Vectors

Create a <u>vector</u> v with components vx, vy, and vz: **v = vec(vx, vy, vz)**

Calculate the <u>magnitude</u> of the vector: **mag(v)**

Create a <u>unit vector</u> in the direction of v: **hat(v)**

Creating 3D objects

<u>Sphere</u>: **myball = sphere(pos=vector(1,2,1), radius=0.5)**

<u>Box</u>: **mybox = box(pos=vector(x0,y0,z0), axis=vector(a,b,c), size=vector(L,H,W))**

<u>Arrow</u>: **myarrow = arrow(pos=vector(0,2,1), axis=vector(5,0,0))**
          (pos gives the location of the *tail* of the arrow, while axis gives the arrow *vector*)

<u>Ring</u>: **myring = ring(pos=vector(1,1,1), axis=vector(0,1,0), radius=0.5, thickness=0.1)**

Manipulating 3D objects

You can <u>refer to *any* property</u> of an object with **[name of object].[property of object]**.   For example, you can obtain the position of the arrow created above via **myarrow.pos**.

You can also <u>change properties</u> of an object using the same syntax.   For example, to change the color of the box above to magenta, use **mybox.color = color.magenta**.

You can <u>give objects additional properties</u> when you create them (*as long as there are no naming conflicts with default properties*). For example, we can associate a charge q with the sphere either when we create it via **myball = sphere(pos=vector(1,2,1), radius=0.5, q=Q)**, or after the fact via **myball.q = Q**.

Plotting data

Create a <u>graph</u>: **mygraph = graph(title='Your title here', xtitle='x-axis label',**
                                        ytitle='y-axis label', fast=False)

Create a <u>graphing object</u> and attach it to your graph: **mycurve = gcurve(graph=mygraph, color=color.cyan, label='label')** (To create *dots* instead of a *line*, replace *gcurve* with *gdots*.)

You can <u>specify other attributes of a graphing object</u> such as its width (for a line) or radius (for dots) either when creating the object or afterwards. For example, to change the line thickness to 3 pixels, use **mycurve.width = 3**.

<u>Add data</u> to the graphing object and <u>plot it</u>: **mycurve.plot([[x_1,y_1], [x_2,y_2], …])**
                    (note that the data is given as a *list of [x,y] pairs*)

Additional Glowscript and VPython documentation can be found here:
https://www.glowscript.org/docs/VPythonDocs/index.html