

Into 3D with VPython

Deepak Mishra
MS-IT
IIIT Hyderabad

Agenda

- What is VPython ?
- What is it capable of ?
- Python Review
- Basic Shapes
- Animating with Vpython
- Advanced animations

What is VPython ?

"3D Programming for Ordinary Mortals"

- Visual + Python = VPython
- Create :
 - Navigable 3D scenes
 - 3D simulations
 - with limited programming experience
- Not a toy

What is VPython ?

- Allows you to focus on :
 - What you want to do, not scratching your head how to do
- Very rapid prototyping
- Invaluable for teaching
- Use other Python libraries

VPython : Capabilities

Get Excited !

Some Vpython simulations to show you
what's possible

Python Review

#TODO

- Python Objects : Instantiating classes
- Lists / Tuples / Dictionaries

from visual import *

Basic shapes

Sphere

```
sphere()
```

```
sphere( pos = (1,1,1) )
```

```
sphere( pos = (1,1,1), radius = 10)
```

```
sphere( pos = (1,1,1), radius = 10,  
        color=color.red)
```

Basic Shapes

```
cylinder()
```

```
arrow()
```

```
cone()
```

Basic Shapes

```
box()
```

```
pyramid()
```

```
curve()
```

VPython basics

- Each shape instance
 - is represented by a object
- Properties like :
 - position
 - color
 - radius / length / width etc.
 - can be specified when creating (constructor)
 - can be modified later (attributed / methods)

Demonstration on basic shapes

Animating with VPython

Step 1 : Instantiate the scene

Step 2 : Decide what properties to change

Step 3 : Run an event loop

**Step 4 : Change the properties
resulting from a physical law**

Step 5 : Set the update interval

Bouncing Ball

```
floor = box      (pos=(0,0,0), length=4, height=0.5, width=4)
ball  = sphere (pos=(0,4,0), radius=1, color=color.red)
ball.velocity = vector(0,-1,0); dt = 0.01
```

```
while 1:
```

```
    rate (100)
```

```
    ball.pos = ball.pos + ball.velocity*dt
```

```
    if ball.y < ball.radius:
```

```
        ball.velocity.y = abs(ball.velocity.y)
```

```
    else:
```

```
        ball.velocity.y = ball.velocity.y - 9.8*dt
```

Advanced animations

- Rotating Objects
- Deleting Objects
- Controlling Opacity
- Controlling lighting
- Controlling animation speed

Demonstration on advanced animation

Other 3D software

- Panda 3D
- Combining with
ODE (Open Dynamics Engine)
- PyGame
- PyOpenGL

Thank You