iRobot

iRobot Coding Help Guide

Turning Root On/Off

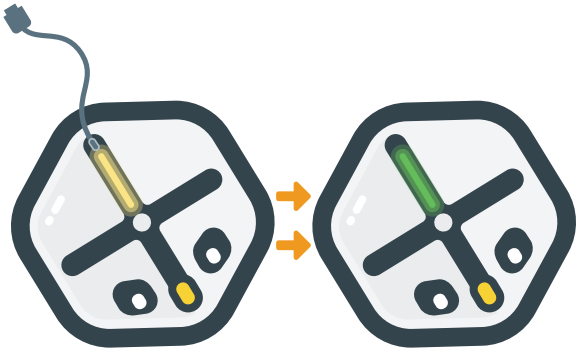
Firmly press and hold Root's nose to turn Root on. Do the same to turn Root off.

Hard Reset

If Root is not responding as expected, hold Root's nose for 10 seconds.

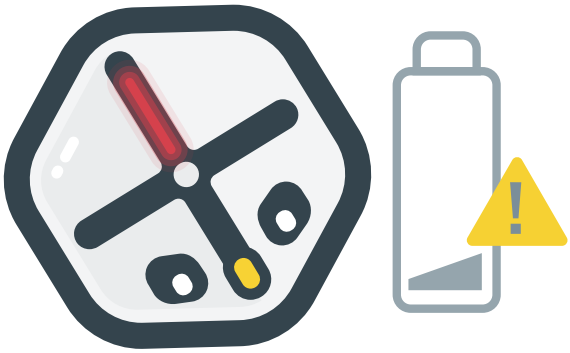
Charging

Plug in a charging cable to start charging. Root will flash yellow while charging and light up green when done.



Low Battery Warning

Root flashes red when the battery gets low. Plug Root in to recharge.



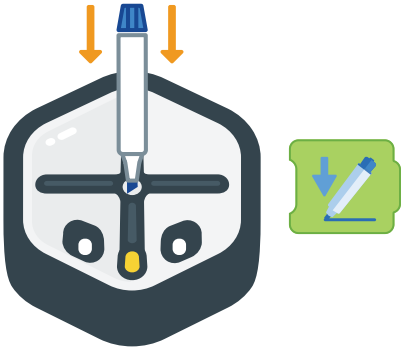
iRobot®

Decorating Root

The top of Root is a whiteboard. Try decorating your robot with whiteboard markers and clings!

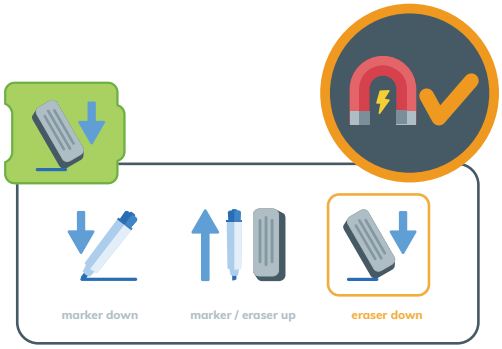
Drawing with Root

To draw, firmly place a marker in Root's marker holder. Use the Marker Block to lift and drop the marker.



Erasing with Root

Edit the Marker Block to lower the eraser. The eraser only works on magnetic whiteboard surfaces.



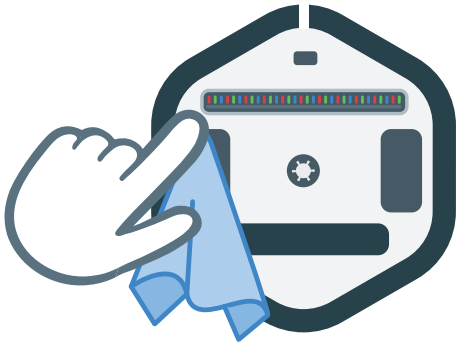
Cleaning the Eraser

To clean, peel off the eraser pad and wash it with soap and water.

iRobot®

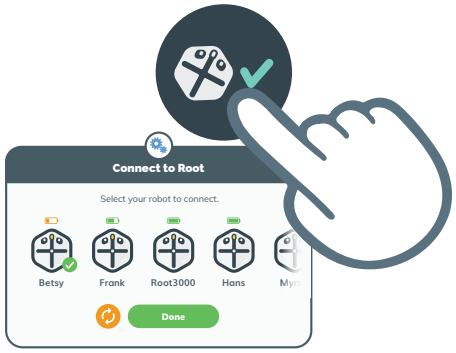
Cleaning the Wheels

If Root's wheels get dirty, wipe them clean with a cloth.



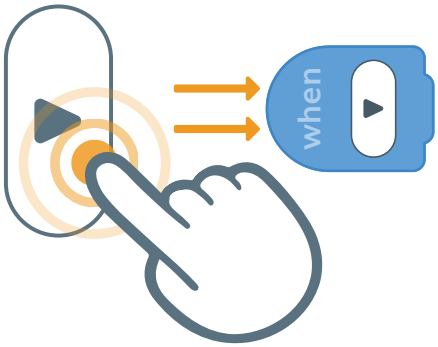
Connecting to Root

To connect to a robot, touch the Root icon and select your robot.



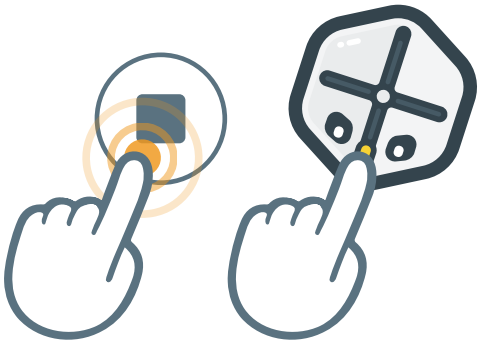
Running a Project

Press the Play button on the left side of the screen to run your project. Every project starts with a When Play block.



Stopping a Project

To stop, press Root's nose or touch the Stop button on the left side of the screen.



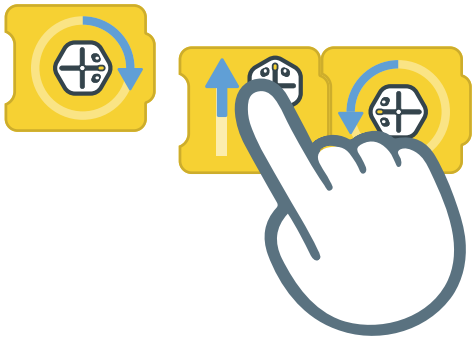
iRobot®

Deleting Blocks

When dragging blocks, a trash can appears. Drag blocks over the trash can to delete.

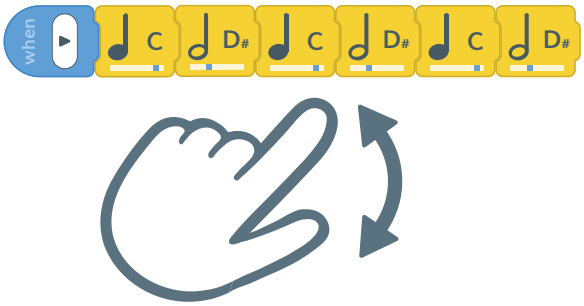
Dragging Multiple Blocks

Press and hold a block to drag all connected blocks to the right.



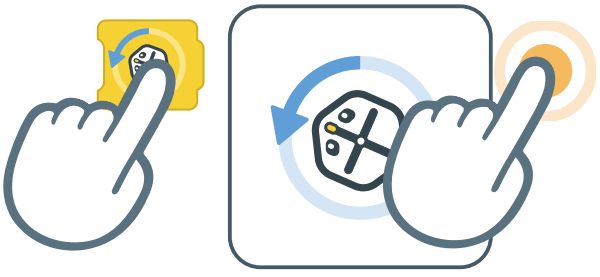
Zooming in the Coding Screen

Pinch to zoom in and out so you can better see your code.



Opening and Closing Editors

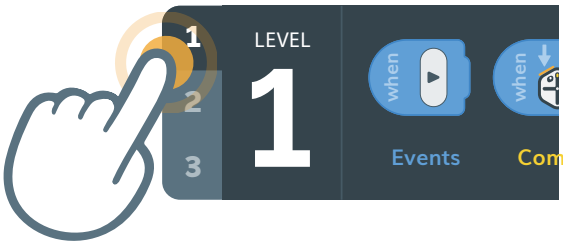
Tap on each block to see the editor. Tap anywhere off the editor window to close it.



iRobot®

Changing Code Level

Tap on 1, 2, or 3 to change the coding level. A warning will appear if your code can not be translated to that level.



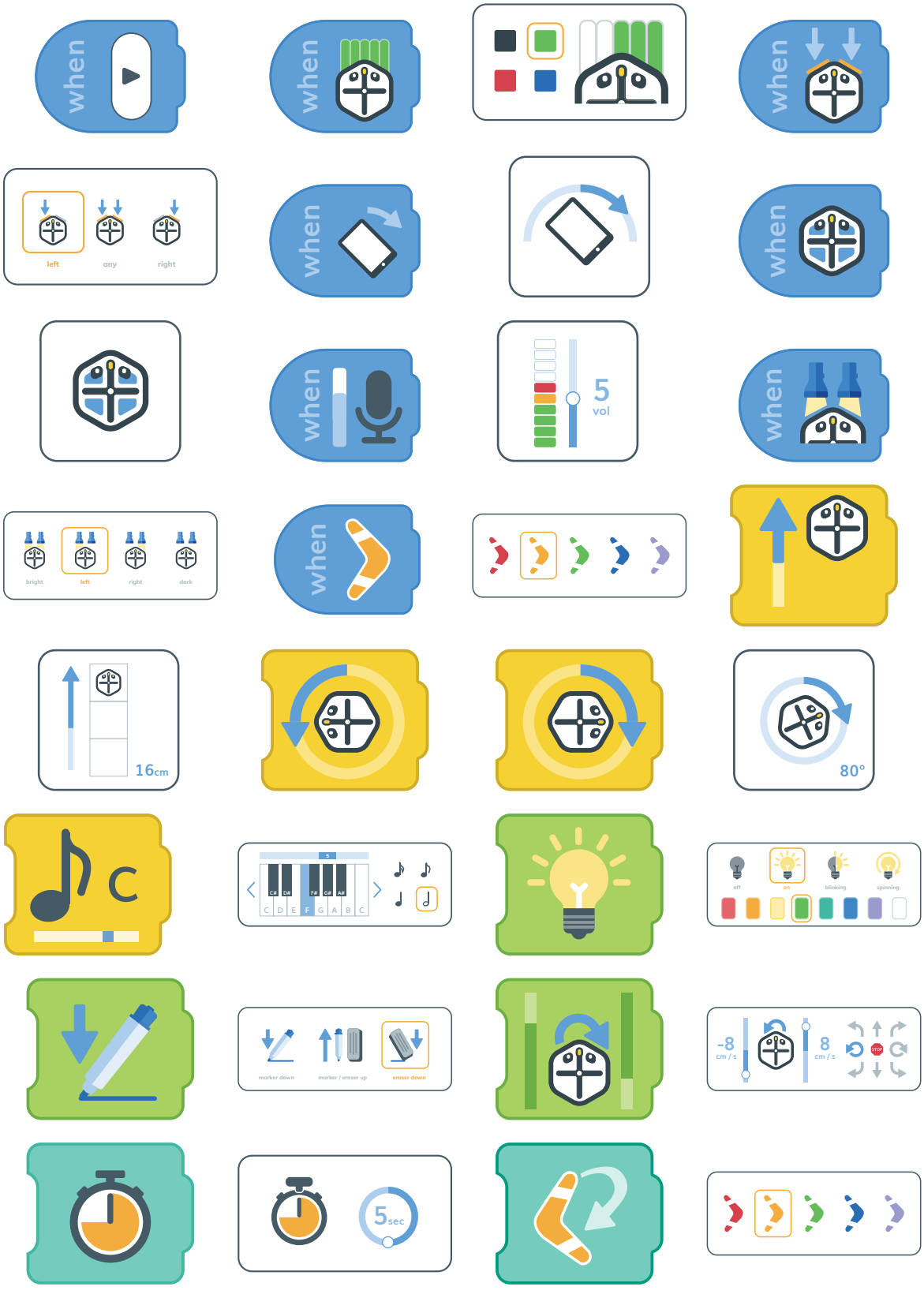
Block Glossary

Drag any block to Square's icon to learn more about what it does.

iRobot®

iRobot

Level 1 Block Glossary



When Play Block

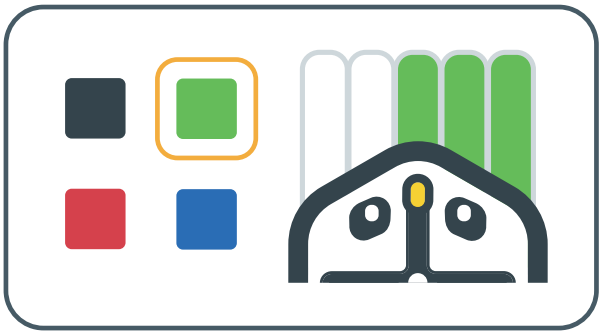
Code after the When Play Block will run as soon as you press the Play button.

When Color Sensor Block

Use the When Color Sensor Block to code Root to sense and respond to a color you have chosen.

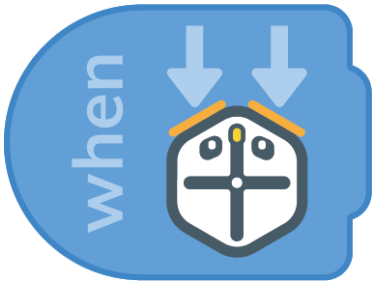
When Color Sensor Block Editor

Tap on each zone and pick the colors that Root responds to.



When Bump Block

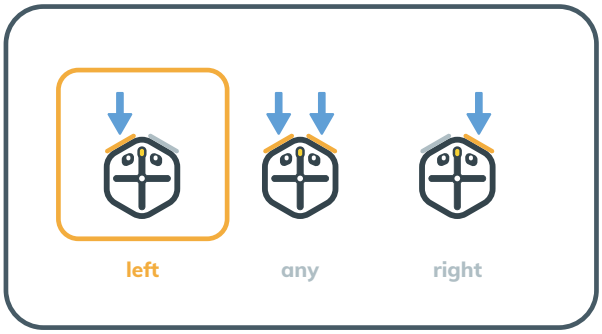
The When Bump Block tells Root to respond when its bump sensors are pressed.



iRobot®

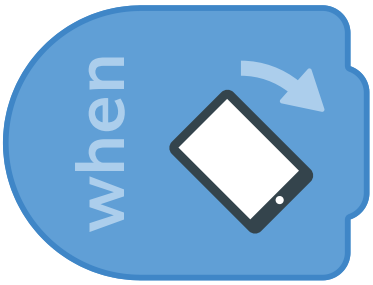
When Bump Block Editor

Use the editor to change which bumpers Root responds to when activated.



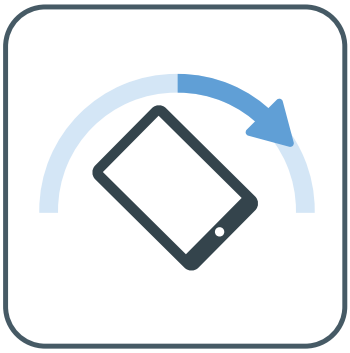
When Tilt Block

With the When Tilt Block, you can code Root to respond when you tilt your device in different directions.



When Tilt Block Editor

Drag the arrow to select the tilt position you'd like Root to respond to.



When Touch Block

With the When Touch Block, code Root to respond when any one of the four zones on top of Root is pressed.

iRobot®

When Touch Block Editor

Use the editor to change the zones Root responds to when activated.

When Sound Block

Use the When Sound Block to code Root to respond to differences in volume.

When Sound Block Editor

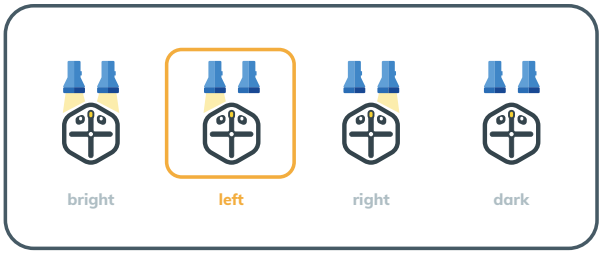
Use the editor to change the volume Root responds to.

When Light Block

Use the When Light Block to code Root to respond to changes in light.

When Light Block Editor

Root has two light sensors that can respond to changes in brightness.

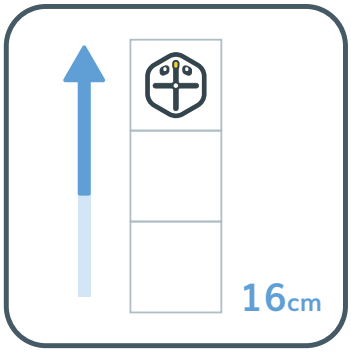


Move Block

Use the Move Block to code Root to move forward or backward. You can also set the distance Root will move.

Move Block Editor

Drag Root to change how far forward or backward Root moves.



Turn Block

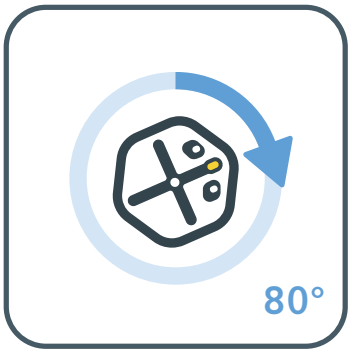
This Turn Block lets you code Root to turn counter-clockwise.

Turn Block

This Turn Block lets you code Root to turn clockwise.

Turn Block Editor

Drag the arrow to change the angle Root turns.



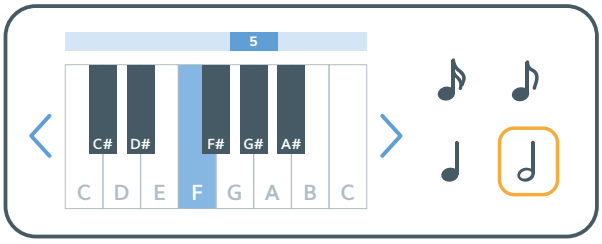
iRobot®

Music Block

Use the Music Block to code Root to play a note. Change the notes and their durations to create a song.

Music Block Editor

Use the editor to select the tone and length of the note. Tap on the left and right arrows to change the octave.

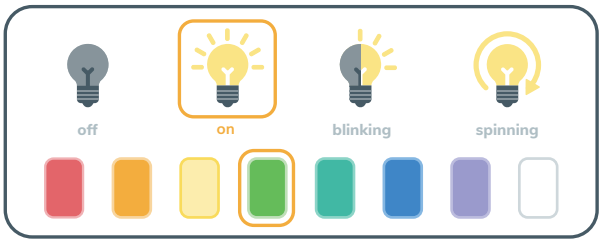


Light Block

The Light Block can be used to turn on the lights on top of Root and change their color and blinking pattern.

Light Block Editor

Use the editor to change the color and pattern of Root's lights.



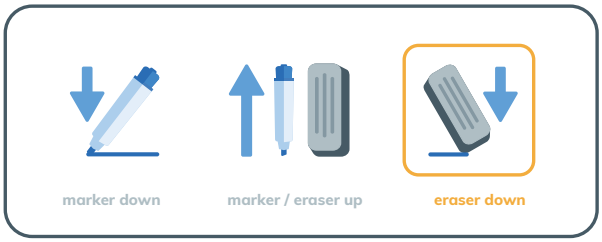
iRobot®

Marker Block

The Marker Block can be used to raise and lower Root's marker and eraser.

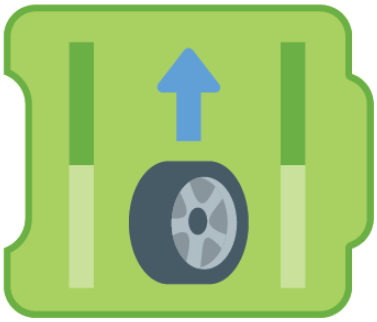
Marker Block Editor

Use the editor to lift and lower the marker and eraser.



Wheel Speeds Block

Use the Wheel Speeds Block to change how fast Root's wheels move and how wide Root turns.



Wheel Speeds Block Editor

Change the speed and direction of each wheel directly or select common actions on the right, such as rotate in place or turn in an arc.

iRobot®

Wait Block

The Wait Block lets you set a specific amount of time to delay before going on to the next block.

Wait Block Editor

Move the dial to change the wait time.

Repeat Block

The Repeat Block tells the code inside to play over and over.

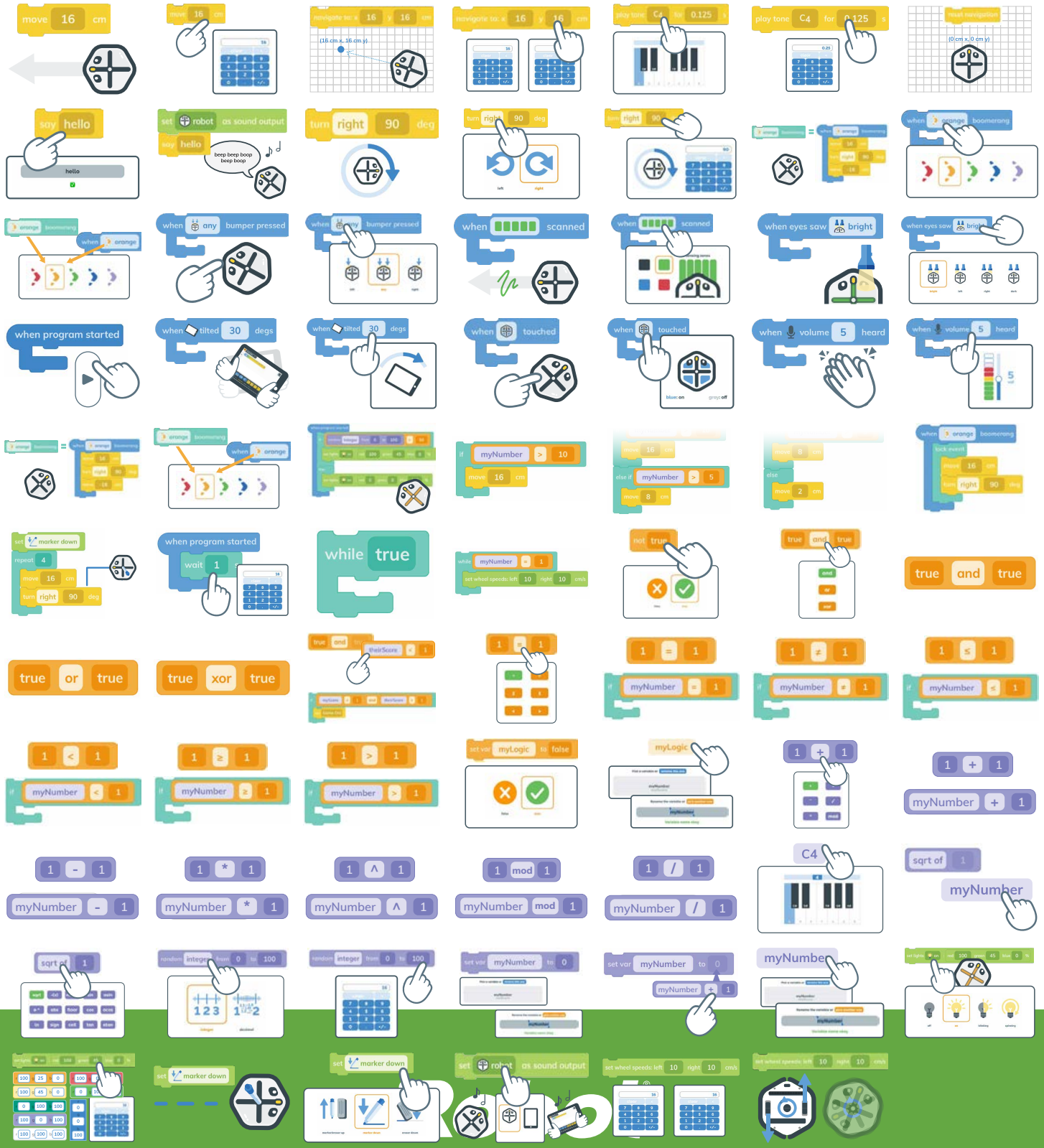
Repeat Block Editor

Edit the Repeat Block to tell your project how many times to play the code blocks inside.

iRobot®

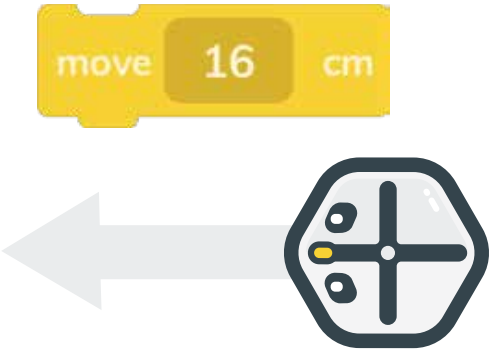
iRobot

Level 2 Block Glossary



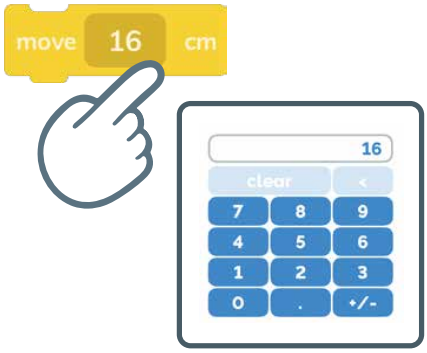
Move Block

Use the Move Block to code Root to move forward or backward in centimeters.



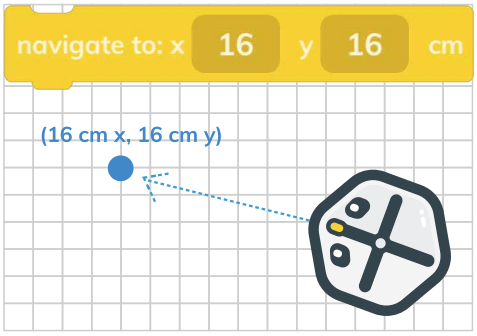
Edit Move Block

Use the move editor to tell Root how many centimeters to move forward or backward.



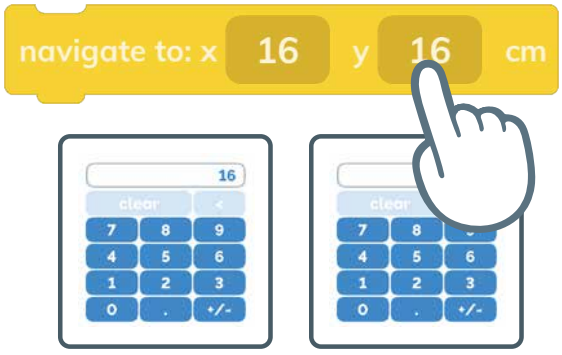
Navigate Block

Root navigates across an invisible grid, with an origin (0 cm, 0 cm) set to Root's starting position.



Edit Navigate Block

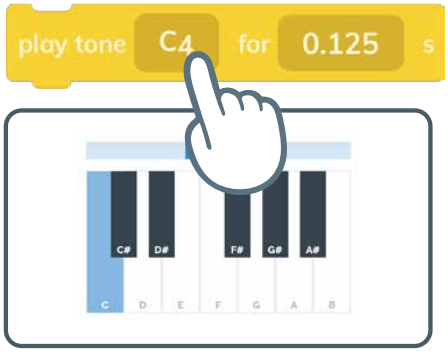
You can tell Root to move to a specific coordinate by editing the Navigate Block.



iRobot®

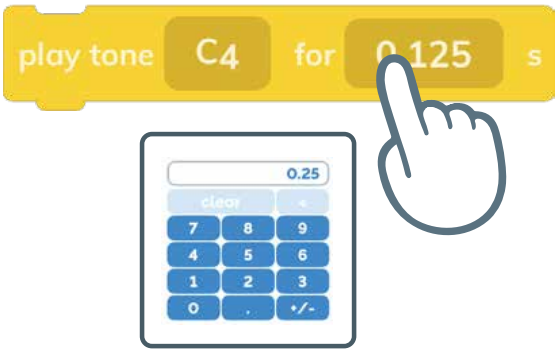
Music Block

The Music Block plays a music note. Open the first editor to choose which note to play.



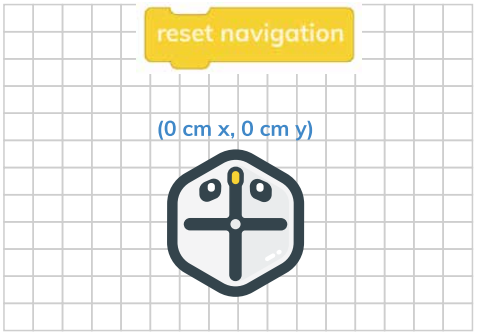
Edit Music Block

Open the second editor to tell Root how may seconds to play your note.



Reset Navigation Block

The Reset Navigation Block resets Root's invisible grid origin (0 cm, 0 cm) to Root's current location.



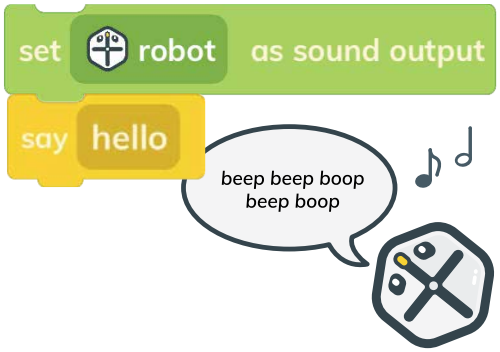
Say Block

Use the Say Block to code Root or your device to say something out loud.

iRobot®

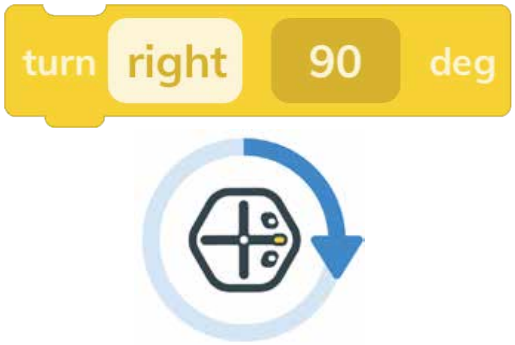
Root Language

If you code Root to speak, Root will translate the words into Root Language.



Turn Block

The Turn Blocks let you code Root to turn left or right.



Edit Turn Block

Use the first editor to tell Root to turn left or right.

Edit Turn Block

Use the second editor to tell Root how far to turn in degrees.

iRobot®

Arc Block

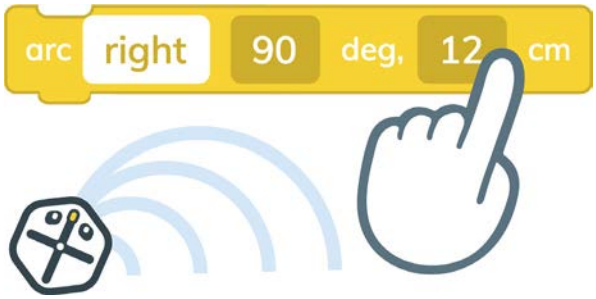
The Arc Block tells Root to drive along the shape of a circle.

Arc Block Editor

The Arc Block's degrees tells your robot how far to drive around the circle.

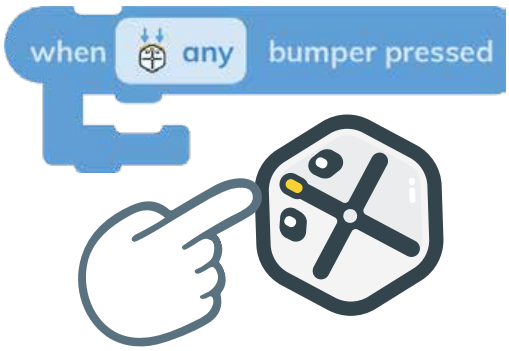
Arc Block Editor

The Arc Block's radius tells your robot how wide your circle should be.



When Bump Block

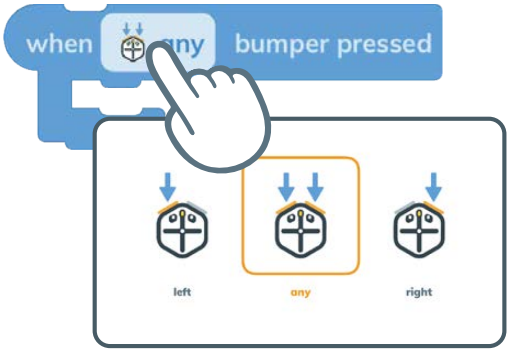
The When Bump Block will run when Root's matching bump sensors are pressed.



iRobot®

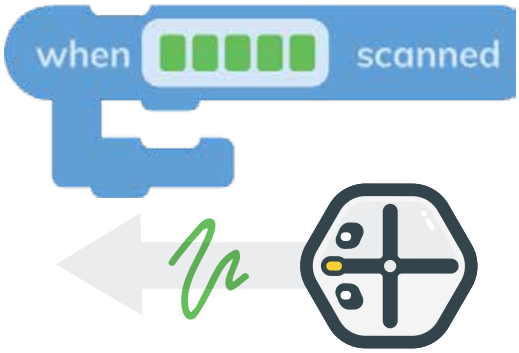
Bump Editor

Use the editor to change which bumpers Root responds to when activated.



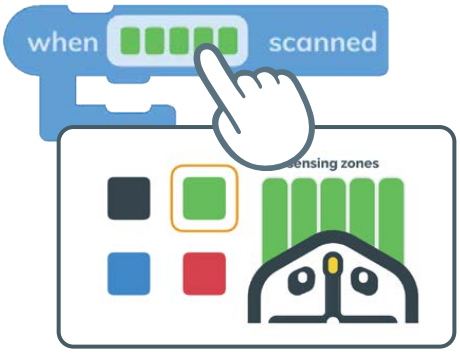
When Color Sensor Block

Use the When Color Sensor Block to code Root to sense and respond to a color you have chosen.



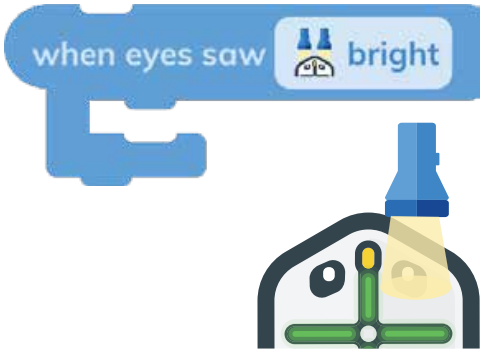
Color Sensor Editor

Open the editor to tell Root which color to sense and which zones to look at.



When Light Block

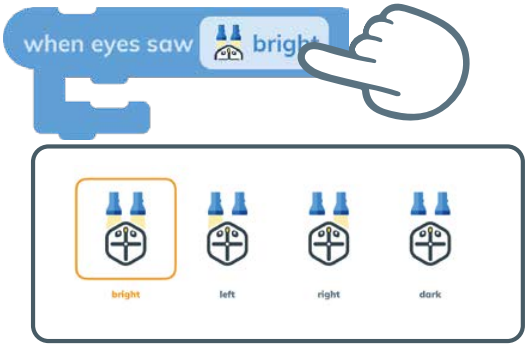
Use the When Light Block to code Root to respond to changes in light.



iRobot®

Light Editor

Root has two light sensors that can respond to changes in brightness.



When Start Block

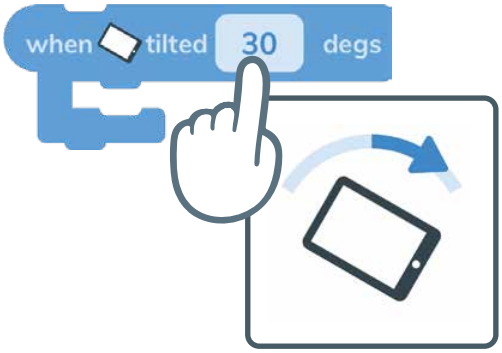
Code after the When Start Block will run as soon as you tap the Start button.

When Tilt Block

With the When Tilt Block, you can code Root to respond when you tilt your device in different directions.

Tilt Editor

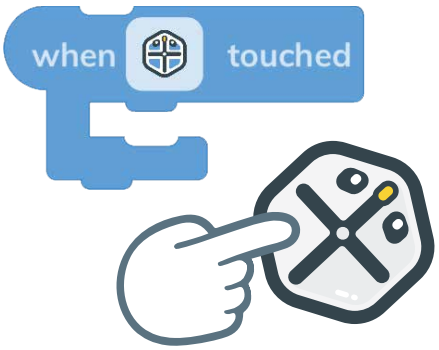
Drag the arrow to select the tilt position you'd like Root to respond to.



iRobot®

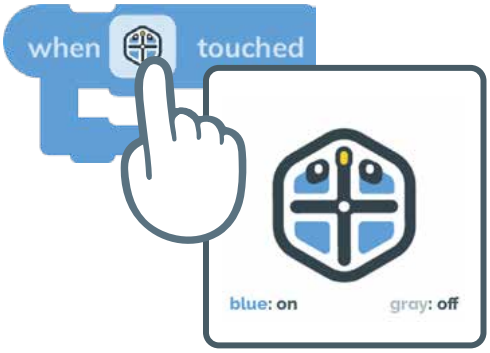
When Touch Block

With the When Touch Block, code Root to respond when one of the four zones on top of Root is pressed.



Touch Editor

Use the editor to change the zones Root responds to when touched.

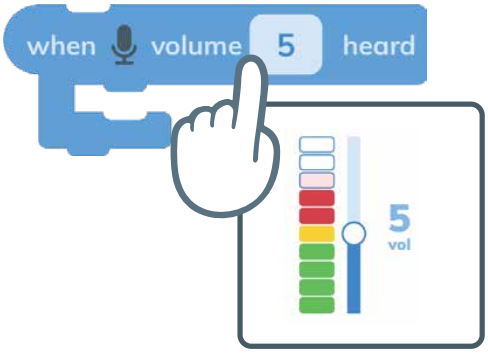


When Sound Block

Use the When Sound Block to code Root to respond to differences in volume.

Sound Editor

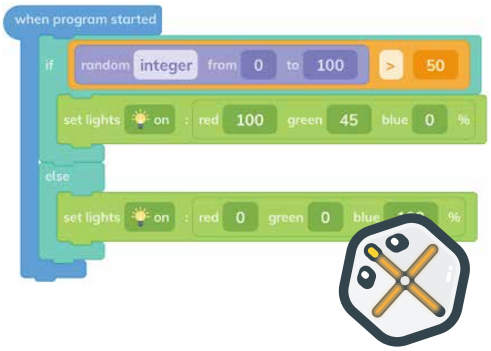
Use the editor to change the volume Root responds to.



iRobot®

Conditionals

Conditional Blocks can be used to tell Root what to do IF something is true or false.



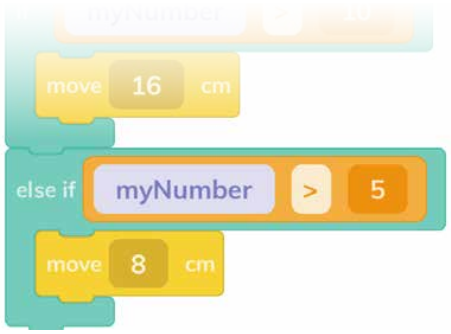
If

Root will follow the code inside this block IF the expression inside is true. If it is false, the program will ignore this block and move to the next one.

iRobot®

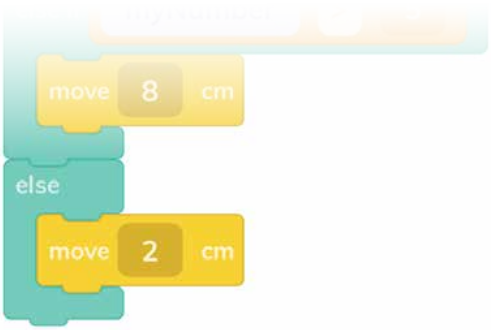
Else If

If the expression inside the If Block is false, the program will read the Else If Block next. If its expression is true, its code will run. If not, the program will move to the next block.



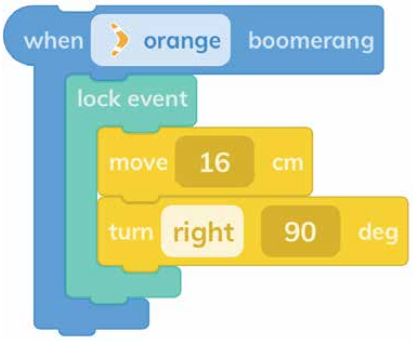
Else

If the expressions inside the If Block and all Else If Blocks above are false, then the code inside the Else Block will run.



Lock Event Block

When Root's program reaches a Lock Event Block, it will run through the code inside completely without interruption by other events.



Repeat Block

The Repeat Block is used to create a loop of code that runs over and over a certain amount of times.

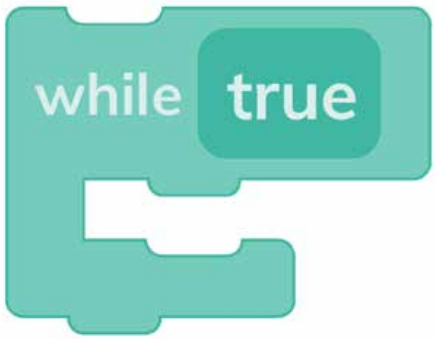
iRobot®

Wait Block

The Wait Block sets an amount of time to delay before going on to the next block.

While Block

The While Block can be filled with a true or false expression.

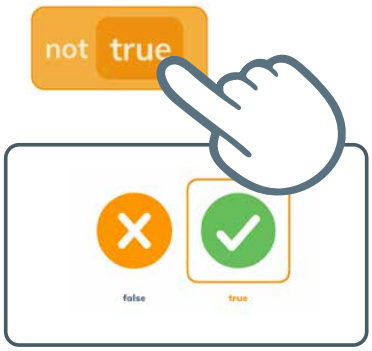


While Rule

If the expression inside the While Block is true, then its code will keep repeating until the expression becomes false.

Not Block

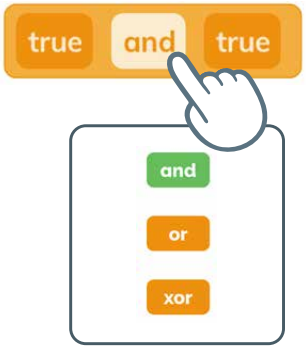
The Not Block flips its expression to its opposite, making things that are true into false and false into true.



iRobot®

Double Operator Block

The Double Operator Blocks will be either true or false depending on the state of their contents



And

The And Block returns true if all the expressions in the block are true.

Or

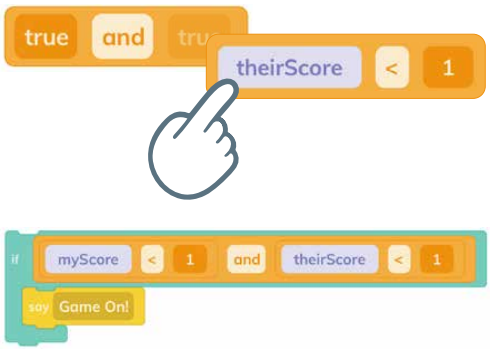
The Or Block returns true if either of the expressions in the block are true.

Exclusive Or

The Exclusive Or Block returns true if only one (and not both) of the expressions in the block is true.

Using Conditionals

You can use Double Operator Blocks with conditionals to tell Root what to do if the expressions inside are true or false.

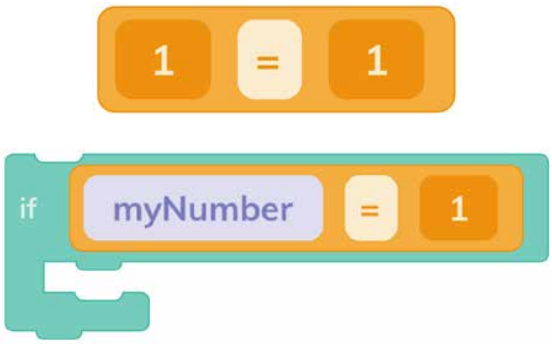


Comparison Block

A Comparison Block will return either true or false depending on the relationship between its contents

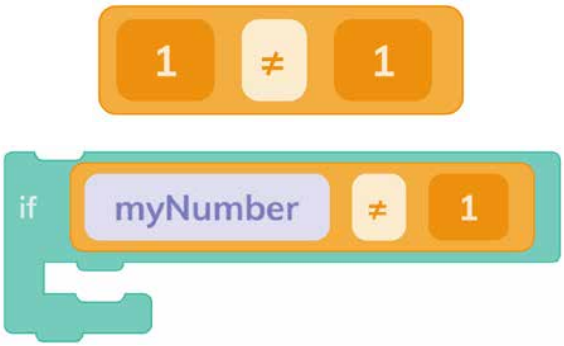
Equal

The Equal Block returns true if the first value is equal to the second value.



Not Equal

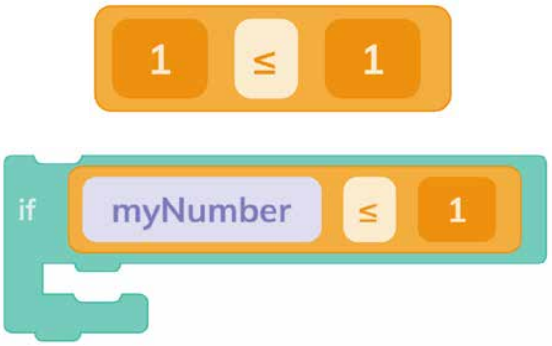
The Not Equal Block returns true if the first value is not equal to the second value.



iRobot®

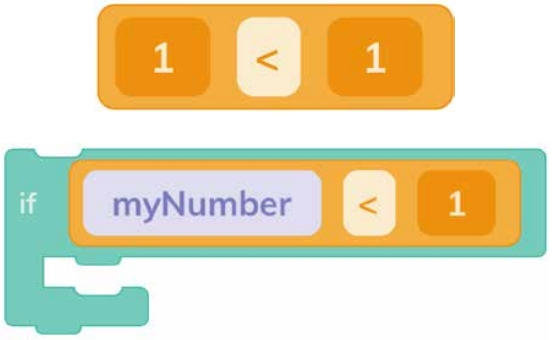
Less Than or Equal To

The Less Than or Equal to Block returns true if the first value is less than or equal to the second value.



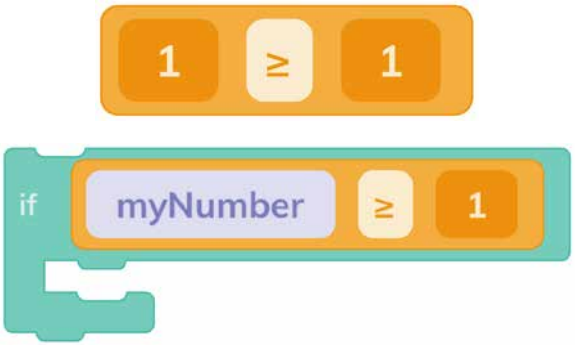
Less Than

The Less Than Block returns true if the first value is less than the second value.



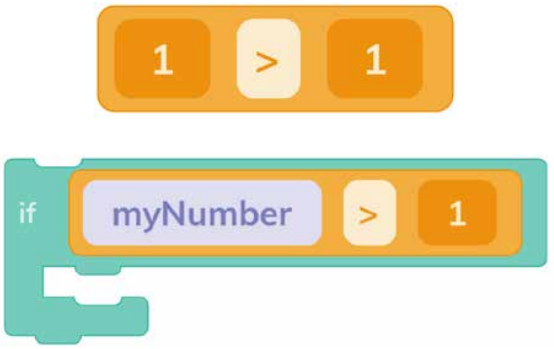
Greater Than or Equal To

The Greater Than or Equal to Block returns true if the first value is greater than or equal to the second value.



Greater Than

The Greater Than Block returns true if the first value is greater than the second value.

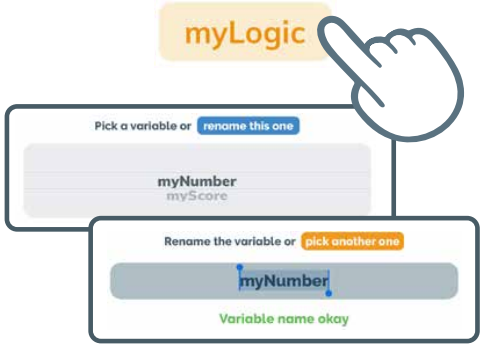


Set Boolean Variable

You can change your Logic Variable to true or false when something happens, like if a bumper is tapped.

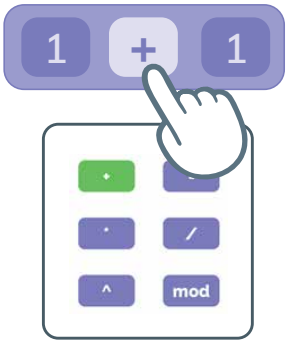
Boolean Variable

Logic Variables can switch between true or false to help Root keep track of what's happening in the project.



Equation Block

The Math Operators Blocks return a result of the selected math operation between its contents.



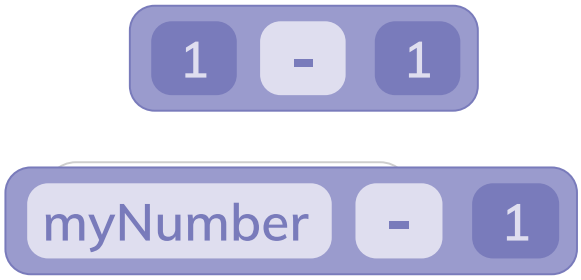
Addition

The Addition Block equals the sum of two values.

iRobot®

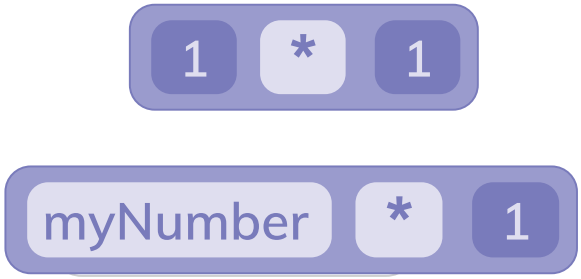
Subraction

The Subtraction Block equals the difference between two values.



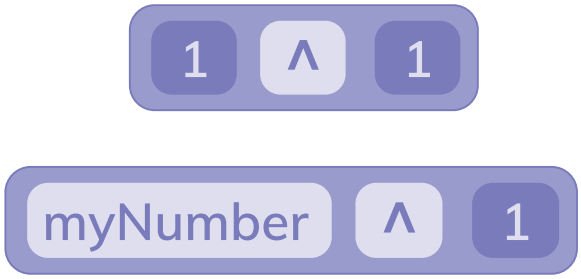
Multiplication

The Multiplication Block equals the product of two values.



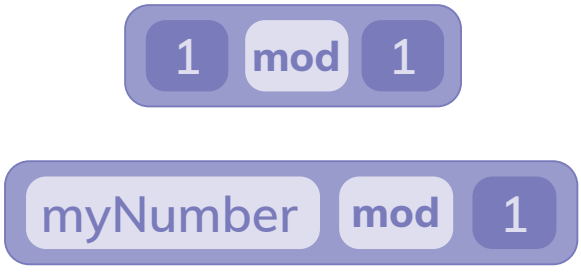
Exponent

The Exponentiation Block equals the first value raised to the power of the second value.



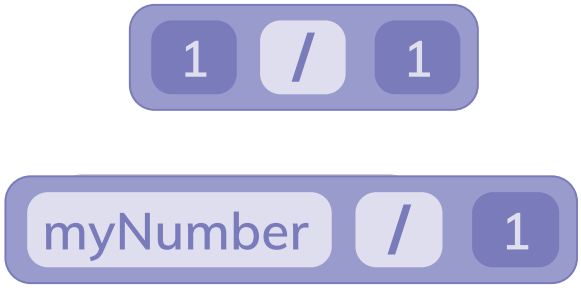
Modulo

The Modulo Block equals the remainder when the first value is divided by the second value.



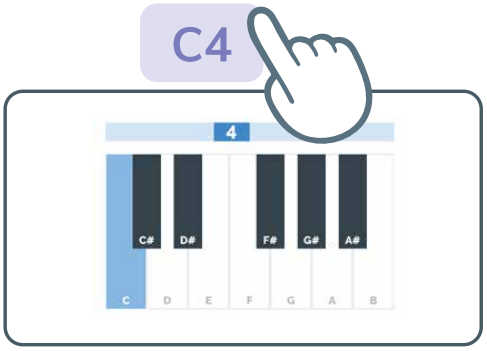
Division

The Division Block equals the quotient of the first value divided by the second value.



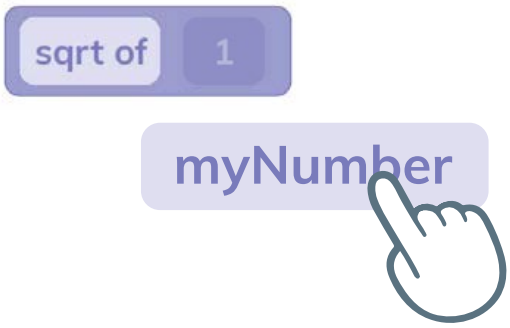
Music Frequency Block

The Music Freqency Block equals the selected note's freqency in Hertz.



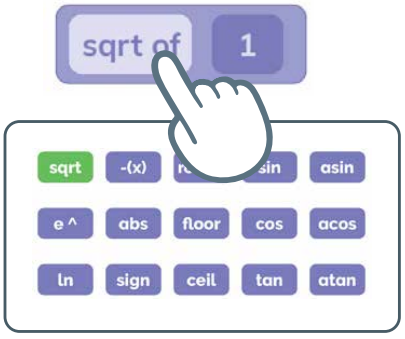
Operation Block

The Operation Block becomes the solution of the operation performed on the value inside.



Operation Editor

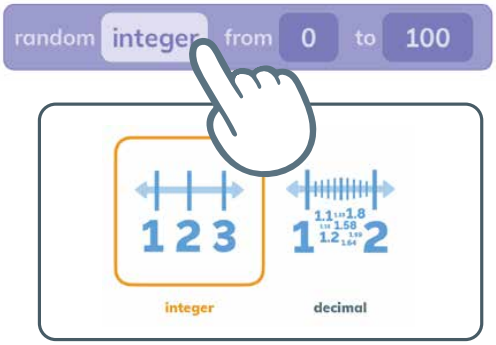
You can choose which math operation to perform by opening the Math Operation Editor.



iRobot®

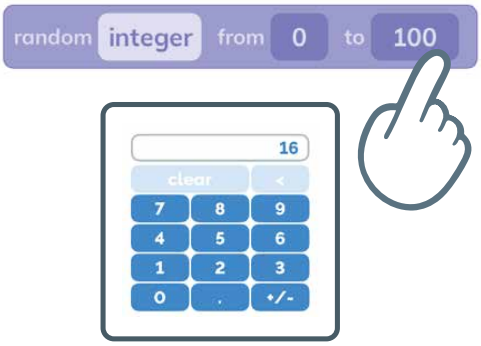
Random Block

The Random Block produces any integer or decimal between the two values inside.



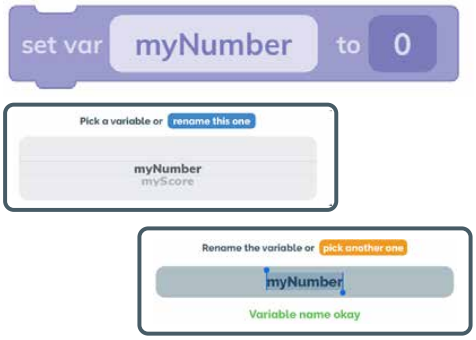
Random Range

You can change the range of the Random Block with the two number editors.



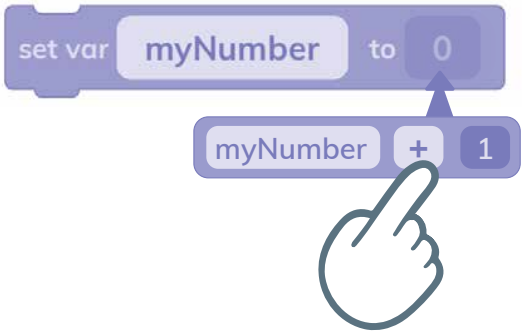
Set Variable Block

The Set Variable Block tells the Number Variable Block what number to hold.



Keeping Score

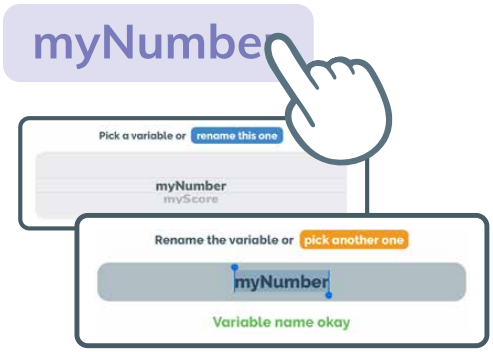
The Set Variable Block can hold a number or a result of a math operation, like an Addition Block that adds one point to the score.



iRobot®

Variable Block

Number Variables hold values to help Root keep track of what's happening in the project.



Light Block

The Light Block sets the color and blinking pattern for the lights on top of Root.

Light Editor

All light colors are a mix of Red, Green, and Blue light. Create different colors by setting the R, G, B values to different numbers.

Marker Block

The Marker Block can be used to raise and lower Root's marker and eraser.

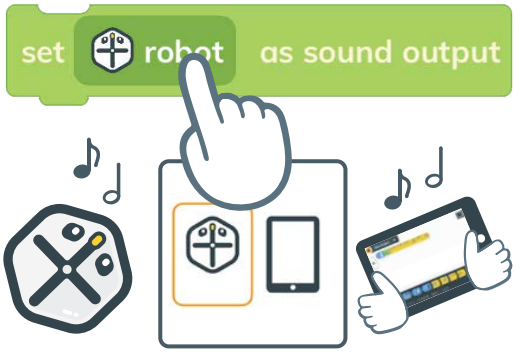
iRobot®

Marker Editor

Use the editor to select desired the marker and eraser posistion.

Set Sound Output Block

The Set Sound Output Block tells your code to play sounds through Root or on your device.

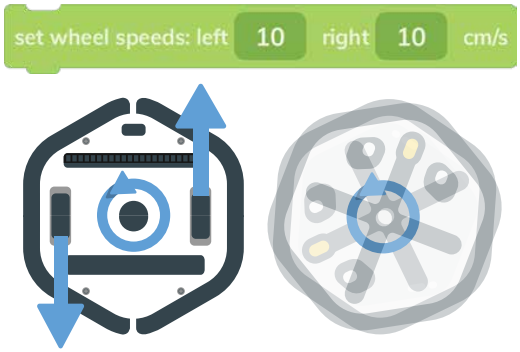


Wheel Speeds Block

Use the Wheel Speeds Block to change how fast Root's wheels move and how wide Root turns.

Wheel Speeds Editor

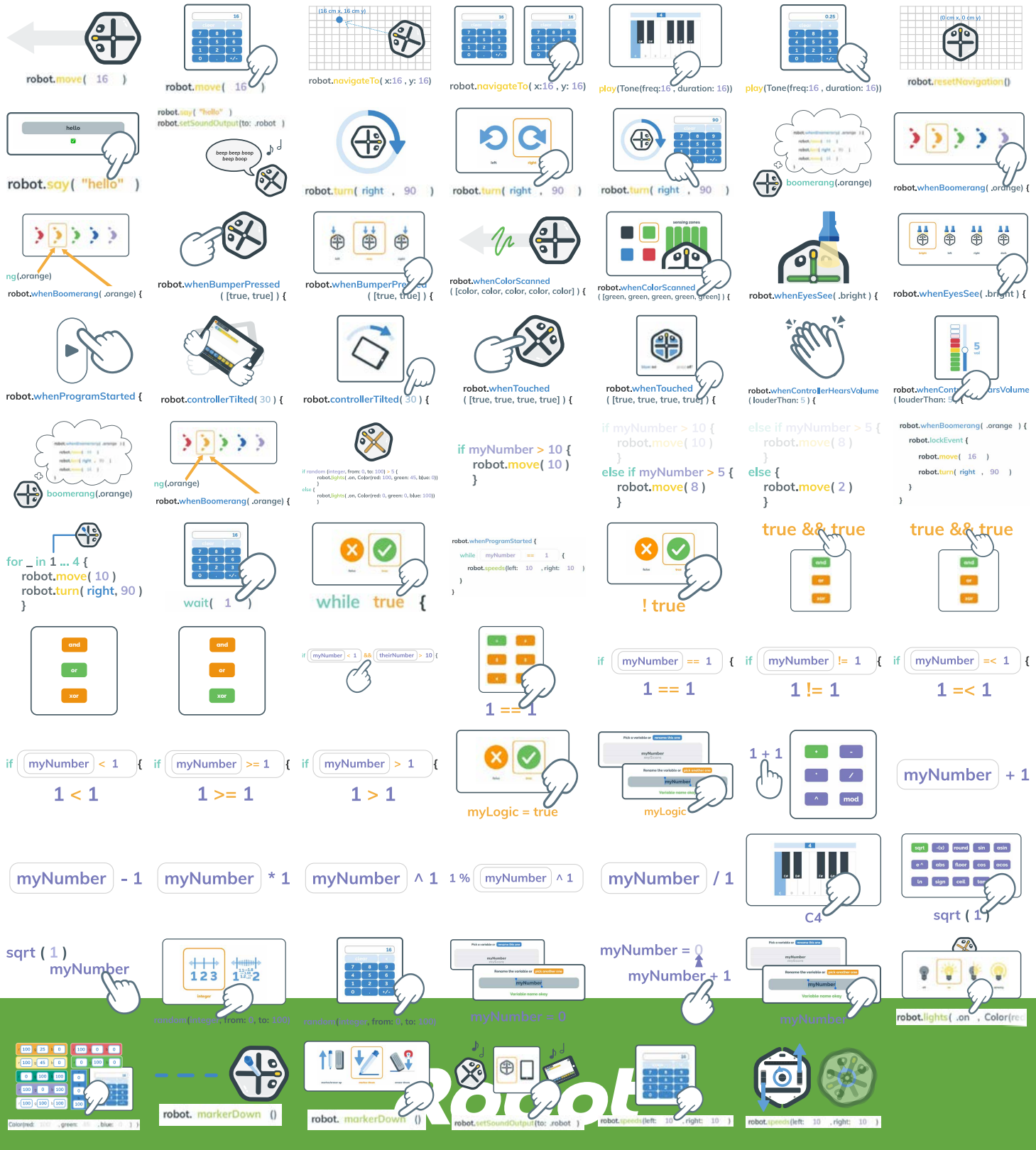
Try making one of Root's wheels turn backwards by making its speed negative. How does Root move?



iRobot®

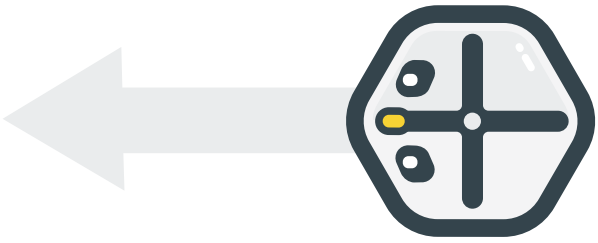
iRobot

Level 3 Block Glossary



move()

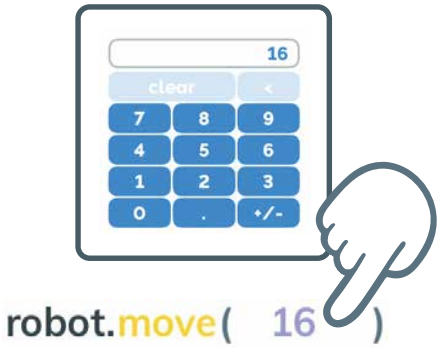
The Move Method codes Root to move forward or backward in centimeters.



robot.move(16)

Centimeters

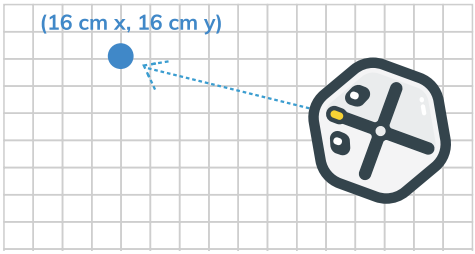
Use the editor to tell Root how many centimeters to move forward or backward.



robot.move(16)

navigateTo()

Root navigates across an invisible grid, with an origin (0 cm, 0 cm) set to Root's starting position.



robot.navigateTo(x:16 , y: 16)

navigateTo()

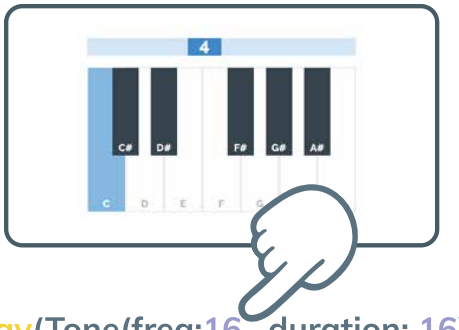
You can tell Root to move to a specific coordinate with a Navigate Method.

robot.navigateTo(x:16 , y: 16)

iRobot®

play()

A Music Method plays a music note. Open the first editor to choose which note to play.




```
play(Tone(freq:16 , duration: 16))
```

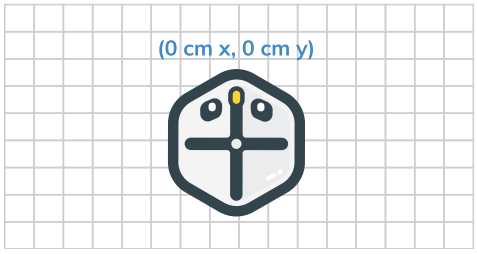
Duration

Open the second editor to tell Root how may seconds to play your note.


```
play(Tone(freq:16 , duration: 16))
```

resetNavigation()

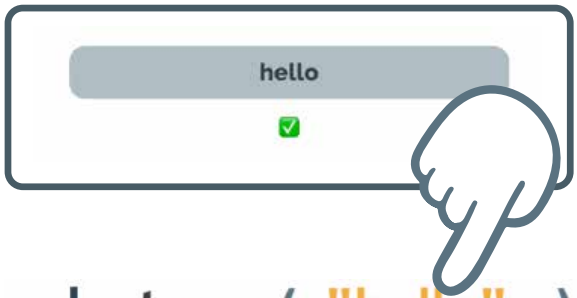
The Reset Navigation Method resets Root's invisible grid origin (0 cm, 0 cm) to Root's current location.




```
robot.resetNavigation()
```

say()

Use a Say Method to code Root or your device to say something out loud.



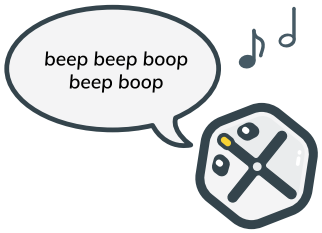

```
robot.say( "hello" )
```

iRobot®

Root Language

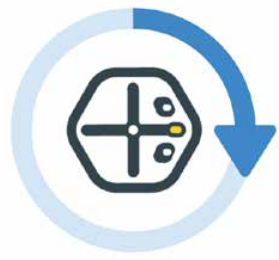
If you code Root to speak, Root will translate the words into Root Language.

```
robot.say( "hello" )
robot.setSoundOutput(to: .robot )
```



turn()

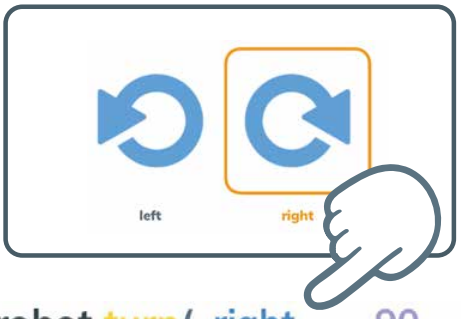
The Turn Method codes Root to turn left or right.



```
robot.turn( right , 90 )
```

Direction

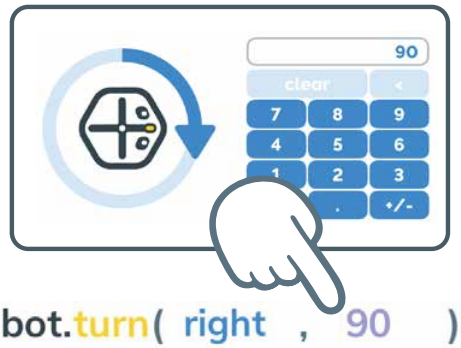
Use the first editor to tell Root to turn left or right.




```
robot.turn( right , 90 )
```

Degrees

Use the second editor to tell Root how many degrees to turn.




```
robot.turn( right , 90 )
```

iRobot®

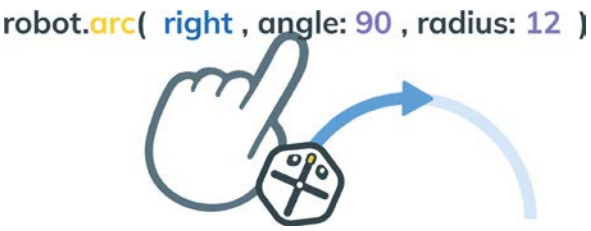
arc()

When Root's program reaches a Boomerang Function, it will do whatever code is inside its corresponding When Boomerang Event.

```
robot.arc( right , angle: 90 , radius: 12 )
```

arc() angle

The Arc Block's degrees tells your robot how far to drive around the circle.

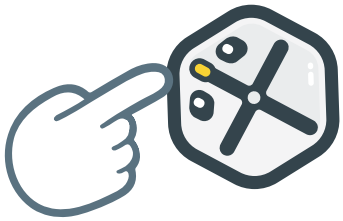


arc() radius

The Arc Block's radius tells your robot how wide your circle should be.

whenBumperPressed()

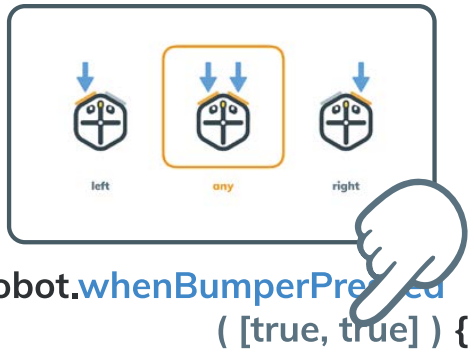
The When Bump Event tells Root to respond when its bump sensors are pressed.



```
robot.whenBumperPressed
( [true, true] ) {
```

Bump Editor

Use the editor to change which bumpers Root responds to when activated.




```
robot.whenBumperPressed
( [true, true] ) {
```

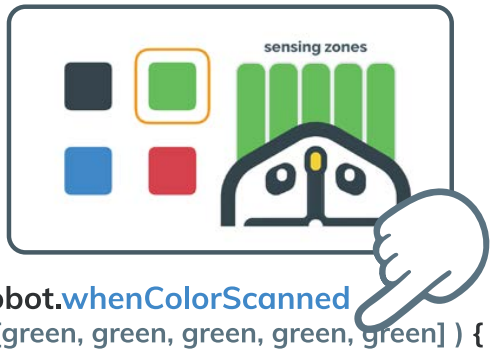
whenColorScanned()

Use the When Color Scanned Event to code Root to sense and respond to a color you have chosen.

```
robot.whenColorScanned
( [color, color, color, color, color] ) {
```

Color Sensor Editor

Open the editor to tell Root which color sense and which zones to look through.




```
robot.whenColorScanned
( [green, green, green, green, green] ) {
```

whenEyesSaw()

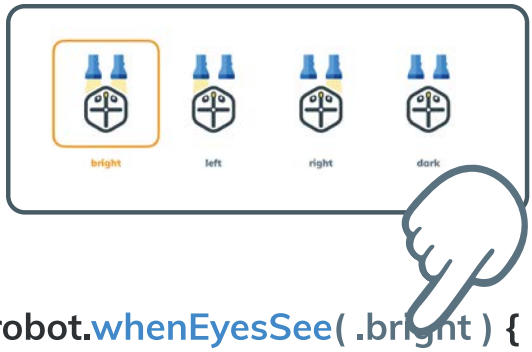
Use the When Eyes Saw Event to code Root to respond to changes in light.

```
robot.whenEyesSee( .bright ) {
```

iRobot®

Light Editor

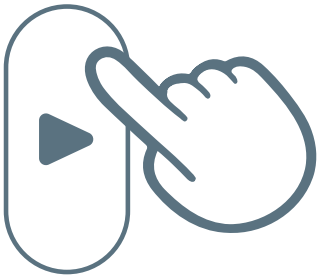
Root has two light sensors that can respond to changes in brightness.



robot.whenEyesSee(.bright) {

whenProgramStarted()

Code after the When Program Started Event will run as soon as you tap the Start button.



robot.whenProgramStarted {

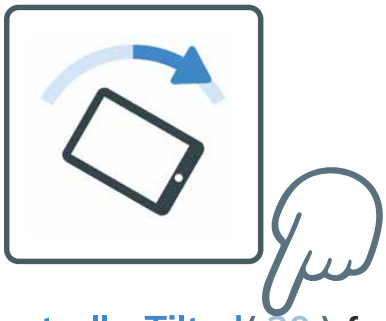
whenControllerTilted()

With the When Controller Tilted Event you can code Root to respond when you tilt your device in different directions.

robot.controllerTilted(30) {

Tilt Editor

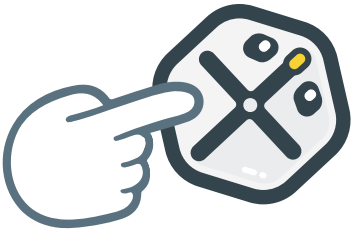
Drag the arrow to select the tilt position you'd like Root to respond to.



robot.controllerTilted(30) {

whenTouched()

With the When Touched Event, code Root to respond when one of the four zones on top of Root is pressed.



```
robot.whenTouched
( [true, true, true, true] ) {
```

Touch Editor

Use the editor to change the zones Root responds to when touched.


```
robot.whenTouched
( [true, true, true, true] ) {
```

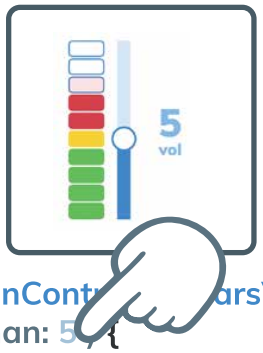
whenControllerHeardVolume()

Use the When Controller Heard Volume Event to code Root to respond to differences in volume.

```
robot.whenControllerHearsVolume
( louderThan: 5 ) {
```

Sound Editor

Use the editor to change the volume Root responds to.




```
robot.whenCont    arsVolume
( louderThan: 5 ) {
```

iRobot®

Conditional Statements

Conditional Statements can be used to tell Root what to do IF something is true or false.

```
if random (integer, from: 0, to: 100) > 5 {
        robot.lights( .on, Color(red: 100, green: 45, blue: 0))
        }
else {
        robot.lights( .on, Color(red: 0, green: 0, blue: 100))
        }
```

if true { }

If the expression inside the if is true, the program will run the enclosed statements. If the expression is false, the program will move on to the next statement.

```
if myNumber > 10 {
    robot.move( 10 )
    }
```

iRobot®

else if true { }

If the expression inside the if is false, the program will read the else if statements next. If its expression is true, its code will run. If not, the program will move to the next statement.

```
if myNumber > 10 {
    robot.move( 10 )
    }
else if myNumber > 5 {
    robot.move( 8 )
    }
```

else { }

If the expressions inside the if and all else if statements above are false, then the code inside the else statement will run.

```
else if myNumber > 5 {
    robot.move( 8 )
    }
else {
    robot.move( 2 )
    }
```

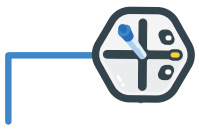
lockEvent { }

When Root sees the Lock Event Statement, it will make sure to finish the code inside without getting interrupted by other events.

```
robot.whenBoomerang( .orange ) {
    robot.lockEvent {
        robot.move( 16 )
        robot.turn( right , 90 )
    }
}
```

for _ in 1 ... x { }

for is used to create a loop of code that repeats “x” number of times.



```
for _ in 1 ... 4 {
    robot.move( 10 )
    robot.turn( right, 90 )
    }
```

iRobot®

wait()

A Wait Statement lets you set an amount of time to delay before going on to the next block.

while true { }

The While Statement can be filled with a True or False expression.

While Loop

If the expression inside is true, then its code will repeat until the expression becomes false.

```
robot.whenProgramStarted {
  while myNumber == 1 {
    robot.speeds(left: 10 , right: 10 )
  }
}
```

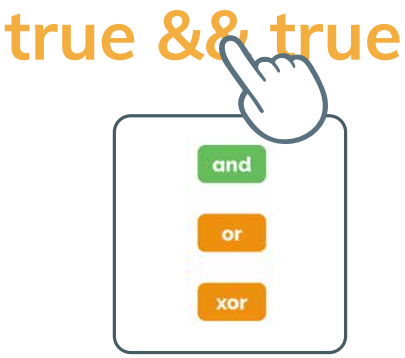
! x

The Not Operator flips its expression to its opposite, making things that are true into false and false into true.

iRobot®

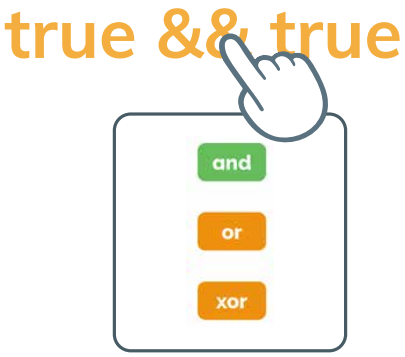
Dual Operator Block

The Boolean Operators will be either true or false all depending on the state of their parameters



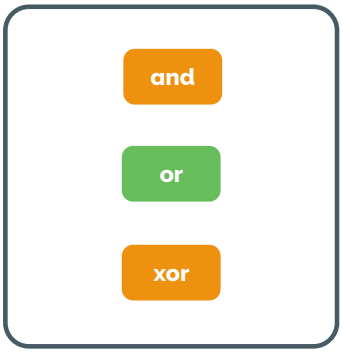
x && y

The And Operator returns true if all the expressions in the operator are true.



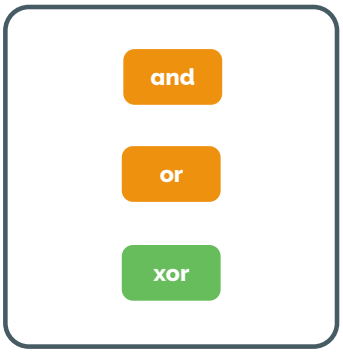
x || y

The Or Operator returns true if either the expressions in the operator are true.



x ^ y

The Exclusive Or Operator returns true if only one of the expressions in the operator is true.



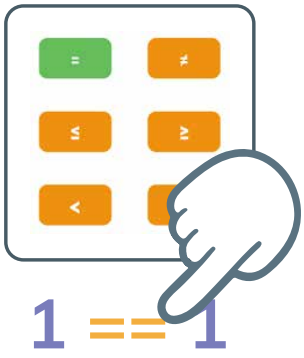
iRobot®

Using Conditionals

You can use Boolean Operators with a conditional to tell Root what to do if the expressions inside are true or false.

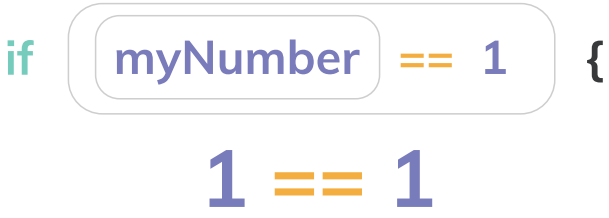
Comparison Operators

Comparison Operators will return either true or false all depending on the state of their parameters.



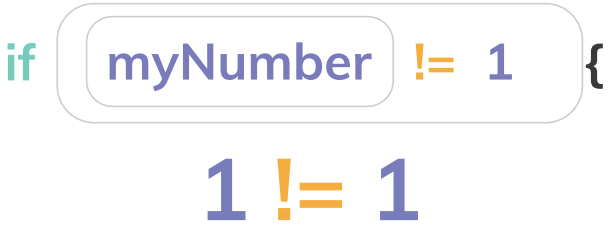
x == y

The Equal Operator returns true if the first value is equal to the second value.



x != y

The Not Equal Operator returns true if the first value is not equal to the second value.



iRobot®

x =< y

The Less Than or Equal To Operator returns true if the first value is less than or equal to the second value.

```
if myNumber =< 1 {
```

1 =< 1

---

x < y

The Less Than Operator returns true if the first value is less than the second value.

```
if myNumber < 1 {
```

1 < 1

---

x >= y

The Greater Than or Equal To Operator returns true if the first value is greater than or equal to the second value.

```
if myNumber >= 1 {
```

1 >= 1

---

x > y

The Greater Than Operator returns true if the first value is greater than the second value.

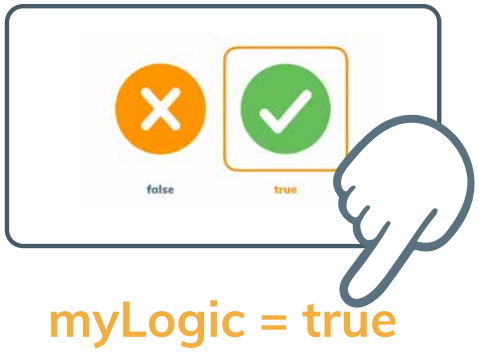
```
if myNumber > 1 {
```

1 > 1

---

myLogic = x

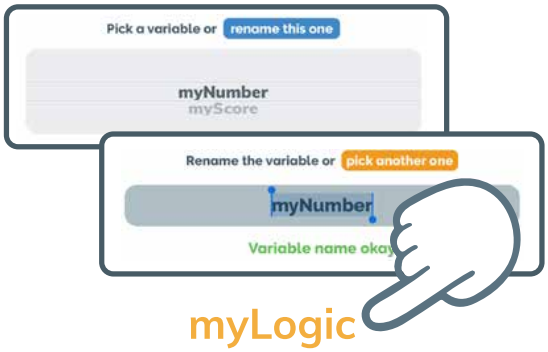
Use Set Boolean Variable Statement to set your Boolean Variable to true or false when something happens, like if a bumper is tapped.



myLogic = true

Boolean Variable

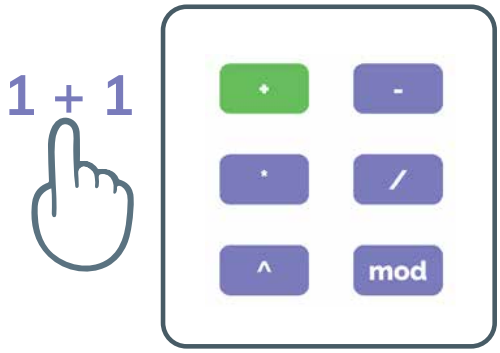
Boolean Variables can switch between true or false to help Root keep track of what's happening in the project.



myLogic

Math Operators

The Math Operators return a result of the specificed math operation between its contents.



x + y

The Addition Operator returns the sum of two values.

iRobot®

x - y

The Subtraction Operator returns the difference between two values.

```
myNumber - 1
```

---

x * y

The Multiplication Operator returns the product of two values.

```
myNumber * 1
```

---

x ^ y

The Exponentiation Operator returns the first value raised to the power of the second value.

```
myNumber ^ 1
```

---

x % y

The Modulo Operator returns the remainder when the two values inside are divided.

```
1 % myNumber ^ 1
```

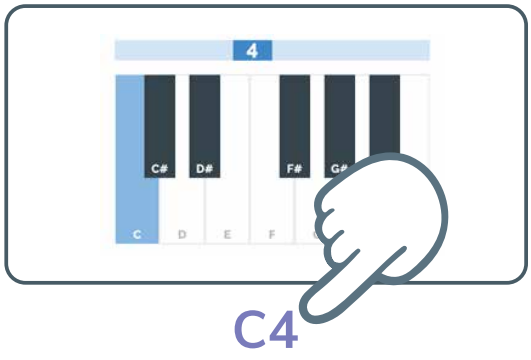
---

x / y

The Division Operator returns the quotient of two values.

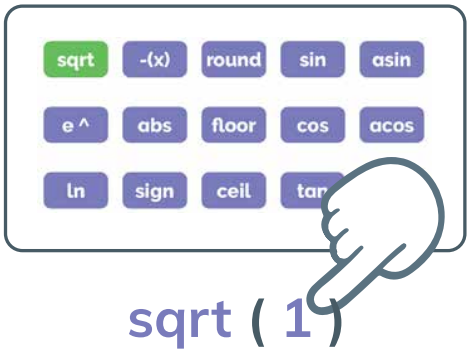
Music Frequency Block

The Music Freqency Constant equals the specified note's freqency in Hertz.



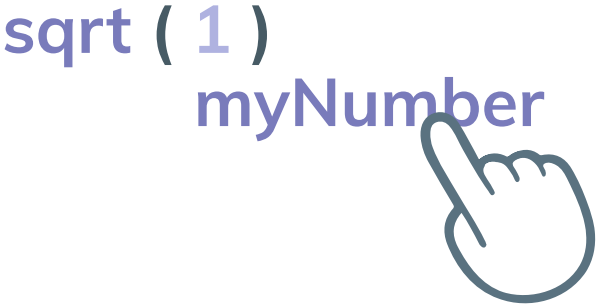
Math Functions

Math Functions return the solution of the function performed on the value inside.



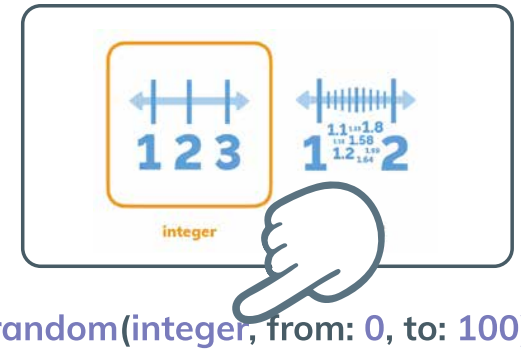
Function Editor

You can choose which function to perform by opening the Math Functions Editor.



random()

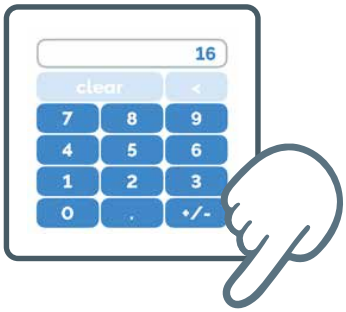
The Random Function produces any integer or decimal between the two values inside.



random(integer, from: 0, to: 100)

Random Range

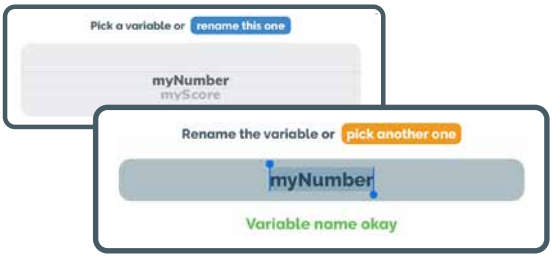
You can change the range of the Random Function picks within with the two number editors.



random(integer, from: 0, to: 100)

myNumber = x

The Set Float Variable Statement defines the value of the corresponding myNumber Variable.



myNumber = 0

Keeping Score

A Float Variable can hold a number or an operator, like myNumber + 1 that adds one point to the score.

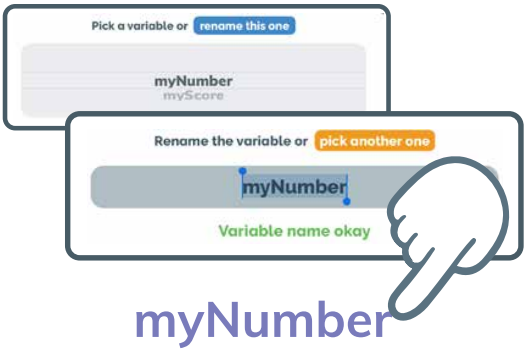
myNumber = 0

myNumber + 1

iRobot®

Float Variable

Float Variables hold values to help Root keep track of what's happening in the project.



myNumber

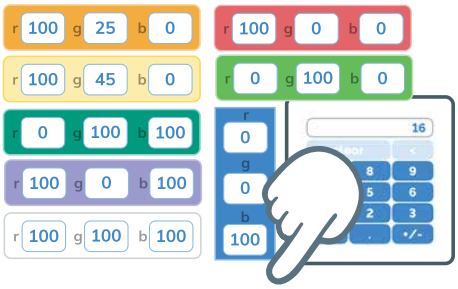
lights()

The Light Method can be used to set the color and blinking pattern for the lights on top of Root.

robot.lights(.on , Color(red

Light Editor

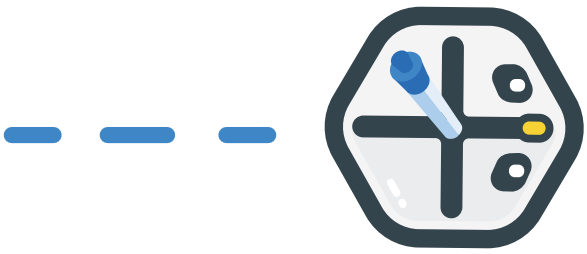
All light colors are a mix of Red, Green, and Blue light. Experiment with setting the R, G, B values to different numbers.



Color(red: 100 , green: 45 , blue: 0))

markerDown()

The Marker Method can be used to raise and lower Root's marker and eraser.



robot. markerDown ()

iRobot®

Marker Editor

Use the editor to lift and lower the marker and eraser.


```
robot. markerDown ()
```

soundOutput()

The Sound Output Method tells your code to play sounds through Root or on your device.

```
robot.setSoundOutput(to: .robot )
```

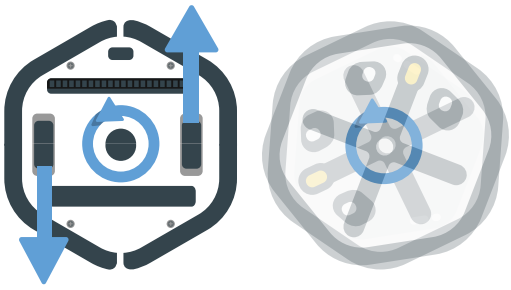
speeds()

The Wheel Speeds Method sets how fast Root's wheels move and how wide Root turns.


```
robot.speeds(left:  10 , right:  10 )
```

Wheel Speeds Editor

Try making one of Root's wheels turn backwards by making its speed negative. How does Root move?



```
robot.speeds(left:  10 , right:  10 )
```

iRobot®