Light Logo
Reference Guide

Turtle

Word	Description	Try This
`all`	Turns all the lights to the color specified by all's input.	`all green` `all red`
`clean`	Turns off all the lights, sets the position to 0, sets the heading to 1 (clockwise), sets the color to 0 and puts the pen down.	`clean`
`reset`	Same as 'clean' and also shows the turtle and resets the brightness to the default value.	`reset`
`fd, bk`	Forward, back. Moves the turtle a number of steps (lights).	`setc 90 fd 10` `setc 50 bk 5`
`setc`	Sets the pen color. The colors are numbered between 0 and 100.	`setc 20 fd 8`
`ht, st`	Hides, shows the turtle.	`reset fd 7 ht`
`pu, pd`	Pen up, pen down.	`fd 5 pu fd 3 pd fd 5 ht`
`pe`	Pen erase. The turtle turns lights off as it moves.	`reset fd 24` `pe fd 5`

`dot`	Turn an LED on to a particular color	`all red`
		`dot 2 blue`
		`dot 4 blue ht`

`stamp`	Sets the color of the light under the turtle to the current pen color.	`reset setc 0 fd 6`
		`setc 30 stamp`
		`ht`

`setpos`	Sets the turtle's position. 0 is the top.	`setc 30` `setpos 5 stamp ht`
`pos`	outputs the turtle's current position.	`reset setpos 2 fd pos` `setc 30 fd pos`

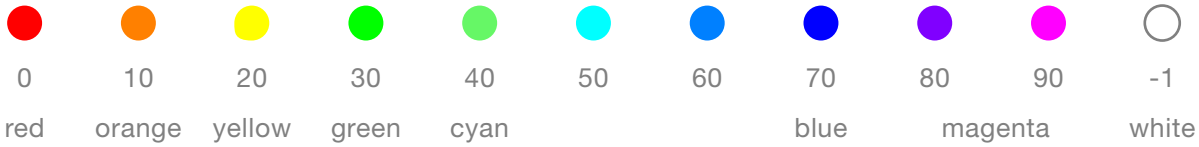
`color`	outputs the turtle's current color.	`fd 5`
		`setc color + 30`
		`fd 5`

`seth`	Sets the turtle's direction clockwise (1) or counterclockwise (-1).	`clean`
		`seth 1 setc 55 fd 10`
		`seth -1 setc 85 fd 5`

`flip`	Flips the turtle's direction from clockwise to counterclokwise and vice-versa.	`clean`
		`seth 1 setc 20 fd 10  flip`
		`setc 90 fd 5`

`setbrightness`	Sets the brightness of the current pen color. The default is 20. 99 is the maximum.	`all red`
		`setbrightness 50`
		`all red`

Colours Numbers and Names



Flow

Word	Description	Try This
`wait`	Waits for a specified amount of time in milisecons (1 second = 1000)	`all green wait 1000 all red`
`repeat`	Repeats a command a number of times.	`clean` `repeat 2 [fd 2 setc color + 10]`

`loop`	Repeats a command indefinitely.	`clean ht`
		`loop [fd 1 wait 50 setc color + 1]`
		Use the “stop button” to stop the loop

`if, ifelse`	Conditionally runs comands based on a condition.	`clean`
		`loop [fd 1 wait 50 setc color + 1 if color > 99 [stop]]`
		`repeat 10 [ifelse pos % 2 = 0 [setc 50] [setc 90] fd 3]`

`print`	Prints a number in the command center.	`print pos`

`stop`	Stops the current procedure.	Write this in a file
		`to test`
		`loop [fd 1 wait 50 setc color + 1 if color > 99 [stop]]`
		`end`
		Load the file and type
		`test`

`ouput`	Outputs a value from the current procedure.	Write this in a file
		`to randomtest`
		`ouput pos + random 2 5`
		`end`
		Load the file and type
		`print randomtest`

`make`	Changes the value of a local variable.	
`let`	Creates a local variable and gives it a value.	

Vocabulary

Math and Logic

Word	Description	Try This
`print`	Prints a number in the command center.	`print pos`

`+ - * / %`	Performs and operation on two numbers. % is modulo.	`clean fd 2`
		`fd pos + 2`
		`fd pos % 5`

`random`	outputs a random number between a minimum and a maximum value inclusive.	`setcolor 10 * random 0 10` `fd random 4 6`

`> < = !=`	Compares two numbers. != is not equals.	`print 2 = 3`
		`print 2 < 3`
		`print 2 != 3`
		(prints 1 when true, and 0 when false)

`and, or, not`	Combines conditions.	`clean fd 5` `print pos > 2 and pos < 10`

Time

Word	Description	Try This
`resett`	Resets the miliseconds timer.	`resett wait 1000 print timer`
`timer`	Outputs the miliseconds timer.	`resett wait 1000 print timer`
`resets`	Resets the seconds timer.	`resets wait 2000 print seconds`
`seconds`	Outputs the seconds timer.	`resets wait 2000 print seconds`
`resetm`	Resets the minutes timer.	`resetm wait 60000 print minutes`
`minutes`	Outputs the minutes timer.	`resetm wait 60000 print minutes`

Sound

Word	Description	Try This
`note`	plays a MIDI note for the specified duration in ms. Note 60 is equivalent to middle C	`note 60 300`
`noteon`	plays a MIDI note.	`noteon 60 wait 1000 noteoff`
`noteoff`	Turns off the MIDI note that is playing.	`noteon 65 wait 1000 noteoff`

Arduino Pins

The following examples require some sort of circuit to make it work.

Word	Description	Try This

`sensor0`
`sensor1`
`sensor2`
`sensor3`
`sensor4`
`sensor5`
	analog reporters that report a value between 0 and 1023.	Connect a sensor to the Arduino Analog In - A0
		`repeat 100[print sensor0 wait 100]`

`connected8`
`connected10`
`connected11`
	digital reporters that report false if there is an open circuit between the pin and GND, true if there is a closed circuit.	Connect a button to the Arduino Digital pin # 8 and try this program
		`to startup`
		`ht`
		`loop`
		`[ifelse connected8`
		`[all green]`
		`[clean]]`
		`end`

`on3`
`on4`
`on5`
`on6`
`on7`
	digital outputs. Sets the pin high.	Connect a motor to Digital pin #3 and #4
		`on3 off4 wait 1000 off3 on4 wait 1000 off4`

`off3`
`off4`
`off5`
`off6`
`off7`
	digital outputs. Sets the pin low.	Connect a motor to Digital pin #3 and #4
		`on3 off4 wait 1000 off3 on4 wait 1000 off4`