OpenGL Command Summary – Eric McCreath 2011 – V1.1

The aim of this summary is to provide a small basic subset of OpenGL commands such that one can get some simple scenes up and running quickly.

Matrix Operations

glMatrixMode(GL_PROJECTION) - make the current matrix the projection matrix.
glMatrixMode(GL_MODELVIEW) – make the current matrix the modelview matrix.
glPushMatrix() - copy and push the current matrix onto the stack.
glPopMatrix() - replace the current matrix with the matrix popped off the stack.
glRotated(angle, vx, vy, vz) - rotate by 'angle' degrees about the vector (vx,vy,vz)
glTranslated(x,y,z) – apply a translation.
glScale(sx,sy,sz) – apply a scale transformation.
glLoadIdentity() - set the matrix to the identity matrix.
glOrtho(left, right, bottom, top, near, far) – set up a orthogonal transformation (normally applied to the projection matrix).
glFrustum(left, right, bottom, top, near, far), gluPerspective(feildofviewY, aspect, near, far) – set up a perspective transformation.
gluLookAt(x_eye, y_eye,z_eye , x_lookat, y_lookat, z_lookat, xv_up, yv_up, zv_up) – set up the camera position of the viewing matrix.

Basic Drawing

glEnable(GL.GL_DEPTH_TEST) – turn on the depth test so that a depth buffer is used for occlusions.
glClearColor(r, g, b, a) – set the clear colour (background).
glClear(flag) – clear the current drawing buffer, where the flags include GL_COLOR_BUFFER, GL_DEPTH_BUFFER_BIT.
glColor3f(r,g,b) - set the current drawing colour.
glBegin(shape) – start a geometric drawing primitive, where shape includes: GL_POINTS, GL_POLYGON, GL_LINE, GL_LINE_STRIP, GL_TRIANGLES, GL_QUADS, GL_QUAD_STRIP
glEnd() - complete the geometric drawing primitive that was started.
glVertex3d(x, y, z) – a vertex of a point, line or polygon (normally placed between a begin and end).
glNormal3f(vx,vy,vx) – set the normal coordinates.
glFlush() - complete the initiated GL commands.
glutWireSphere(radius, slices, stacks) – draw a wire frame sphere .
glutSolidSphere(radius, slices, stacks) – draw a solid sphere.
glutSolidCube(GLdouble size) – draw a solid cube.
glutSolidTorus(inner_radius, outer_radius, sides, rings) – draw a solid torus.
glutSolidCone(radius, height, slices, stacks) – draw a cone .
glutSolidTeapot(size) – draw a teapot.

Textures

glGenTextures(arraysize,intarray) – fill the int array with available names for textures (textures are named using an integer).
glBindTexture(GL_TEXTURE_2D, name) – this makes the named texture the current texture (it also creates a texture the first time it is called for a particular texture name)
glTexImage2D(GL_TEXTURE_2D, 0, GL_RGB, width, height, 0, GL_RGB, GL_UNSIGNED_BYTE, textrueData) - load the texture data of an image that is widthxheight and is in RGB unsigned byte format into the current texture.
glEnable(GL_TEXTURE_2D) – turn on 2D textures
glTexCoord2d(u,v) – set the texture coordinate at a particular vertex.

glTexEnvf(GL_TEXTURE_ENV, GL_TEXTURE_ENV_MODE, GL_REPLACE) - use the colour of the texture to determine the colour of the surface.
glTexEnvf(GL_TEXTURE_ENV, GL_TEXTURE_ENV_MODE, GL_MODULATE) - combine the texture colour with the lighting effects.
gl.glTexParameteri(GL_TEXTURE_2D, texdim, value) - set up what happens when the texture is smaller that the surface it is rendered on ('texdim' is eitherGL_TEXTURE_WRAP_T or GL_TEXTURE_WRAP_S, and 'value' includes GL_REPEAT or GL_CLAMP)
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MIN_FILTER, GL_NEAREST),
glTexParameteri(GL_TEXTURE_2D, GL_TEXTURE_MAG_FILTER, GL_LINEAR) - setting up the magnified and minimised interpolation used.

Lighting

glEnable(GL_LIGHTING) - turn on lighting
glEnable(GL_LIGHT1) – turn on light 1
glLightfv(GL_LIGHT1, GL_AMBIENT, color_vector) - set the ambient colour of light 1
glLightfv(GL_LIGHT1, GL_DIFFUSE, color_vector) – set the diffuse colour of light 1
glLightfv(GL_LIGHT1, GL_SPECULAR, color_vector) – set the specular colour of light 1
glLightfv(GL_LIGHT1, GL_POSITION, color_vector) - set the position of light 1

glMaterialfv(GL_FRONT_AND_BACK, GL_AMBIENT_AND_DIFFUSE, diffuse_colour)- set the ambient and diffuse colour of a material.
glMaterialfv(GL_FRONT_AND_BACK, GL_SPECULAR, specular_color) – set the specular colour of a material .
glMaterialf(GL_FRONT_AND_BACK, GL_SHININESS, shine) – set the shininess of a surface (range 1 to 128) .

Note in JOGL the glLightfv, glGenTextures, and glMaterialfv methods have an addition parameter which given the index into the vector (normally just set to 0).