Topics today

- GLUT
- GLUI
- OpenGL
- “Hello Triangle!”
GLUT

• GLUT is a toolkit that provides basic functions to create windows that we can render to.

• Handles events such as mouse click, keyboard button click, window resizing.

• No buttons, sliders or checkboxes, …
GLUI

- C++ library that provides buttons, spinners, sliders...
Open GL

• Graphics Rendering C API.

• Allows us to “talk” to the GPU.

• The API is defined as a set of functions which may be called by the client program.

• Cross platform Windows, OS X, Linux, iOS, Android.
What isn’t OpenGL

• Not a game engine.
• Not a photorealistic rendering API.
• Not object oriented.
• No input, physics.
OpenGL 1.x

- Uses Fixed Pipeline State Machine.
- Fixed Pipeline - the math is hardcoded (transformation, shading model, texturing).
- State Machine - You put it into various states (or modes) that then remain in effect until you change them. Can imagine it to be a huge machine with switches. But you don't see internals.
• Renders Geometric Primitives: Triangles, Quads, Lines, Points

• Renders images by means of texture mapping.
Quad or triangle

Lines

Points

3D scanner
Representing Geometric Primitives

- All primitives represented using vertices.
- Vertex is a collection of attributes: position, color, normal vector, texture coordinates.

\[ p(x,y,z) \quad c(r,g,b,a) \]
OpenGL Command Syntax

```
prefix  command name
    打交道 vertex3fv( v )

Number of components
  2 - (x, y)
  3 - (x, y, z)
  4 - (x, y, z, w)

Data Type
  b - byte
  ub - unsigned byte
  s - short
  us - unsigned short
  i - int
  ui - unsigned int
  f - float
  d - double

Vector
  omit "v" for scalar form
    打交道 vertex2f( x, y )
```

```
glColor4f(1.0, 0.0, 0.0, 1.0)
glNormal3f(0.5, 0.6, 0.7)
```
Controlling state

Setting State
  glPointSize( size );
  glLineWidth( size );

Enabling Features
  glEnable( GL_LIGHTING );
  glDisable( GL_TEXTURE_2D );
Rendering primitives

Draws White Triangle (white is default color)

```c
glBegin(GL_TRIANGLES);
    glVertex2d(0.0, 0.0);
    glVertex2d(0.0, 150.0);
    glVertex2d(150.0, 150.0);
glEnd()
```
Rendering primitives

Draws Red Triangle

```c
glBegin(GL_TRIANGLES);
    glColor3f(1.0,0.0,0.0,1.0);
    glVertex2d(0.0, 0.0);
    glVertex2d(0.0, 150.0);
    glVertex2d(150.0, 150.0);
glEnd()
```
Other primitives

- GL_LINE_STRIP
- GL_LINE_LOOP
- GL_TRIANGLE_STRIP
- GL_TRIANGLE_FAN
- GL_QUAD_STRIP
- GL_POLYGON