

Shapes	arc( cx, cy, rx, ry, th1, th2 )	Draw an arc with center (cx,cy) with radii (rx,ry), from angle th1 to angle th2
Setup	background( gray )	Set the background to (gray, gray, gray )
Setup	background( r, g, b )	Set the background to r, g, b
Setup	background( c )	Set the background to the color c
Shapes	box( b )	Draw a 3D box with dimensions b x b x b around the origin
Shapes	box( l, w, h )	Draw a 3D box with dimensions l x w x h around the origin
Color	color( c )	Set the current color to the color variable c
Color	color( gray )	Set the current color to ( gray, gray, gray )
Color	color( h, s, b )	Set the currnt color to (h, s, b ) if in HSB space
Color	color( r, g, b )	Set the currnt color to ( r, g, b ) if in RGB space
Color	colorMode( mode )	Set the color specification mode to RGB or HSB
Setup	draw( )	The function that gets called over and over to draw your scene
Shapes	ellipse( cx, cy, w, h )	Draw an ellipse in CENTER mode
Shapes	ellipse( cx, cy, x/2., y/2. )	Draw an ellipse in RADIUS mode
Shapes	ellipse( ulx, uly, llx, lly )	Draw an ellipse in CORNERS mode
Shapes	ellipse( ulx, uly, w, h )	Draw an ellipse in CORNER mode
Shapes	ellipseMode( m )	CORNER, CORNERS, CENTER, RADIUS
Drawing	fill( c )	Fill using the color c
Variables	height	Screen height in pixels
Math	int( f )	Truncate f to an integer
Variables	key	The keyboard key that is pressed
Variables	keyPressed	true if a keyboard key has been pressed
Keyboard	keyPressed( )	Gets called when a key is pressed -- what got pressed is in the variable named "key"
Shapes	line( x0, y0, x1, y1 )	Draw a line
Setup	loop( )	Starts automatic calling of draw( )
Math	map( input, lowin, highin, lowout, highout )	Lineary map the input variable from the range [lowin,highin] to [lowout,highout]
Math	max( f1, f2 )	Maximum of the two numbers
Time	millis( )	# milliseconds since the program started
Math	min( f1, f2 )	Minimum of the two numbers
Variables	mouseButton	LEFT, CENTER, RIGHT
Mouse	mouseDragged( )	Gets called when the mouse has been moved with one or more buttons down
Mouse	mouseMoved( )	Gets called when the mouse has been moved with all buttons up
Variables	mousePressed	true if a mouse button has been pressed
Mouse	mousePressed( )	Gets called when a mouse button is pressed -- what got pressed is in the variable named "mouseButton"
Mouse	mouseReleased( )	Gets called when a mouse button is released
Variables	mouseX	The current mouse X position
Variables	mouseY	The current mouse Y position
Drawing	noFill( )	Don't do any filling
Variables	PI	$\pi$
Shapes	point( x, y )	Put a dot at (x,y)
Printing	print( s )	Print the string into the console
Printing	println( s )	Print the string into the console, adding a return
Shapes	quad( x0, y0, x1, y1, x2, y2, x3, y3 )	Draw a quadrilateral
Math	radians( d )	Convert a number of degrees to radians
Randomness	random( low, high )	Return a random number between low and high
Shapes	rect( cx, cy, w, h )	Draw a rectangle in CENTER mode
Shapes	rect( cx, cy, x/2., y/2. )	Draw a rectangle in RADIUS mode
Shapes	rect( ulx, uly, llx, lly )	Draw a rectangle in CORNERS mode
Shapes	rect( ulx, uly, w, h )	Draw a rectangle in CORNER mode
Shapes	rectMode( m )	CORNER, CORNERS, CENTER, RADIUS
Transformations	rotate( r )	Perform a 2D CW rotation of r radians
Transformations	rotateX( r )	Perform an X rotation of r radians
Transformations	rotateY( r )	Perform a Y rotation of r radians
Transformations	rotateZ( r )	Perform a Z rotation of r radians
Math	round( f )	Round f to the nearest integer
Transformations	scale( sx, sy )	Scale by (sx,sy)
Setup	setup( )	The function that gets called when your program starts
Setup	size( w, h )	Set the size of the graphics window to w x h pixels
Printing	status( s )	Print a string into the status area
Drawing	stroke( c )	Outline using the color c
Drawing	strokeWeight( w )	Thickness of the outline
Text	text( s, x, y )	Draw the text "s" on the screen at (x,y) with the current fill color
Text	textFont( theFont )	Set theFont as the current font
Transformations	translate( tx, ty )	Translate by (tx,ty)
Shapes	triangle( x0, y0, x1, y1, x2, y2 )	Draw a triangle
Variables	width	Screen width in pixels