

Master 8a

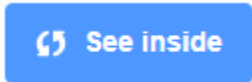
## Classifying Triangles by Side Lengths

### Using a Block-Coding Program

Follow the link to access the file: What type of triangle?

<https://scratch.mit.edu/projects/552987916/>

Click **See Inside**.

 See inside

The code for this application is incomplete.

#### Here is a description of the application:

- Cat asks the user to enter the number of EQUAL sides on the triangle.
- The application checks to make sure the user doesn't enter a value greater than 3. If a value greater than 3 is entered, it can't be a triangle.
  - If 0 is entered, then the triangle will be scalene.
  - If 2 is entered, then the triangle will be isosceles.
  - If 3 is entered, then the triangle will be equilateral.

**Master 8b**

# Classifying Triangles by Side Lengths (cont'd) Using a Block-Coding Program

Here is the code for the partially completed application:

```

when green flag clicked
  say "I will classify your triangle based on the side lengths." for 3 seconds
  ask "How many equal sides are in your triangle?" and wait
  set equalSides to answer
  if equalSides > 3 then
    say "That can't be a triangle - a triangle only has 3 sides in total." for 3 seconds
  else
    if equalSides = 0 then
      say "Scalene" for 2 seconds
    if equalSides = 2 then
      say "" for 2 seconds
    if [ ] then
      say "Equilateral" for 2 seconds
  
```

A yellow callout box points to the condition `equalSides > 3` with the text: "if the user enters a number greater than 3, then it can't be a triangle. Checks this first."

Master 8c

## Classifying Triangles by Side Lengths (cont'd) Using a Block-Coding Program

Your task is to complete the code for the application:  
Complete the **Say** block to indicate the type of triangle  
if the user enters 2 for number of equal sides.

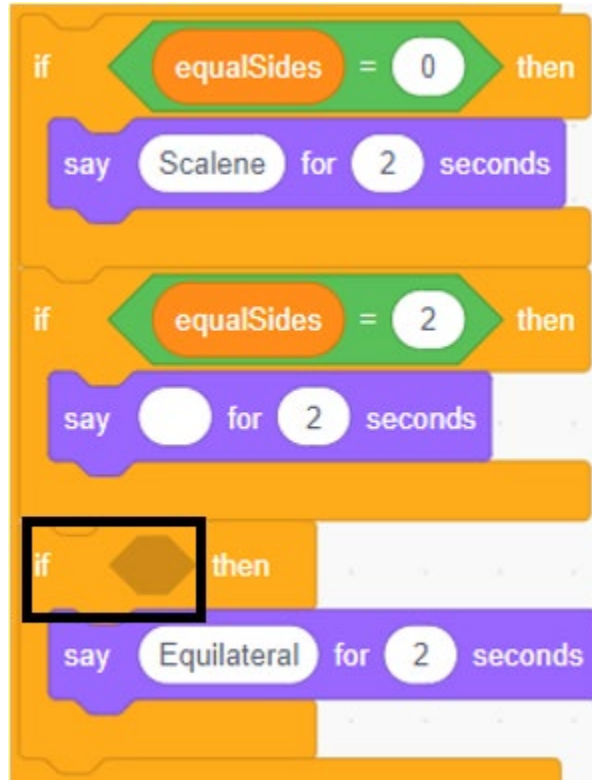
```
if (equalSides = 0) then
  say Scalene for 2 seconds
if (equalSides = 2) then
  say [ ] for 2 seconds
if ( ) then
  say Equilateral for 2 seconds
```

The image shows three Scratch 'if' blocks. The first block has a condition 'equalSides = 0' and a 'say Scalene for 2 seconds' block. The second block has a condition 'equalSides = 2' and a 'say' block with an empty text field, which is highlighted with a black box. The third block has an empty condition and a 'say Equilateral for 2 seconds' block.




Master 8d

## Classifying Triangles by Side Lengths (cont'd) Using a Block-Coding Program

Provide the correct condition for the equilateral triangle in the **If** statement shown:



**Tips:**

- The **conditional operator** blocks (  ) can be found under **Operators** (  ).
- The variable **equalSides** block can be found under **Variables** (  ).

Master 8e

# Classifying Triangles by Side Lengths (cont'd) Using a Block-Coding Program

Follow the link to access the file: What type of triangle?

<https://scratch.mit.edu/projects/552694138/>

Test out the application by executing the code.

Click the green flag.

What do think this application does?

The image shows a Scratch script for classifying triangles based on the greatest angle. The script starts with a 'when clicked' event, followed by a 'say Hello! for 2 seconds' block, and an 'ask What is the greatest angle in your triangle? and wait' block. The user's answer is stored in a variable named 'angle'. A conditional block checks if the angle is greater than 180 or less than 1. If true, it says 'This can't be a triangle. for 3 seconds'. Otherwise, it checks if the angle is greater than 90. If true, it says 'Your triangle is an obtuse triangle. for 3 seconds'. If false, it checks if the angle is less than 91 or greater than 59. If true, it says 'This is NOT an obtuse triangle. for 3 seconds'. If false, it checks if the angle is less than 60. If true, it says 'Check again. This can't be the greatest angle. for 2 seconds'. A yellow callout box points to the first conditional block with the text: 'Checks to make sure it's a triangle before determining the type of triangle.'

## **Classifying Triangles by Side Lengths (cont'd) Using a Block-Coding Program**

Currently, the application shows if the triangle is obtuse or not obtuse, based on the measure of the greatest interior angle.

The greatest interior angle must be greater than a certain measure for the triangle to be considered obtuse. What is that measure?

How does the application know that the user has entered a measure that can't possibly be the greatest interior angle?

Consider what the interior angles in a triangle add to.

Try to make sense of the program by testing different measures to see what happens.

Master 8g

# Classifying Triangles by Side Lengths (cont'd) Using a Block-Coding Program

Follow the link to access the file: What type of triangle?

<https://scratch.mit.edu/projects/552997968/>

Notice that additional conditional statements (**If ... then**) have now been added, but the program is incomplete.

The image shows a Scratch script with the following blocks:

- when green flag clicked
- say Hello! for 2 seconds
- ask What is the greatest angle in your triangle? and wait
- set angle to answer
- if  $\text{angle} > 180$  or  $\text{angle} < 1$  then
  - say This can't be a triangle. for 3 seconds
- else
  - if  $\text{angle} > 90$  then
    - say [ ] for 3 seconds
  - if [ ] then
    - say Your triangle is a right triangle. for 2 seconds
  - if  $\text{angle} < 90$  or  $\text{angle} > 59$  then
    - say Your triangle is an acute triangle. for 2 seconds
  - if  $\text{angle} < 60$  then
    - say Check again. This can't be the greatest angle. for 2 seconds

A yellow callout box points to the first 'if' block with the text: "Checks to make sure it's a triangle before determining the type of triangle."

Master 8h

## Classifying Triangles by Side Lengths (cont'd) Using a Block-Coding Program

Your task is to complete the code for the application:

```

if (angle > 90) then
  say [ ] for 3 seconds
endif

if ( ) then
  say "Your triangle is a right triangle." for 2 seconds
endif

if (angle < 90 or angle > 59) then
  say "Your triangle is an acute triangle." for 2 seconds
endif

if (angle < 60) then
  say "Check again. This can't be the greatest angle." for 2 seconds
endif
  
```

Complete the **Say** block to indicate the type of triangle if the user enters a measure greater than 90 for the greatest interior angle:



Master 8i

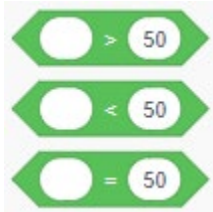


## Classifying Triangles by Side Lengths (cont'd) Using a Block-Coding Program

Provide the correct condition for the Right triangle in the **If** statement shown:

The image shows a Scratch script with four 'if-then' blocks:

- Block 1: **if**  $\text{angle} > 90$  **then** **say** [ ] **for** 3 **seconds**
- Block 2: **if** [ ] **then** **say** "Your triangle is a right triangle." **for** 2 **seconds**
- Block 3: **if**  $\text{angle} < 90$  **or**  $\text{angle} > 59$  **then** **say** "Your triangle is an acute triangle." **for** 2 **seconds**
- Block 4: **if**  $\text{angle} < 60$  **then** **say** "Check again. This can't be the greatest angle." **for** 2 **seconds**

**Tips:**

- The **conditional operator** blocks (  ) can be found under **Operators** (  ).
- The variable **equalSides** block can be found under **Variables** (  ).

## Classifying Triangles by Side Lengths (cont'd) Using a Block-Coding Program

### Challenge A


Alter the code in the *Classifying Triangles based on Angles* program (<https://scratch.mit.edu/projects/552694138/>) so that it determines the sum of the other two angles.

### Tips:

- You will need to use an operator variable to subtract the measure of the greatest angle from 180 to determine what is left for the other two angles.

- The operators blocks (  ) are located under

**Operators** (  Operators ).

- The variable angle block can be found under **Variables** (  Variables ).

- You can output this information using a **Say** block,

found under **Looks** (  Looks ).

Master 8k

## Classifying Triangles by Side Lengths (cont'd) Using a Block-Coding Program

### Challenge B

Alter the code to draw the type of triangle (scalene, isosceles, or equilateral) based on the number of equal sides entered by the user for the Classifying Triangles by Side Length application.

(<https://scratch.mit.edu/projects/552987916/>)

It might take a few tries to draw isosceles and scalene triangles so that they are closed completely.

### Challenge C

Alter the code to draw each type of triangle (right, obtuse, or acute) based on the angle entered by the user for the *Classifying Triangles by Angle* application (<https://scratch.mit.edu/projects/552997968/>).

It might take a few tries to draw the right, obtuse and acute triangles so that they are closed completely. You do not have to draw the exact triangles based on the greatest angle measures, but be sure to draw examples of each type of triangle.

### Self-check in

What have you learned about conditions so far?

Did you get stuck? If so, what did you do?

Did you turn to your classmates for help? If so, how did they help?

What are you doing to help the learning of others?

This is “hard fun.” What do you think we mean by “hard fun”?

What other activities do you do that are “hard fun”?

Go on “spy walks” to see what your classmates have done.