

VPython Quick Reference

Arithmetic	+ - * / ** (exponentiate) % (modulus)		
Shortcuts	+= -= *= /=		
Comparisons	== != < <= > >=		
Logic	and or not True False		
Math functions	cos(t) acos(x) log(x) (natural log) sin(t) asin(x) exp(x) (e ^x) tan(t) atan(x) atan2(y,x) (angle in polar coordinates) abs(x) sqrt(x) pow(x,y) (x ^y) round(x) floor(x) ceil(x) int(x) (round normally, down, up, or toward zero) max(x,y,...) min(x,y,...) factorial(n) combin(m,n) pi (3.14...) random() (pseudo-random number between 0 and 1)		
Control structures	<pre>if balance >= 1000: print("You're rich!") elif balance > 0: print("Keep saving pennies!") else: print("You're broke.")</pre>	<pre>while t < 10: t += dt doStuff()</pre>	<pre>for i in range(100): print("I love VPython!")</pre>
Function definitions	<pre>def startStop(): global running running = not running</pre>	<pre>def hypotenuse(a, b): return sqrt(a*a + b*b)</pre>	
Formatting numbers	"{:.3f}".format(theNumber) # round to 3 decimal places		
Lists (arrays)	<pre>x = [] # create an empty list for i in range(100): x.append(initialValue) # build the list x[0] = aValue # first entry has index 0 x[99] = x[98] + dx # last entry is 99; x[100] doesn't exist</pre>		
3D shapes	<pre>box(pos=vec1, size=vec2, color=vec3) sphere(pos=vec1, radius=num1, color=vec2) cylinder(pos=vec1, axis=vec2, radius=num1, color=vec3) spaceVector = vector(x,y,z) colorVector = vector(r,g,b)</pre>		
Scene attributes	<pre>scene.width = w # in nominal screen pixels scene.height = h scene.resizable = False # turns off user resizing scene.background = aColor scene.center = vector(x,y,z) scene.range = r # from center to edge scene.fov = theta # field of view in radians (make tiny for 2D look) scene.autoscale = False # turns off auto-scaling scene.userzoom = False # turns off zoomability scene.userspin = False # turns off rotatability</pre>		
Animation	<pre>while y > 0: rate(60) # try to run at 60 iterations per second ball.pos = vector(newx, newy, newz)</pre>		
Leaving a trail	<pre>ball = sphere(make_trail=True, trail_type="points", interval=10) ball.clear_trail()</pre>		
Plotting a graph	<pre>graph(title="A Graph", xtitle="t (s)", ytitle="x (m)", width=450, height=300, align="right", background=color.white) xDots = gdots(color=color.green, size=1, interval=10) xDots.plot(t,x) # add a point to the graph xDots.delete() # delete all dots and start over</pre>		
GUI widgets	<pre>button(text="Start/stop", bind=startStop) xSlider = slider(left=10, length=200, min=0, max=5, step=1, value=2, bind=adjustx) readout = wtext(text="25") scene.append_to_caption("\n\n") # \n is new line</pre>		