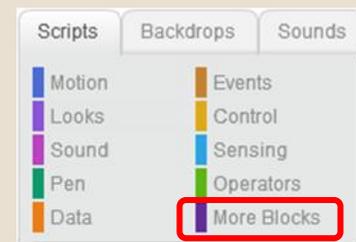
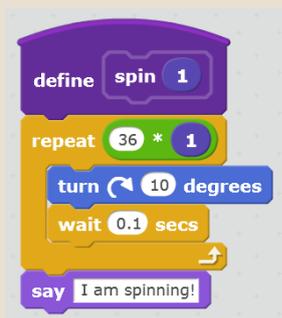




All Blocks of Scratch

Scratch has over 100 coding blocks, and each one has a unique use. They are all colour-coded into 9 different categories as seen below:

You can also create your own block under *More Blocks*.
For example, you may create a block called *spin* as this:



Here, spin 2 means to let the sprite spin for 2 full turns.

Keep in mind that all blocks can be used for *Sprites*, but only some can be used for *Backdrops*.

Here you can find all blocks in Scratch. They are organized into tasks that can be done, and sections that use the same colours as those used in Scratch.

I want to move the sprite...

go to mouse-pointer = follow where the mouse cursor is, or go to a random position

go to x: 0 y: 0 = go to any position, x is between -240 and 240, y between -180 and 180

move 10 steps = move some steps towards the current direction

change x by 10 = move left or right by a number of steps

change y by 10 = move up or down by a number of steps

set x to 0 = move left or right to a position **set y to 0** = move up or down to a position

glide 1 secs to x: 0 y: 0 = move smoothly to a position on the screen within N seconds

x position = the value of the x position where the sprite currently is at

y position = the value of the y position where the sprite currently is at

I want to change the direction / facing of the sprite...

turn  **15 degrees** = turn left for some degrees

turn  **15 degrees** = turn right for some degrees

point in direction **90** = turn to any direction. 0 degree is facing left, 90 up, 180 right, 270 down and 360 back to left.

point towards **mouse-pointer** = always face where the mouse point is

direction = the degree of which the sprite is currently facing

I want to control how the sprite bounce off stage edges...

if on edge, bounce = test if the sprite touches the edges of the stage and if so let it bounce off

set rotation style **left-right** = change the way sprite bounces off the edge: left-right, all around or don't rotate.

I want to show some text on stage...

say **Hello!** = show text in a rectangle box next to the sprite

say **Hello!** **for 2 secs** = show text in a rectangle box next to the sprite for some seconds

think **Hmm...** = show text in a rectangle box next to the sprite

think **Hmm...** **for 2 secs** = show text in a rectangle box next to the sprite for some seconds

I want to control how sprite is showing on stage...

hide = hide the sprite **show** = show the sprite **size** = size of the sprite

go back 1 layers = move the sprite back some layer so it may be hiding

go to front = move the sprite to the front / top layer

change size by 10 = make the sprite bigger with a number > 0 or smaller with a number < 0

set size to 100 % = make the sprite bigger with a number >100 or smaller with a number <100

I want to change the costume of the sprite...

next costume = show the next costume **costume #** = number of the current costume

switch costume to = show one of the costumes of the sprite

I want to change the effects of the sprite...

change **effect by** = change the selected effect by certain amount

set **effect to** = change the selected effect to a value. 0 means no effect.

clear graphic effects = remove all the effects

There are 7 ways to change the effects of a sprite in Scratch 2, they are: color, fisheye, whirl, pixelate, mosaic, brightness and ghost. Effects can be added on top of each other. Every effect has a certain value range.

I want to change the backdrop of the stage...

backdrop name = name of the current backdrop

backdrop # = number of the current backdrop **next backdrop** = show the next backdrop

switch backdrop to = show one of the backdrops

switch backdrop to **and wait** = show one of the backdrops until it's finished

I want to play a sound file ...

play sound = play the selected sound

play sound **until done** = play the selected song and wait till it's finished

stop all sounds = stop playing all sounds

I want to play drum beats ...

play drum **for** **beats** = play the selected drum for certain beats. There are 18 different types of drums to select from the list.

rest for **beats** = stop and wait for some beats

I want to play music instruments ...

play note for **beats** = play the selected note for certain beats.

set instrument to = change the current instrument to one of the 21 types on the list.

I want to control how music instruments are played...

change volume by = increase (>0) or decrease (<0) the volume by certain number

set volume to % = set volume to a percentage: less loud < 100, louder > 100

volume = the value of current volume

change tempo by = change current tempo by certain amount. To increase (playing faster) use a value > 0, and a value < to decrease (playing slower).

set tempo to **bpm** = set the tempo to a specific value

tempo = the current value of

tempo is the speed at which Scratch plays notes and drums. It is measured as *beats per minute*, for example if tempo = 60, it means 1 beat equals to 1 second.

I want to draw something on stage...

clear = clear all drawings on the stage

pen up **pen down** = sprite can draw a line when it moves and pen is down

change pen color by = change the color value by a certain number

set pen color to = set pen color to a value

set pen color to = set pen color to be the one anywhere on the screen by clicking the area

change pen size by = change pen size bigger (>0) or smaller (<0)

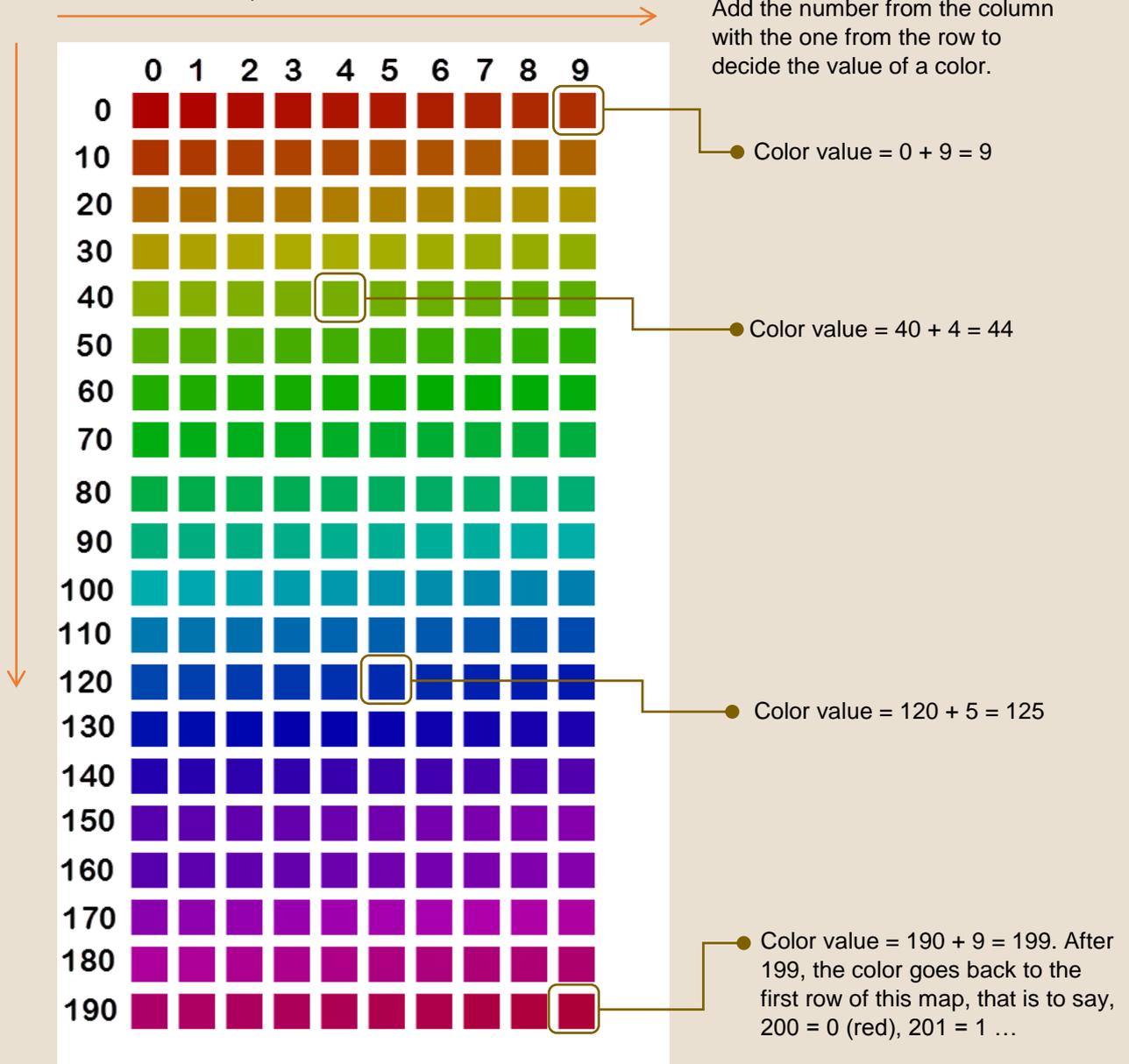
set pen size to = set pen size to a specific value

change pen shade by = change the size of the shade by a value

set pen shade to = set the size of the shade to a value

stamp = print the sprite at current position

The Scratch color map:



I want to show the score of a game on stage...

First, go to **Data** tab to make a *variable*, then use the following blocks:

change *variable* **by** **1** = add(>0) / reduce(<0) the value of selected variable by a certain amount

set *variable* **to** **0** = set the value of a selected variable

hide variable *variable* **show variable** *variable* = show/hide selected variable on the stage

variable = the value of the variable

I want the sprite or backdrop to do something when mouse is clicked or key pressed...

when  **clicked** = attach blocks you want to run when the green flag is clicked

when **key pressed** = attach blocks you want to run when a key is pressed on the keyboard

when this sprite clicked = attach blocks you want to run when the left mouse button is clicked at this sprite

I want to let other sprites or backdrop know when a sprite has done something ...

broadcast = broadcast a message with a specific name

broadcast **and wait** = broadcast a message with a specific name till it's done

when I receive = attach blocks you want to run when the a message is received

I want the sprite to do something when the sound is getting louder ...

when **>** = attach blocks you want to run when the microphone of the computer hears sounds louder than a specific value

I want to know when the backdrop has changed ...

when backdrop switches to = attach blocks you want to run when the backdrop has changed to a certain one

I want to do the same thing many times...

- forever** = run the blocks inside again and again till the red stop button is clicked
- repeat 10** = run the blocks inside for a certain number of times
- repeat until** = run the blocks inside till the specific conditions are met

I want to run blocks only when something has happened ...

- if then** = run the blocks inside only when the conditions are met
- if then else** = run the blocks below *if* when the conditions are met, or blocks below *else* if not

I want to pause or stop the running of my program ...

- wait 1 secs** = pause the program for some seconds and then continue
- wait until** = pause the program until the conditions are met
- stop all** = stop *all, this script or other scripts in sprite* from running

I want to use clones of a sprite ...

- when I start as a clone** = the first block for all other blocks that are to control a clone
- create clone of myself** = make a clone of a sprite
- delete this clone** = delete the current clone so it disappear from the stage

I want to know what the sprite is touching ...

touching color  **?** = test if the sprite is touching a specific color

color  **is touching**  **?** = test if the first color is touching the second one

touching  **mouse-pointer** **?** = test if the sprite is touching mouse pointer, edge of the stage or another sprite

I want to enter some words and let sprite know ...

ask **and wait** = show the question text next to the sprite, and a text box at the bottom of the stage, waiting for inputs from keyboard

answer = the text entered from keyboard

I want to know whether a key is pressed or mouse button clicked ...

key  **space** **pressed?** = test if a key is pressed on the keyboard

mouse down? = test if the left mouse button is clicked

mouse x **mouse y** = the current x and y positions of the mouse

distance to  **mouse-pointer** = the distance between the current sprite and mouse pointer, or another sprite

I want to use the camera and microphone on my computer ...

loudness = the value of how loud the microphone hears from outside

turn video  **off** = turn on / off computer camera

set video transparency to **%** = set the transparency of the camera image

video  **motion** **on**  **this sprite** = the value of how much motion or direction is currently in the video image

I want to know the details of my sprite ...

x position  **of**  **Sprite1** = the value of one attribute of the selected sprite.

The attributes can be: *x position, y position, direction, costume #, costume name, size and volume.*

I want to use time in my program ...

days since 2000 = the number of days since 2000 (decimal)

current **minute** = the value of *current year, month, date, day of week, hour, minute and second* of now

reset timer = set the timer to 0 and start over

timer = the number of seconds since the timer is started

I want to do some maths in my program ...

+ **-** ***** **/** = basic maths

sqrt **of** **9** = a whole collection of math functions that can be selected from the list

mod = get the remainder of the first number when it's divided by the second one

round = get the rounding of a decimal number to the nearest whole number

I want to compare 2 numbers ...

< **=** **>** = test whether the first number is less than, equal to or bigger than the second one

I want to combine conditions ...

and = it is only true when both first and second conditions are met

or = it is true as long as one of the conditions is met

not = reverse the outcome of the condition: if the condition is met, this becomes false; or true if the condition is not met.

I want to connect word ...

join **hello** **world** = attach the second word to the end of the first one

length of **world** = the number of the letters (or numbers) in a word or sentence

letter **1** **of** **world** = the Nth letter in a word or sentence

I want to control hardware from blocks in Scratch ...

In Scratch 2.0, you can connect external hardware like PicoBoard, LEGO WeDo 1.0 and 2.0, and make programs to control motors, sensors and lights in the real world. This is done through *Extension*.

What is LEGO® WeDo 2.0 ?

WeDo makes your LEGO® model move and fly! It comes with a set of 280 pieces of bricks, 1 smart hub, 1 motor, 1 motion sensor and 1 tilt sensor. You can build great Scratch program to read sensors and control motor.

The LEGO® WeDo 2.0 also comes with self-paced learning software (can to be installed on laptop or tablet) which includes:

- 1 Get Started Project, divided in 4 parts, to teach the basic functions of WeDo 2.0
- 8 Guided Projects linked to curriculum standards, with step-by-step instructions
- 8 Open Projects linked to curriculum standards, with initial design brief that inspires open-ended problem solving and exploration.

Check our web site for future resources on LEGO® WeDo 2.0!

