

Learning VPython / Glowscript

M. L. Horner

The Seven Hills School

March 15, 2014

Outline

- 1 What are VPython and Glowscript?
- 2 Using them in the classroom
- 3 Learning Resources
 - VPython
 - Glowscript

VPython and Glowscript are programming languages intended for first-year physics student use in modeling physical systems. But how does one get started? Brief answers to the following questions will be presented. Which language? How should it be introduced to students? Where and how can a teacher learn enough to be ahead of the students?

- written by physics teachers and students for learning introductory physics
- physics-focussed interactive, 3D animation
- If you can make a realistic simulation of something, then you understand how it works.
- Display of simulation can always be zoomed and rotated by default.
- focus on describing physical interactions; 3D eye-candy should be realistic enough to be gratifying and require minimal effort from student programmer

VPython

- python based
- mature; lots of useful modules
- on your computer
- user needs VPython
- won't run in a web page

Glowscript

- javascript and the WebGL based
- newer; user-made modules aren't possible
- in the cloud
- web only; some tablet access

Which to start with

- doesn't matter for beginner stuff
- Glowscript is a lot easier to use in class

Obtaining VPython/Glowscript

Both are freeware

- free as in beer
- free as in speech

assuming you don't want to rewrite the software:

- VPython: <http://vpython.org/index.html> then on left choose your operating system and follow directions
- Glowscript: <http://www.glowscript.org> sign in (accounts are free as in beer)

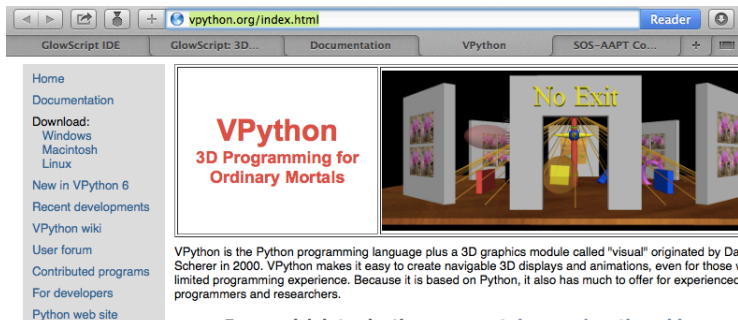
Everywhere

- Matter and Interactions textbook
- numerical model-building as core part of learning

- a couple of 80-minute periods
- don't try to teach programming; give syntax and example lines of code; learning is in the values for variables
- make a ball bounce - with many variations

<http://www.vpython.org>

- Documentation
- Examples
- Help



Home
Documentation
Download:
Windows
Macintosh
Linux
New in VPython 6
Recent developments
VPython wiki
User forum
Contributed programs
For developers
Python web site

VPython 3D Programming for Ordinary Mortals

VPython is the Python programming language plus a 3D graphics module called "visual" originated by Da Scherer in 2000. VPython makes it easy to create navigable 3D displays and animations, even for those v limited programming experience. Because it is based on Python, it also has much to offer for experienced programmers and researchers.

For a quick introduction, see youtube.com/vpythonvideos

Descriptions of the options available in the left margin:

Documentation: Overview, tutorials, and detailed documentation

Download: Free downloads for [Windows](#), [Macintosh](#), and [Linux](#)

What's new in VPython 6: New mouse/keyboard handling; native buttons, sliders, etc.

Recent developments: News and history

VPython wiki: FAQ, questions and answers, contributions from users

User forum. (There are [archives](#) of a mailing list that was used until Feb. 2013.)

Contributed programs: Interesting and useful programs contributed by users

For developers: For those interested in contributing to further development of Python

Python web site: The Python programming language, on which VPython is based

<http://vpython.org/contents/docs/index.html>

Home

If you're new to Python
and VPython: [Introduction](#)

A VPython tutorial

Pictures of 3D objects

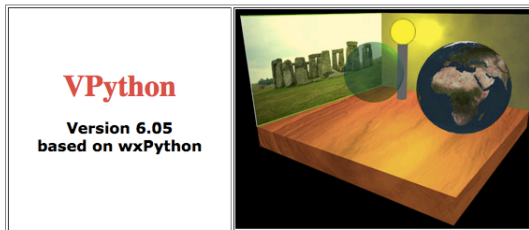
Choose a 3D object ▾

Work with 3D objects ▾

Windows/Events ▾

What's new in VPython 6

[VPython web site](#)
[VPython license](#)
[Python web site](#)
[Math module](#) (sqrt etc.)
[Numpy module](#) (arrays)



Changes from VPython 5

VPython 6 will run almost all old VPython programs correctly without change. The following detail about changes may be important in a few unusual cases.

An animation loop must contain a `rate` or `sleep` statement, which limits the number of loop iterations per second as before but also when appropriate (about 30 times per second) updates the 3D scene and handles mouse and keyboard events. Without a `rate` or `sleep` statement, the scene will not be updated until and unless the loop is completed. Most animation loops already contain a `rate` statement anyway, to make the animation not run too fast.

You should use the new function `sleep` rather than `time.sleep`. The new function periodically renders the scene and processes mouse events, making it possible to continue using `zoom` and `rotate` whereas `time.sleep` does not do this. Programs that use `time.sleep` will work, but you won't be able to zoom or rotate during the sleep period.

You must import `visual` or `vis` before importing `graph` or `controls` or `filedialog`, which most users have always done anyway.

Be sure to read what's new in [VPython 6](#).

[How to learn about VPython](#)

Home

If you're new to Python
and VPython: Introduction

A VPython tutorial

Pictures of 3D objects

✓ Choose a 3D object

Overview

arrow

box

cone

curve

cylinder

ellipsoid

extrusion

faces

frame

helix

label

lights

points

pyramid

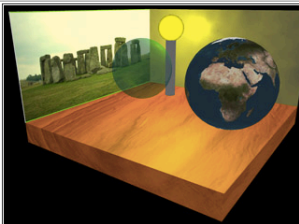
ring

sphere

text

VPython

Version 6.05
based on wxPython



Changes from VPython 5

VPython 6 will run almost all old VPython programs correctly without change. The following details about changes may be important in a few unusual cases.

An animation loop must contain a **rate** or **sleep** statement, which limits the number of loop iterations per second as before but also when appropriate (about 30 times per second) updates the 3D scene and handles mouse and keyboard events. Without a **rate** or **sleep** statement, the scene will not be updated until and unless the loop is completed. Most animation loops already contain a **rate** statement anyway, to make the animation not run too fast.

You should use the new function **sleep** rather than `time.sleep`. The new function periodically renders the scene and processes mouse events, making it possible to continue using `zoom` and `rotate` whereas `time.sleep` does not do this. Programs that use `time.sleep` will work, but you won't be able to zoom or rotate during the sleep period.

You must import `visual` or `vis` before importing `graph` or `controls` or `filedialog`, which most users have always done anyway.

Be sure to read what's new in [VPython 6](#).

[How to learn about VPython](#)

- start with bouncing ball example and change it
- skim through the help files
- see www.SOSAAPT.weebly.com for an example beginning exercise (used with students with and without programming experience)

<http://www.glowscript.org>

- Examples
- Help

glowscript.org

Sign in
Help

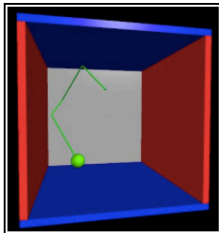
GlowScript is an easy-to-use, powerful environment for creating 3D animations and publishing them on the web. Here at glowscript.org, you can write and run GlowScript programs right in your browser, store them in the cloud for free, and easily share them with others.

New in GlowScript 1.0: print, file operations

The Help explains which browsers support GlowScript and provides full documentation.

See the [Example programs](#).

To get started writing your own programs you need to Sign in.



GlowScript 1.0

[Example programs](#) | [Forum](#)

glowscript.org

Sign in
Hide Help

[Home](#) [Pictures of 3D objects](#)

GlowScript

3D animations in a browser

Version 1.0



Choose from the menus above for full documentation of GlowScript.

GlowScript ("Graphics Library on Web") makes it easy to write real-time navigable 3D animations that run in a browser. It is based on the JavaScript language plus the WebGL 3D graphics library, both of which are included in most [modern browsers](#).

The design of GlowScript is based on that of [VPython](#). Development of GlowScript was initiated by David Scherer and Bruce Sherwood

GlowScript is an easy-to-use, powerful environment for creating 3D animations and publishing them on the web. Here at glowscript.org, you can write and run GlowScript programs right in

- start with bouncing ball example and change it
- skim through the help files