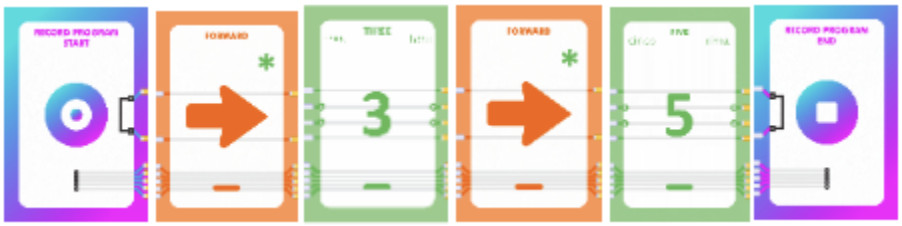


Knowledge/enrichment:

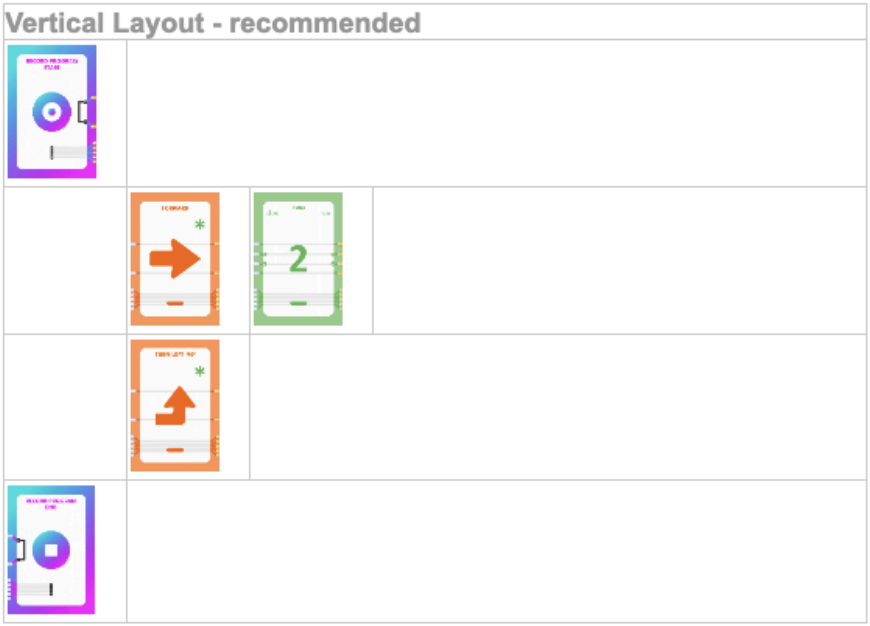
The colors of the cards are not essential knowledge at this age. If students inquire as to why the cards are different colors, it is because programmers most often use something called 'object-oriented programming.' Here is a diagram that represents a simplified version of the concept.

Explain:

Up to this point, students have been placing their programming cards in a linear fashion.

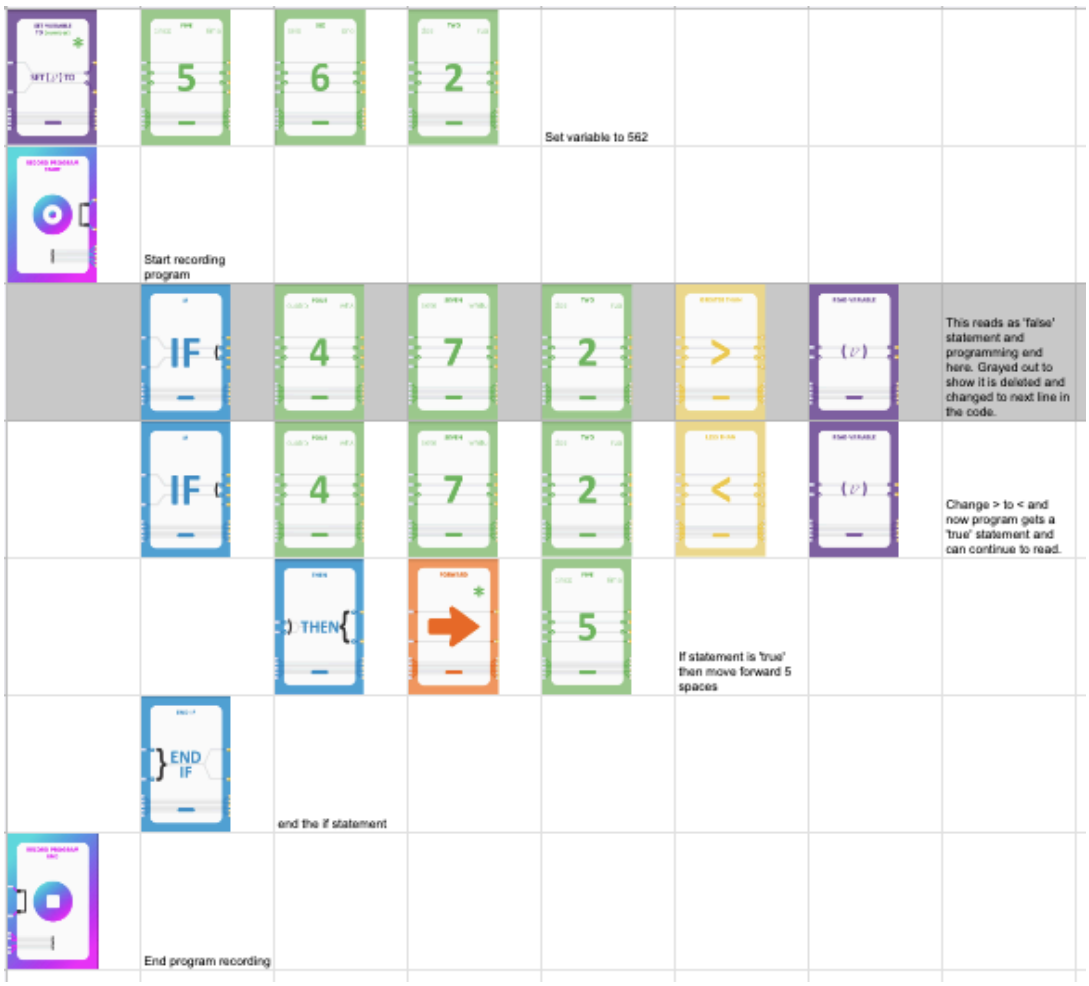


Second graders should now begin laying out their programming cards in a vertical fashion, with grouped elements indented.



This is much easier for anyone else to be able to read and understand the code and once the student progresses into text-based coding this then becomes natural for them. It does take up a lot of desk space and helps with the readability of the code.

Walk students through the creation of code by providing them with a grid organizer to place their cards on.



Click here to see the coding cards on Google Spreadsheet.

Key talking points:

- We must first set our variable, so the computer will know what the value of the number is.
- If the value of 472 is greater than 'v' = 562 (this produces a false statement. The computer program will stop reading the code lines.
- Change the > symbol to a < symbol now.
- If the value of 472 is less than 562 is now a true statement.
- Then move forward 5 spaces (the robot will move forward 5 spaces when the < symbol is used).
- End the if - then statement
- End the program

	Elaborate: Allow students to assign various numbers in the 'set variable' code and monitor for understanding of: • Correct use of < and > symbols • When the computer will get a 'false' statement and stop reading the code • When the computer will get a 'true' statement and move forward a given number of spaces.
Check for understanding:	**Evaluate: Inquiry questions** • Can you 'read' your algorithm (step-by-step directions)? • Why can't the computer use the 'alligator' method? • Why did you move from creating your code horizontally to vertically?