

OpenGL Reference Card



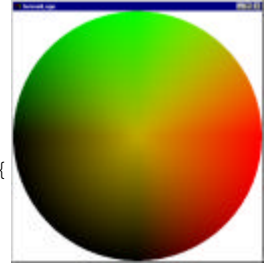
Sample program

A basic program to create a window and draw a sphere with triangles:

```
#include <GL/glut.h>
#include <math.h>

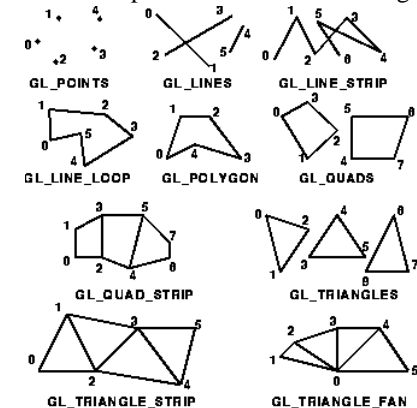
display(void) {
    float i;
    glClearColor(1., 1., 1., .0);
    glClear(GL_COLOR_BUFFER_BIT);
    glBegin(GL_TRIANGLE_FAN);
    glVertex2f(0.0f, 0.0f);
    for(i=0;i<=(2*3.1416)+.1;i=i+.1){
        glColor3f(cos(i), sin(i),0);
        glVertex2f(cos(i), sin(i));
    }
    glEnd();
    glutSwapBuffers();
}

void main(int argc, char** argv) {
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_RGB | GLUT_DOUBLE);
    glutInitWindowSize(512, 512);
    glutInitWindowPosition(20, 20);
    glutCreateWindow("tecnun");
    glutDisplayFunc(display);
    glutMainLoop();
}
```



```
void main(int argc, char** argv) {
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_RGB | GLUT_DOUBLE);
    glutInitWindowSize(512, 512);
    glutInitWindowPosition(20, 20);
    glutCreateWindow("tecnun");
    glutDisplayFunc(display);
    glutMainLoop();
}
```

The primitives of open GL and their name in glBegin():



glTexCoord1d (GLdouble s)
 glTexCoord4i (GLint s, GLint t, GLint r, GLint q)
 [1dv, 1f, 1fv, 1i, 1iv, 1s, 1sv, 2d, 2dv, 2f, 2fv, 2i, 2iv, 2s, 2sv, 3d, 3dv, 3f, 3fv, 3i, 3iv, 3s, 3sv, 4d, 4dv, 4f, 4fv, 4i, 4iv, 4s, 4sv]
 glTexCoordPointer (GLint size, GLenum type, GLsizei stride, const GLvoid *pointer)
 glTexEnvf (GLenum target, GLenum pname, GLfloat param)
 [fv, i, iv]
 glTexGend (GLenum coord, GLenum pname, GLdouble param)
 [dv, f, fv, i, iv]
 glTexImage1D (GLenum target, GLint level, GLint internalformat, GLsizei width, GLborder, GLenum format, GLenum type, const GLvoid *pixels)
 glTexImage2D (GLenum target, GLint level, GLint internalformat, GLsizei width, GLsizei height, GLint border, GLenum format, GLenum type, const GLvoid *pixels)
 glTexParameterf (GLenum target, GLenum pname, GLfloat param) [i]
 glTexParameterfv (GLenum target, GLenum pname, const GLfloat *params) [iv]
 glTexSubImage1D (GLenum target, GLint level, GLint xoffset, GLsizei width, GLenum format, GLenum type, const GLvoid *pixels)
 glTexSubImage2D (GLenum target, GLint level, GLint xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLenum type, const GLvoid *pixels)
 glTranslated (GLdouble x, GLdouble y, GLdouble z) [f]
 glVertex2d (GLdouble x, GLdouble y)
 [2dv, 2f, 2fv, 2i, 2iv, 2s, 2sv, 3d, 3dv, 3f, 3fv, 3i, 3iv, 3s, 3sv]
 glVertex4d (GLdouble x, GLdouble y, GLdouble z, GLdouble w)
 [v4dv, 4f, 4fv, 4i, 4iv, 4s, 4sv]
 glVertexPointer (GLint size, GLenum type, GLsizei stride, const GLvoid *pointer)
 glViewport (GLint x, GLint y, GLsizei width, GLsizei height)

GLU LIBRARY

const GLubyte* gluErrorString (GLenum errCode)
 const wchar_t* gluErrorUnicodeStringEXT (GLenum errCode)
 const GLubyte* gluGetString (GLenum name)
 gluOrtho2D (GLdouble left, GLdouble right, GLdouble bottom, GLdouble top)
 gluPerspective (GLdouble fovy, GLdouble aspect, GLdouble zNear, GLdouble zFar)
 gluPickMatrix (GLdouble x, GLdouble y, GLdouble width, GLdouble height, GLint viewport[4])
 gluLookAt (GLdouble eyex, GLdouble eyey, GLdouble eyez, GLdouble centerx, GLdouble centery, GLdouble centerz, GLdouble upx, GLdouble upy, GLdouble upz)
 int gluProject (GLdouble objx, GLdouble objy, GLdouble objz, const GLdouble modelMatrix[16], const GLdouble projMatrix[16], const GLint viewport[4], GLdouble *winx, GLdouble *winy, GLdouble *winz)
 int gluUnProject (GLdouble winx, GLdouble winy, GLdouble winz, const GLdouble modelMatrix[16], const GLdouble projMatrix[16], const GLint viewport[4], GLdouble *objx, GLdouble *objy, GLdouble *objz)
 int gluScaleImage (GLenum format, GLint widthin, GLint heightin, GLenum typein, const void *datain, GLint widthout, GLint heightout, GLenum typeout, void *dataout)
 int gluBuild1DMipmaps (GLenum target, GLint components, GLint width, GLenum format, GLenum type, const void *data)
 int gluBuild2DMipmaps (GLenum target, GLint components, GLint width, GLint height, GLenum format, GLenum type, const void *data)

GLUquadric* gluNewQuadric (void)
 gluDeleteQuadric (GLUquadric *state)
 gluQuadricNormals (GLUquadric *quadObject, GLenum normals)
 gluQuadricTexture (GLUquadric *quadObject, GLboolean textureCoords)
 gluQuadricOrientation (GLUquadric *quadObject, GLenum orientation)
 gluQuadricDrawStyle (GLUquadric *quadObject, GLenum drawStyle)
 gluCylinder (GLUquadric *qobj, GLdouble baseRadius, GLdouble topRadius, GLdouble height, GLint slices, GLint stacks)
 gluDisk (GLUquadric *qobj, GLdouble innerRadius, GLdouble outerRadius, GLint slices, GLint loops)
 gluPartialDisk (GLUquadric *qobj, GLdouble innerRadius, GLdouble outerRadius, GLint slices, GLint loops, GLdouble startAngle, GLdouble sweepAngle)
 gluSphere (GLUquadric *qobj, GLdouble radius, GLint slices, GLint stacks)
 gluQuadricCallback (GLUquadric *qobj, GLenum which, void (CALLBACK* fn)())
 GLUtesselator* gluNewTess(void)
 gluDeleteTess(GLUtesselator *tess)
 gluTessBeginPolygon(GLUtesselator *tess, void *polygon_data)
 gluTessBeginContour(GLUtesselator *tess)
 gluTessVertex(GLUtesselator *tess, GLdouble coords[3], void *data)
 gluTessEndContour(GLUtesselator *tess)
 gluTessEndPolygon(GLUtesselator *tess)
 gluTessProperty(GLUtesselator *tess, GLenum which, GLdouble value)
 gluTessNormal(GLUtesselator *tess, GLdouble x, GLdouble y, GLdouble z)
 gluTessCallback(GLUtesselator *tess, GLenum which, void (CALLBACK *fn)())
 gluGetTessProperty(GLUtesselator *tess, GLenum which, GLdouble *value)
 GLUnurbs* gluNewNurbsRenderer (void)
 gluDeleteNurbsRenderer (GLUnurbs *nobj)
 gluBeginSurface (GLUnurbs *nobj)
 gluBeginCurve (GLUnurbs *nobj)
 gluEndCurve (GLUnurbs *nobj)
 gluEndSurface (GLUnurbs *nobj)
 gluBeginTrim (GLUnurbs *nobj)
 gluEndTrim (GLUnurbs *nobj)
 gluPwlCurve (GLUnurbs *nobj, GLint count, GLfloat *array, GLint stride, GLenum type)
 gluNurbsCurve (GLUnurbs *nobj, GLint nknots, GLfloat *knot, GLint stride, GLfloat *ctarray, GLint order, GLenum type)
 gluNurbsSurface(GLUnurbs *nobj, GLint sknot_count, float *sknot, GLint tknot_count, GLfloat *tknot, GLint s_stride, GLint t_stride, GLfloat *ctarray, GLint sorder, GLint torder, GLenum type)
 gluLoadSamplingMatrices (GLUnurbs *nobj, const GLfloat modelMatrix[16], const GLfloat projMatrix[16], const GLint viewport[4])
 gluNurbsProperty (GLUnurbs *nobj, GLenum property, GLfloat value)
 gluGetNurbsProperty (GLUnurbs *nobj, GLenum property, GLfloat *value)
 gluNurbsCallback (GLUnurbs *nobj, GLenum which, void (CALLBACK* fn)())



CAMPUS TECNOLÓGICO DE LA UNIVERSIDAD DE NAVARRA
 NAFARROAKO UNIBERTSITATEKO CAMPUS TEKNOLOGIKOIA
 Copyright © 2004 Nicolás Serrano Bércena. v1.0
 Permission is granted to make personal copies of these sheets provided the copyright notice and this permission notice are preserved on all copies

Copyright © 2004 Nicolás Serrano Bércena - Permissions on back
<http://www.tecnun.es/graficos>

GL LIBRARY

types: b GLbyte, s GLshort, i GLint, f GLfloat, d GLdouble, ub GLubyte, us GLushort, ui GLuint

glAccum (GLenum op, GLfloat value)
glAlphaFunc (GLenum func, GLclampf ref)
GLboolean glAreTexturesResident (GLsizei n, const GLuint *textures, GLboolean *residences)
glArrayElement (GLint i)
glBegin (GLenum mode)
glBindTexture (GLenum target, GLuint texture)
glBitmap (GLsizei width, GLsizei height, GLfloat xorig, GLfloat yorig, GLfloat xmove, GLfloat ymove, const GLubyte *bitmap)
glBlendFunc (GLenum sfactor, GLenum dfactor)
glCallList (GLuint list)
glCallLists (GLsizei n, GLenum type, const GLvoid *lists)
glClear (GLbitfield mask)
glClearAccum (GLfloat red, GLfloat green, GLfloat blue, GLfloat alpha)
glClearColor (GLclampf red, GLclampf green, GLclampf blue, GLclampf alpha)
glClearDepth (GLclampd depth)
glClearIndex (GLfloat c)
glClearStencil (GLint s)
glClipPlane (GLenum plane, const GLdouble *equation)
glColor3b (GLbyte red, GLbyte green, GLbyte blue)
[3bv, 3d, 3dv, 3f, 3fv, 3i, 3iv, 3s, 3sv, 3ub, 3ubv, 3ui, 3uiv, 3us, 3usv]
glColor4b (GLbyte red, GLbyte green, GLbyte blue, GLbyte alpha)
[4bv, 4d, 4dv, 4f, 4fv, 4i, 4iv, 4s, 4sv, 4ub, 4ubv, 4ui, 4uiv, 4us, 4usv]
glColorMask (GLboolean red, GLboolean green, GLboolean blue, GLboolean alpha)
glColorMaterial (GLenum face, GLenum mode)
glColorPointer (GLint size, GLenum type, GLsizei stride, const GLvoid *pointer)
glCopyPixels (GLint x, GLint y, GLsizei width, GLsizei height, GLenum type)
glCopyTexImage1D (GLenum target, GLint level, GLenum internalFormat, GLint x, GLint y, GLsizei width, GLint border)
glCopyTexImage2D (GLenum target, GLint level, GLenum internalFormat, GLint x, GLint y, GLsizei width, GLsizei height, GLint border)
glCopyTexSubImage1D (GLenum target, GLint level, GLint xoffset, GLint x, GLint y, GLsizei width)
glCopyTexSubImage2D (GLenum target, GLint level, GLint xoffset, GLint yoffset, GLint x, GLint y, GLsizei width, GLsizei height)
glCullFace (GLenum mode)
glDeleteLists (GLuint list, GLsizei range)
glDeleteTextures (GLsizei n, const GLuint *textures)
glDepthFunc (GLenum func)
glDepthMask (GLboolean flag)
glDepthRange (GLclampd zNear, GLclampd zFar)
glDisable (GLenum cap)
glDisableClientState (GLenum array)
glDrawArrays (GLenum mode, GLint first, GLsizei count)
glDrawBuffer (GLenum mode)
glDrawElements (GLenum mode, GLsizei count, GLenum type, const GLvoid *indices)
glDrawPixels (GLsizei width, GLsizei height, GLenum format, GLenum type, const GLvoid *pixels)
glEdgeFlag (GLboolean flag)
glEdgeFlagPointer (GLsizei stride, const GLvoid *pointer)

glEdgeFlagv (const GLboolean *flag)
glEnable (GLenum cap)
glEnableClientState (GLenum array)
glEnd (void)
glEndList (void)
glEvalCoord1d (GLdouble u)
[1dv, 1f, 1fv, 2d, 2dv, 2f, 2fv]
glEvalMesh1 (GLenum mode, GLint i1, GLint i2)
glEvalMesh2 (GLenum mode, GLint i1, GLint i2, GLint j1, GLint j2)
glEvalPoint1 (GLint i)
glEvalPoint2 (GLint i, GLint j)
glFeedbackBuffer (GLsizei size, GLenum type, GLfloat *buffer)
glFinish (void)
glFlush (void)
glFogf (GLenum pname, GLfloat param) [i]
glFogfv (GLenum pname, const GLfloat *params) [iv]
glFrontFace (GLenum mode)
glFrustum (GLdouble left, GLdouble right, GLdouble bottom, GLdouble top, GLdouble zNear, GLdouble zFar)
GLuint glGenLists (GLsizei range)
glGenTextures (GLsizei n, GLuint *textures)
glGetBooleanv (GLenum pname, GLboolean *params)
glGetClipPlane (GLenum plane, GLdouble *equation)
glGetDoublev (GLenum pname, GLdouble *params)
GLenum glGetError (void)
glGetFloatv (GLenum pname, GLfloat *params)
glGetIntegerv (GLenum pname, GLint *params)
glGetLightfv (GLenum light, GLenum pname, GLfloat *params) [iv]
glGetMapdv (GLenum target, GLenum query, GLdouble *v) [fv, iv]
glGetMaterialfv (GLenum face, GLenum pname, GLfloat *params) [iv]
glGetPixelMapfv (GLenum map, GLfloat *values) [uiv, usv]
glGetPointerv (GLenum pname, GLvoid* *params)
glGetPolygonStipple (GLubyte *mask)
const GLubyte * glGetString (GLenum name)
glGetTexEnvfv (GLenum target, GLenum pname, GLfloat *params)
glGetTexEnviv (GLenum target, GLenum pname, GLint *params)
glGetTexGendv (GLenum coord, GLenum pname, GLdouble *params) [fv, iv]
glGetTexImage (GLenum target, GLint level, GLenum format, GLenum type, GLvoid *pixels)
glGetTexLevelParameterfv (GLenum target, GLint level, GLenum pname, GLfloat *params) [iv]
glGetTexParameterfv (GLenum target, GLenum pname, GLfloat *params) [iv]
glHint (GLenum target, GLenum mode)
glIndexMask (GLuint mask)
glIndexPointer (GLenum type, GLsizei stride, const GLvoid *pointer)
glIndxvd (GLdouble c)
[dv, f, fv, i, iv, s, sv, ub, ubv]
glInitNames (void)
glInterleavedArrays (GLenum format, GLsizei stride, const GLvoid *pointer)
GLboolean glIsEnabled (GLenum cap)
GLboolean glIsList (GLuint list)
GLboolean glIsTexture (GLuint texture)
glLightModelfv (GLenum pname, GLfloat param) [fv, i, iv]
glLightf (GLenum light, GLenum pname, GLfloat param) [fv, i, iv]
glLineStipple (GLint factor, GLushort pattern)
glLineWidth (GLfloat width)
glListBase (GLuint base)
glLoadIdentity (void)
glLoadMatrixd (const GLdouble *m) [f]
glLoadName (GLuint name)

glLogicOp (GLenum opcode)
glMap1d (GLenum target, GLdouble u1, GLdouble u2, GLint stride, GLint order, const GLdouble *points) [1f]
glMap2d (GLenum target, GLdouble u1, GLdouble u2, GLint ustride, GLint uorder, GLdouble v1, GLdouble v2, GLint vstride, GLint vorder, const GLdouble *points) [2f]
glMapGrid1d (GLint un, GLdouble u1, GLdouble u2) [1f]
glMapGrid2d (GLint un, GLdouble u1, GLdouble u2, GLint vn, GLdouble v1, GLdouble v2) [2f]
glMaterialf (GLenum face, GLenum pname, GLfloat param) [fv, i, iv]
glMatrixMode (GLenum mode)
glMultMatrixd (const GLdouble *m) [f]
glNewList (GLuint list, GLenum mode)
glNormal3b (GLbyte nx, GLbyte ny, GLbyte nz)
[3bv, 3d, 3dv, 3f, 3fv, 3i, 3iv, 3s, 3sv]
glNormalPointer (GLenum type, GLsizei stride, const GLvoid *pointer)
glOrtho (GLdouble left, GLdouble right, GLdouble bottom, GLdouble top, GLdouble zNear, GLdouble zFar)
glPassThrough (GLfloat token)
glPixelMapfv (GLenum map, GLsizei mapsize, const GLfloat *values) [uiv, usv]
glPixelStoref (GLenum pname, GLfloat param) [i]
glPixelTransferf (GLenum pname, GLfloat param) [i]
glPixelZoom (GLfloat xfactor, GLfloat yfactor)
glPointSize (GLfloat size)
glPolygonMode (GLenum face, GLenum mode)
glPolygonOffset (GLfloat factor, GLfloat units)
glPolygonStipple (const GLubyte *mask)
glPopAttrib (void)
glPopClientAttrib (void)
glPopMatrix (void)
glPopName (void)
glPrioritizeTextures (GLsizei n, const GLuint *textures, const GLclampf *priorities)
glPushAttrib (GLbitfield mask)
glPushClientAttrib (GLbitfield mask)
glPushMatrix (void)
glPushName (GLuint name)
glRasterPos2d (GLdouble x, GLdouble y)
[2dv, 2f, 2fv, 2i, 2iv, 2s, 2sv, 3d, 3dv, 3f, 3fv, 3i, 3iv, 3s, 3sv]
glRasterPos4d (GLdouble x, GLdouble y, GLdouble z, GLdouble w)
[4dv, 4f, 4fv, 4i, 4iv, 4s, 4sv]
glReadBuffer (GLenum mode)
glReadPixels (GLint x, GLint y, GLsizei width, GLsizei height, GLenum format, GLenum type, GLvoid *pixels)
glRectd (GLdouble x1, GLdouble y1, GLdouble x2, GLdouble y2)
[dv, f, fv, i, iv, s, sv]
GLint glRenderMode (GLenum mode)
glRotated (GLdouble angle, GLdouble x, GLdouble y, GLdouble z)
glRotatf (GLfloat angle, GLfloat x, GLfloat y, GLfloat z)
glScaled (GLdouble x, GLdouble y, GLdouble z) [f]
glScissor (GLint x, GLint y, GLsizei width, GLsizei height)
glSelectBuffer (GLsizei size, GLuint *buffer)
glShadeModel (GLenum mode)
glStencilFunc (GLenum func, GLint ref, GLuint mask)
glStencilMask (GLuint mask)
glStencilOp (GLenum fail, GLenum zfail, GLenum zpass)