**Specification**

USPD: US.ECO.00017-01 90

Component: IEcoOpenGL1

UGUID:

Status: Draft

Date: November 8, 2021

Version: 1.0

|  |  |
| --- | --- |
| **Authors** | **Company** |
| Vladimir Bashev | PEERF |
|  |  |
|  |  |

Content

[**1.** **Overview** 12](#_Toc182239027)

[**1.1.** **Introduction** 12](#_Toc182239028)

[**1.2.** **Note** 12](#_Toc182239029)

[**1.3.** **Links** 12](#_Toc182239030)

[**2.** **Eco.OpenGL1 Component** 13](#_Toc182239031)

[**3.** **IEcoOpenGL1 Interface** 14](#_Toc182239032)

[**3.1.** **IEcoOpenGL1 IDL** 14](#_Toc182239033)

[**3.1.1.** **get\_Commands function** 15](#_Toc182239034)

[**3.1.2.** **CreateContext function** 15](#_Toc182239035)

[**3.1.3.** **MakeCurrent function** 15](#_Toc182239036)

[**4.** **IEcoOpenGL1Context Interface** 16](#_Toc182239037)

[**4.1.** **IEcoOpenGL1Context IDL** 16](#_Toc182239038)

[**4.1.1.** **function** 17](#_Toc182239039)

[**5.** **IEcoOpenGL1Commands Interface** 18](#_Toc182239040)

[**5.1.** **IEcoOpenGL1Commands IDL** 18](#_Toc182239041)

[**5.1.1.** **Accum function** 34](#_Toc182239042)

[**5.1.2.** **AlphaFunc function** 34](#_Toc182239043)

[**5.1.3.** **Begin function** 34](#_Toc182239044)

[**5.1.4.** **Bitmap function** 34](#_Toc182239045)

[**5.1.5.** **BlendFunc function** 34](#_Toc182239046)

[**5.1.6.** **CallList function** 34](#_Toc182239047)

[**5.1.7.** **CallLists function** 34](#_Toc182239048)

[**5.1.8.** **Clear function** 34](#_Toc182239049)

[**5.1.9.** **ClearAccum function** 34](#_Toc182239050)

[**5.1.10.** **ClearColor function** 34](#_Toc182239051)

[**5.1.11.** **ClearDepth function** 35](#_Toc182239052)

[**5.1.12.** **ClearIndex function** 35](#_Toc182239053)

[**5.1.13.** **ClearStencil function** 35](#_Toc182239054)

[**5.1.14.** **ClipPlane function** 35](#_Toc182239055)

[**5.1.15.** **Color3b function** 35](#_Toc182239056)

[**5.1.16.** **Color3bv function** 35](#_Toc182239057)

[**5.1.17.** **Color3d function** 35](#_Toc182239058)

[**5.1.18.** **Color3dv function** 35](#_Toc182239059)

[**5.1.19.** **Color3f function** 35](#_Toc182239060)

[**5.1.20.** **Color3fv function** 36](#_Toc182239061)

[**5.1.21.** **Color3i function** 36](#_Toc182239062)

[**5.1.22.** **Color3iv function** 36](#_Toc182239063)

[**5.1.23.** **Color3s function** 36](#_Toc182239064)

[**5.1.24.** **Color3sv function** 36](#_Toc182239065)

[**5.1.25.** **Color3ub function** 36](#_Toc182239066)

[**5.1.26.** **Color3ubv function** 36](#_Toc182239067)

[**5.1.27.** **Color3ui function** 36](#_Toc182239068)

[**5.1.28.** **Color3uiv function** 36](#_Toc182239069)

[**5.1.29.** **Color3us function** 36](#_Toc182239070)

[**5.1.30.** **Color3usv function** 37](#_Toc182239071)

[**5.1.31.** **Color4b function** 37](#_Toc182239072)

[**5.1.32.** **Color4bv function** 37](#_Toc182239073)

[**5.1.33.** **Color4d function** 37](#_Toc182239074)

[**5.1.34.** **Color4dv function** 37](#_Toc182239075)

[**5.1.35.** **Color4f function** 37](#_Toc182239076)

[**5.1.36.** **Color4fv function** 37](#_Toc182239077)

[**5.1.37.** **Color4i function** 37](#_Toc182239078)

[**5.1.38.** **Color4iv function** 37](#_Toc182239079)

[**5.1.39.** **Color4s function** 38](#_Toc182239080)

[**5.1.40.** **Color4sv function** 38](#_Toc182239081)

[**5.1.41.** **Color4ub function** 38](#_Toc182239082)

[**5.1.42.** **Color4ubv function** 38](#_Toc182239083)

[**5.1.43.** **Color4ui function** 38](#_Toc182239084)

[**5.1.44.** **Color4uiv function** 38](#_Toc182239085)

[**5.1.45.** **Color4us function** 38](#_Toc182239086)

[**5.1.46.** **Color4usv function** 38](#_Toc182239087)

[**5.1.47.** **ColorMask function** 38](#_Toc182239088)

[**5.1.48.** **ColorMaterial function** 38](#_Toc182239089)

[**5.1.49.** **CopyPixels function** 39](#_Toc182239090)

[**5.1.50.** **CullFace function** 39](#_Toc182239091)

[**5.1.51.** **DeleteLists function** 39](#_Toc182239092)

[**5.1.52.** **DepthFunc function** 39](#_Toc182239093)

[**5.1.53.** **DepthMask function** 39](#_Toc182239094)

[**5.1.54.** **DepthRange function** 39](#_Toc182239095)

[**5.1.55.** **Disable function** 39](#_Toc182239096)

[**5.1.56.** **DrawBuffer function** 39](#_Toc182239097)

[**5.1.57.** **DrawPixels function** 39](#_Toc182239098)

[**5.1.58.** **EdgeFlag function** 40](#_Toc182239099)

[**5.1.59.** **EdgeFlagv function** 40](#_Toc182239100)

[**5.1.60.** **Enable function** 40](#_Toc182239101)

[**5.1.61.** **End function** 40](#_Toc182239102)

[**5.1.62.** **EndList function** 40](#_Toc182239103)

[**5.1.63.** **EvalCoord1d function** 40](#_Toc182239104)

[**5.1.64.** **EvalCoord1dv function** 40](#_Toc182239105)

[**5.1.65.** **EvalCoord1f function** 40](#_Toc182239106)

[**5.1.66.** **EvalCoord1fv function** 40](#_Toc182239107)

[**5.1.67.** **EvalCoord2d function** 40](#_Toc182239108)

[**5.1.68.** **EvalCoord2dv function** 41](#_Toc182239109)

[**5.1.69.** **EvalCoord2f function** 41](#_Toc182239110)

[**5.1.70.** **EvalCoord2fv function** 41](#_Toc182239111)

[**5.1.71.** **EvalMesh1 function** 41](#_Toc182239112)

[**5.1.72.** **EvalMesh2 function** 41](#_Toc182239113)

[**5.1.73.** **EvalPoint1 function** 41](#_Toc182239114)

[**5.1.74.** **EvalPoint2 function** 41](#_Toc182239115)

[**5.1.75.** **FeedbackBuffer function** 41](#_Toc182239116)

[**5.1.76.** **Finish function** 41](#_Toc182239117)

[**5.1.77.** **Flush function** 42](#_Toc182239118)

[**5.1.78.** **Fogf function** 42](#_Toc182239119)

[**5.1.79.** **Fogfv function** 42](#_Toc182239120)

[**5.1.80.** **Fogi function** 42](#_Toc182239121)

[**5.1.81.** **Fogiv function** 42](#_Toc182239122)

[**5.1.82.** **FrontFace function** 42](#_Toc182239123)

[**5.1.83.** **Frustum function** 42](#_Toc182239124)

[**5.1.84.** **GenLists function** 42](#_Toc182239125)

[**5.1.85.** **GetBooleanv function** 42](#_Toc182239126)

[**5.1.86.** **GetClipPlane function** 42](#_Toc182239127)

[**5.1.87.** **GetDoublev function** 43](#_Toc182239128)

[**5.1.88.** **GetError function** 43](#_Toc182239129)

[**5.1.89.** **GetFloatv function** 43](#_Toc182239130)

[**5.1.90.** **GetIntegerv function** 43](#_Toc182239131)

[**5.1.91.** **GetLightfv function** 43](#_Toc182239132)

[**5.1.92.** **GetLightiv function** 43](#_Toc182239133)

[**5.1.93.** **GetMapdv function** 43](#_Toc182239134)

[**5.1.94.** **GetMapfv function** 43](#_Toc182239135)

[**5.1.95.** **GetMapiv function** 43](#_Toc182239136)

[**5.1.96.** **GetMaterialfv function** 44](#_Toc182239137)

[**5.1.97.** **GetMaterialiv function** 44](#_Toc182239138)

[**5.1.98.** **GetPixelMapfv function** 44](#_Toc182239139)

[**5.1.99.** **GetPixelMapuiv function** 44](#_Toc182239140)

[**5.1.100.** **GetPixelMapusv function** 44](#_Toc182239141)

[**5.1.101.** **GetPolygonStipple function** 44](#_Toc182239142)

[**5.1.102.** **GetString function** 44](#_Toc182239143)

[**5.1.103.** **GetTexEnvfv function** 44](#_Toc182239144)

[**5.1.104.** **GetTexEnviv function** 44](#_Toc182239145)

[**5.1.105.** **GetTexGendv function** 44](#_Toc182239146)

[**5.1.106.** **GetTexGenfv function** 45](#_Toc182239147)

[**5.1.107.** **GetTexGeniv function** 45](#_Toc182239148)

[**5.1.108.** **GetTexImage function** 45](#_Toc182239149)

[**5.1.109.** **GetTexLevelParameterfv function** 45](#_Toc182239150)

[**5.1.110.** **GetTexLevelParameteriv function** 45](#_Toc182239151)

[**5.1.111.** **GetTexParameterfv function** 45](#_Toc182239152)

[**5.1.112.** **GetTexParameteriv function** 45](#_Toc182239153)

[**5.1.113.** **Hint function** 45](#_Toc182239154)

[**5.1.114.** **IndexMask function** 45](#_Toc182239155)

[**5.1.115.** **Indexd function** 46](#_Toc182239156)

[**5.1.116.** **Indexdv function** 46](#_Toc182239157)

[**5.1.117.** **Indexf function** 46](#_Toc182239158)

[**5.1.118.** **Indexfv function** 46](#_Toc182239159)

[**5.1.119.** **Indexi function** 46](#_Toc182239160)

[**5.1.120.** **Indexiv function** 46](#_Toc182239161)

[**5.1.121.** **Indexs function** 46](#_Toc182239162)

[**5.1.122.** **Indexsv function** 46](#_Toc182239163)

[**5.1.123.** **InitNames function** 46](#_Toc182239164)

[**5.1.124.** **IsEnabled function** 46](#_Toc182239165)

[**5.1.125.** **IsList function** 47](#_Toc182239166)

[**5.1.126.** **LightModelf function** 47](#_Toc182239167)

[**5.1.127.** **LightModelfv function** 47](#_Toc182239168)

[**5.1.128.** **LightModeli function** 47](#_Toc182239169)

[**5.1.129.** **LightModeliv function** 47](#_Toc182239170)

[**5.1.130.** **Lightf function** 47](#_Toc182239171)

[**5.1.131.** **Lightfv function** 47](#_Toc182239172)

[**5.1.132.** **Lighti function** 47](#_Toc182239173)

[**5.1.133.** **Lightiv function** 47](#_Toc182239174)

[**5.1.134.** **LineWidth function** 48](#_Toc182239175)

[**5.1.135.** **ListBase function** 48](#_Toc182239176)

[**5.1.136.** **LoadIdentity function** 48](#_Toc182239177)

[**5.1.137.** **LoadMatrixd function** 48](#_Toc182239178)

[**5.1.138.** **LoadMatrixf function** 48](#_Toc182239179)

[**5.1.139.** **LoadName function** 48](#_Toc182239180)

[**5.1.140.** **LogicOp function** 48](#_Toc182239181)

[**5.1.141.** **Map1d function** 48](#_Toc182239182)

[**5.1.142.** **Map1f function** 48](#_Toc182239183)

[**5.1.143.** **Map2d function** 48](#_Toc182239184)

[**5.1.144.** **Map2f function** 49](#_Toc182239185)

[**5.1.145.** **MapGrid1d function** 49](#_Toc182239186)

[**5.1.146.** **MapGrid1f function** 49](#_Toc182239187)

[**5.1.147.** **MapGrid2d function** 49](#_Toc182239188)

[**5.1.148.** **MapGrid2f function** 49](#_Toc182239189)

[**5.1.149.** **Materialf function** 49](#_Toc182239190)

[**5.1.150.** **Materialfv function** 49](#_Toc182239191)

[**5.1.151.** **Materiali function** 49](#_Toc182239192)

[**5.1.152.** **Materialiv function** 49](#_Toc182239193)

[**5.1.153.** **MatrixMode function** 50](#_Toc182239194)

[**5.1.154.** **MultMatrixd function** 50](#_Toc182239195)

[**5.1.155.** **MultMatrixf function** 50](#_Toc182239196)

[**5.1.156.** **NewList function** 50](#_Toc182239197)

[**5.1.157.** **Normal3b function** 50](#_Toc182239198)

[**5.1.158.** **Normal3bv function** 50](#_Toc182239199)

[**5.1.159.** **Normal3d function** 50](#_Toc182239200)

[**5.1.160.** **Normal3dv function** 50](#_Toc182239201)

[**5.1.161.** **Normal3f function** 50](#_Toc182239202)

[**5.1.162.** **Normal3fv function** 50](#_Toc182239203)

[**5.1.163.** **Normal3i function** 51](#_Toc182239204)

[**5.1.164.** **Normal3iv function** 51](#_Toc182239205)

[**5.1.165.** **Normal3s function** 51](#_Toc182239206)

[**5.1.166.** **Normal3sv function** 51](#_Toc182239207)

[**5.1.167.** **Ortho function** 51](#_Toc182239208)

[**5.1.168.** **PassThrough function** 51](#_Toc182239209)

[**5.1.169.** **PixelMapfv function** 51](#_Toc182239210)

[**5.1.170.** **PixelMapuiv function** 51](#_Toc182239211)

[**5.1.171.** **PixelMapusv function** 51](#_Toc182239212)

[**5.1.172.** **PixelStoref function** 52](#_Toc182239213)

[**5.1.173.** **PixelStorei function** 52](#_Toc182239214)

[**5.1.174.** **PixelTransferf function** 52](#_Toc182239215)

[**5.1.175.** **PixelTransferi function** 52](#_Toc182239216)

[**5.1.176.** **PixelZoom function** 52](#_Toc182239217)

[**5.1.177.** **PointSize function** 52](#_Toc182239218)

[**5.1.178.** **PolygonMode function** 52](#_Toc182239219)

[**5.1.179.** **PolygonStipple function** 52](#_Toc182239220)

[**5.1.180.** **PopAttrib function** 52](#_Toc182239221)

[**5.1.181.** **PopMatrix function** 52](#_Toc182239222)

[**5.1.182.** **PopName function** 53](#_Toc182239223)

[**5.1.183.** **PushAttrib function** 53](#_Toc182239224)

[**5.1.184.** **PushMatrix function** 53](#_Toc182239225)

[**5.1.185.** **PushName function** 53](#_Toc182239226)

[**5.1.186.** **RasterPos2d function** 53](#_Toc182239227)

[**5.1.187.** **RasterPos2dv function** 53](#_Toc182239228)

[**5.1.188.** **RasterPos2f function** 53](#_Toc182239229)

[**5.1.189.** **RasterPos2fv function** 53](#_Toc182239230)

[**5.1.190.** **RasterPos2i function** 53](#_Toc182239231)

[**5.1.191.** **RasterPos2iv function** 54](#_Toc182239232)

[**5.1.192.** **RasterPos2s function** 54](#_Toc182239233)

[**5.1.193.** **RasterPos2sv function** 54](#_Toc182239234)

[**5.1.194.** **RasterPos3d function** 54](#_Toc182239235)

[**5.1.195.** **RasterPos3dv function** 54](#_Toc182239236)

[**5.1.196.** **RasterPos3f function** 54](#_Toc182239237)

[**5.1.197.** **RasterPos3fv function** 54](#_Toc182239238)

[**5.1.198.** **RasterPos3i function** 54](#_Toc182239239)

[**5.1.199.** **RasterPos3iv function** 54](#_Toc182239240)

[**5.1.200.** **RasterPos3s function** 54](#_Toc182239241)

[**5.1.201.** **RasterPos3sv function** 55](#_Toc182239242)

[**5.1.202.** **RasterPos4d function** 55](#_Toc182239243)

[**5.1.203.** **RasterPos4dv function** 55](#_Toc182239244)

[**5.1.204.** **RasterPos4f function** 55](#_Toc182239245)

[**5.1.205.** **RasterPos4fv function** 55](#_Toc182239246)

[**5.1.206.** **RasterPos4i function** 55](#_Toc182239247)

[**5.1.207.** **RasterPos4iv function** 55](#_Toc182239248)

[**5.1.208.** **RasterPos4s function** 55](#_Toc182239249)

[**5.1.209.** **RasterPos4sv function** 55](#_Toc182239250)

[**5.1.210.** **ReadBuffer function** 56](#_Toc182239251)

[**5.1.211.** **ReadPixels function** 56](#_Toc182239252)

[**5.1.212.** **Rectd function** 56](#_Toc182239253)

[**5.1.213.** **Rectdv function** 56](#_Toc182239254)

[**5.1.214.** **Rectf function** 56](#_Toc182239255)

[**5.1.215.** **Rectfv function** 56](#_Toc182239256)

[**5.1.216.** **Recti function** 56](#_Toc182239257)

[**5.1.217.** **Rectiv function** 56](#_Toc182239258)

[**5.1.218.** **Rects function** 56](#_Toc182239259)

[**5.1.219.** **Rectsv function** 56](#_Toc182239260)

[**5.1.220.** **RenderMode function** 57](#_Toc182239261)

[**5.1.221.** **Rotated function** 57](#_Toc182239262)

[**5.1.222.** **Rotatef function** 57](#_Toc182239263)

[**5.1.223.** **Scaled function** 57](#_Toc182239264)

[**5.1.224.** **Scalef function** 57](#_Toc182239265)

[**5.1.225.** **Scissor function** 57](#_Toc182239266)

[**5.1.226.** **SelectBuffer function** 57](#_Toc182239267)

[**5.1.227.** **ShadeModel function** 57](#_Toc182239268)

[**5.1.228.** **StencilFunc function** 57](#_Toc182239269)

[**5.1.229.** **StencilOp function** 58](#_Toc182239270)

[**5.1.230.** **TexCoord1d function** 58](#_Toc182239271)

[**5.1.231.** **TexCoord1dv function** 58](#_Toc182239272)

[**5.1.232.** **TexCoord1f function** 58](#_Toc182239273)

[**5.1.233.** **TexCoord1fv function** 58](#_Toc182239274)

[**5.1.234.** **TexCoord1i function** 58](#_Toc182239275)

[**5.1.235.** **TexCoord1iv function** 58](#_Toc182239276)

[**5.1.236.** **TexCoord1s function** 58](#_Toc182239277)

[**5.1.237.** **TexCoord1sv function** 58](#_Toc182239278)

[**5.1.238.** **TexCoord2d function** 58](#_Toc182239279)

[**5.1.239.** **TexCoord2dv function** 59](#_Toc182239280)

[**5.1.240.** **TexCoord2f function** 59](#_Toc182239281)

[**5.1.241.** **TexCoord2fv function** 59](#_Toc182239282)

[**5.1.242.** **TexCoord2i function** 59](#_Toc182239283)

[**5.1.243.** **TexCoord2iv function** 59](#_Toc182239284)

[**5.1.244.** **TexCoord2s function** 59](#_Toc182239285)

[**5.1.245.** **TexCoord2sv function** 59](#_Toc182239286)

[**5.1.246.** **TexCoord3d function** 59](#_Toc182239287)

[**5.1.247.** **TexCoord3dv function** 59](#_Toc182239288)

[**5.1.248.** **TexCoord3f function** 60](#_Toc182239289)

[**5.1.249.** **TexCoord3fv function** 60](#_Toc182239290)

[**5.1.250.** **TexCoord3i function** 60](#_Toc182239291)

[**5.1.251.** **TexCoord3iv function** 60](#_Toc182239292)

[**5.1.252.** **TexCoord3s function** 60](#_Toc182239293)

[**5.1.253.** **TexCoord3sv function** 60](#_Toc182239294)

[**5.1.254.** **TexCoord4d function** 60](#_Toc182239295)

[**5.1.255.** **TexCoord4dv function** 60](#_Toc182239296)

[**5.1.256.** **TexCoord4f function** 60](#_Toc182239297)

[**5.1.257.** **TexCoord4fv function** 60](#_Toc182239298)

[**5.1.258.** **TexCoord4i function** 61](#_Toc182239299)

[**5.1.259.** **TexCoord4iv function** 61](#_Toc182239300)

[**5.1.260.** **TexCoord4s function** 61](#_Toc182239301)

[**5.1.261.** **TexCoord4sv function** 61](#_Toc182239302)

[**5.1.262.** **TexEnvf function** 61](#_Toc182239303)

[**5.1.263.** **TexEnvfv function** 61](#_Toc182239304)

[**5.1.264.** **TexEnvi function** 61](#_Toc182239305)

[**5.1.265.** **TexEnviv function** 61](#_Toc182239306)

[**5.1.266.** **TexGend function** 61](#_Toc182239307)

[**5.1.267.** **TexGendv function** 62](#_Toc182239308)

[**5.1.268.** **TexGenf function** 62](#_Toc182239309)

[**5.1.269.** **TexGenfv function** 62](#_Toc182239310)

[**5.1.270.** **TexGeni function** 62](#_Toc182239311)

[**5.1.271.** **TexGeniv function** 62](#_Toc182239312)

[**5.1.272.** **TexImage1D function** 62](#_Toc182239313)

[**5.1.273.** **TexParameterf function** 62](#_Toc182239314)

[**5.1.274.** **TexParameteri function** 62](#_Toc182239315)

[**5.1.275.** **TexParameteriv function** 62](#_Toc182239316)

[**5.1.276.** **Translated function** 62](#_Toc182239317)

[**5.1.277.** **Translatef function** 63](#_Toc182239318)

[**5.1.278.** **Vertex2d function** 63](#_Toc182239319)

[**5.1.279.** **Vertex2dv function** 63](#_Toc182239320)

[**5.1.280.** **Vertex2f function** 63](#_Toc182239321)

[**5.1.281.** **Vertex2fv function** 63](#_Toc182239322)

[**5.1.282.** **Vertex2i function** 63](#_Toc182239323)

[**5.1.283.** **Vertex2iv function** 63](#_Toc182239324)

[**5.1.284.** **Vertex2s function** 63](#_Toc182239325)

[**5.1.285.** **Vertex2sv function** 63](#_Toc182239326)

[**5.1.286.** **Vertex3d function** 64](#_Toc182239327)

[**5.1.287.** **Vertex3dv function** 64](#_Toc182239328)

[**5.1.288.** **Vertex3f function** 64](#_Toc182239329)

[**5.1.289.** **Vertex3fv function** 64](#_Toc182239330)

[**5.1.290.** **Vertex3i function** 64](#_Toc182239331)

[**5.1.291.** **Vertex3iv function** 64](#_Toc182239332)

[**5.1.292.** **Vertex3s function** 64](#_Toc182239333)

[**5.1.293.** **Vertex3sv function** 64](#_Toc182239334)

[**5.1.294.** **Vertex4d function** 64](#_Toc182239335)

[**5.1.295.** **Vertex4dv function** 64](#_Toc182239336)

[**5.1.296.** **Vertex4f function** 65](#_Toc182239337)

[**5.1.297.** **Vertex4fv function** 65](#_Toc182239338)

[**5.1.298.** **Vertex4i function** 65](#_Toc182239339)

[**5.1.299.** **Vertex4iv function** 65](#_Toc182239340)

[**5.1.300.** **Vertex4s function** 65](#_Toc182239341)

[**5.1.301.** **Vertex4sv function** 65](#_Toc182239342)

[**5.1.302.** **Viewport function** 65](#_Toc182239343)

[**6.** **Error codes** 66](#_Toc182239344)

[**Appendix A: Training programs** 67](#_Toc182239345)

1. **Overview**

This document describes the requirements for the implementation of the Eco.OpenGL1 component. (Component)

* 1. **Introduction**

Description.

* 1. **Note**
* Keywords
  1. **Links**

This paragraph contains links to information to help you understand this document:

[] – name of the link

Available by: http://address

1. **Eco.OpenGL1 Component**

The Eco.OpenGL1 component is

The component has the following description:

1. **IEcoOpenGL1 Interface**
   1. **IEcoOpenGL1 IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl"  import "IEcoOpenGL1Context.idl"  import "IEcoOpenGL1Commands.idl" | | | |
| [  object,  uguid(29E18D21-A15E-4703-89F3-72A7CAAA0657),  ] | | | |
| interface IEcoOpenGL1 : IEcoUnknown { | | | |
|  | | | |
| IEcoOpenGL1Commands\* | | ***get\_Commands*** | ([in] void); |
|  | |  |  |
| IEcoOpenGL1Context\* | | ***CreateContext*** | ([in] void\* pIContext); |
|  | |  |  |
| int16\_t | | ***MakeCurrent*** | ([in] IEcoOpenGL1Context\* pIRenderContext,  [in] void\* pIContext); |
|  | |  |  |
| } | |  |  |

* + 1. **get\_Commands function**

The function

* + 1. **CreateContext function**

The function

* + 1. **MakeCurrent function**

The function

1. **IEcoOpenGL1Context Interface**
   1. **IEcoOpenGL1Context IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl" | | | |
| [  object,  uguid(565ABC44-C31C-4383-84B6-7D76ACB07DFC),  ] | | | |
| interface IEcoOpenGL1Context : IEcoUnknown { | | | |
|  | | | |
|  | |  |  |
|  | |  |  |
| } | |  |  |

* + 1. **function**

The function

1. **IEcoOpenGL1Commands Interface**
   1. **IEcoOpenGL1Commands IDL**

|  |
| --- |
| **ECO IDL** |
| import "IEcoBase1.idl" | | | |
| [  object,  uguid(9C787CEE-BA56-4690-B30D-4B002E2B361A),  ] | | | |
| interface IEcoOpenGL1Commands : IEcoUnknown { | | | |
|  | | | |
| void | | ***Accum*** | ([in] GLenum op,  GLfloat value); |
|  | |  |  |
| void | | ***AlphaFunc*** | ([in] GLenum func,  GLclampf ref); |
|  | |  |  |
| void | | ***Begin*** | ([in] GLenum mode); |
|  | |  |  |
| void | | ***Bitmap*** | ([in] GLsizei width,  GLsizei height,  GLfloat xorig,  GLfloat yorig,  GLfloat xmove,  GLfloat ymove,  const GLubyte \*bitmap); |
|  | |  |  |
| void | | ***BlendFunc*** | ([in] GLenum sfactor,  GLenum dfactor); |
|  | |  |  |
| void | | ***CallList*** | ([in] GLuint list); |
|  | |  |  |
| void | | ***CallLists*** | ([in] GLsizei n,  GLenum type,  const GLvoid \*lists); |
|  | |  |  |
| void | | ***Clear*** | ([in] GLbitfield mask); |
|  | |  |  |
| void | | ***ClearAccum*** | ([in] GLfloat red,  GLfloat green,  GLfloat blue,  GLfloat alpha); |
|  | |  |  |
| void | | ***ClearColor*** | ([in] GLclampf red,  GLclampf green,  GLclampf blue,  GLclampf alpha); |
|  | |  |  |
| void | | ***ClearDepth*** | ([in] GLclampd depth); |
|  | |  |  |
| void | | ***ClearIndex*** | ([in] GLfloat c); |
|  | |  |  |
| void | | ***ClearStencil*** | ([in] GLint s); |
|  | |  |  |
| void | | ***ClipPlane*** | ([in] GLenum plane,  const GLdouble \*equation); |
|  | |  |  |
| void | | ***Color3b*** | ([in] GLbyte red,  GLbyte green,  GLbyte blue); |
|  | |  |  |
| void | | ***Color3bv*** | ([in] const GLbyte \*v); |
|  | |  |  |
| void | | ***Color3d*** | ([in] GLdouble red,  GLdouble green,  GLdouble blue); |
|  | |  |  |
| void | | ***Color3dv*** | ([in] const GLdouble \*v); |
|  | |  |  |
| void | | ***Color3f*** | ([in] GLfloat red,  GLfloat green,  GLfloat blue); |
|  | |  |  |
| void | | ***Color3fv*** | ([in] const GLfloat \*v); |
|  | |  |  |
| void | | ***Color3i*** | ([in] GLint red,  GLint green,  GLint blue); |
|  | |  |  |
| void | | ***Color3iv*** | ([in] const GLint \*v); |
|  | |  |  |
| void | | ***Color3s*** | ([in] GLshort red,  GLshort green,  GLshort blue); |
|  | |  |  |
| void | | ***Color3sv*** | ([in] const GLshort \*v); |
|  | |  |  |
| void | | ***Color3ub*** | ([in] GLubyte red,  GLubyte green,  GLubyte blue); |
|  | |  |  |
| void | | ***Color3ubv*** | ([in] const GLubyte \*v); |
|  | |  |  |
| void | | ***Color3ui*** | ([in] GLuint red,  GLuint green,  GLuint blue); |
|  | |  |  |
| void | | ***Color3uiv*** | ([in] const GLuint \*v); |
|  | |  |  |
| void | | ***Color3us*** | ([in] GLushort red,  GLushort green,  GLushort blue); |
|  | |  |  |
| void | | ***Color3usv*** | ([in] const GLushort \*v); |
|  | |  |  |
| void | | ***Color4b*** | ([in] GLbyte red,  GLbyte green,  GLbyte blue,  GLbyte alpha); |
|  | |  |  |
| void | | ***Color4bv*** | ([in] const GLbyte \*v); |
|  | |  |  |
| void | | ***Color4d*** | ([in] GLdouble red,  GLdouble green,  GLdouble blue,  GLdouble alpha); |
|  | |  |  |
| void | | ***Color4dv*** | ([in] const GLdouble \*v); |
|  | |  |  |
| void | | ***Color4f*** | ([in] GLfloat red,  GLfloat green,  GLfloat blue,  GLfloat alpha); |
|  | |  |  |
| void | | ***Color4fv*** | ([in] const GLfloat \*v); |
|  | |  |  |
| void | | ***Color4i*** | ([in] GLint red,  GLint green,  GLint blue,  GLint alpha); |
|  | |  |  |
| void | | ***Color4iv*** | ([in] const GLint \*v); |
|  | |  |  |
| void | | ***Color4s*** | ([in] GLshort red,  GLshort green,  GLshort blue,  GLshort alpha); |
|  | |  |  |
| void | | ***Color4sv*** | ([in] const GLshort \*v); |
|  | |  |  |
| void | | ***Color4ub*** | ([in] GLubyte red,  GLubyte green,  GLubyte blue,  GLubyte alpha); |
|  | |  |  |
| void | | ***Color4ubv*** | ([in] const GLubyte \*v); |
|  | |  |  |
| void | | ***Color4ui*** | ([in] GLuint red,  GLuint green,  GLuint blue,  GLuint alpha); |
|  | |  |  |
| void | | ***Color4uiv*** | ([in] const GLuint \*v); |
|  | |  |  |
| void | | ***Color4us*** | ([in] GLushort red,  GLushort green,  GLushort blue,  GLushort alpha); |
|  | |  |  |
| void | | ***Color4usv*** | ([in] const GLushort \*v); |
|  | |  |  |
| void | | ***ColorMask*** | ([in] GLboolean red,  GLboolean green,  GLboolean blue,  GLboolean alpha); |
|  | |  |  |
| void | | ***ColorMaterial*** | ([in] GLenum face,  GLenum mode); |
|  | |  |  |
| void | | ***CopyPixels*** | ([in] GLint x,  GLint y,  GLsizei width,  GLsizei height,  GLenum type); |
|  | |  |  |
| void | | ***CullFace*** | ([in] GLenum mode); |
|  | |  |  |
| void | | ***DeleteLists*** | ([in] GLuint list,  GLsizei range); |
|  | |  |  |
| void | | ***DepthFunc*** | ([in] GLenum func); |
|  | |  |  |
| void | | ***DepthMask*** | ([in] GLboolean flag); |
|  | |  |  |
| void | | ***DepthRange*** | ([in] GLclampd zNear,  GLclampd zFar); |
|  | |  |  |
| void | | ***Disable*** | ([in] GLenum cap); |
|  | |  |  |
| void | | ***DrawBuffer*** | ([in] GLenum mode); |
|  | |  |  |
| void | | ***DrawPixels*** | ([in] GLsizei width,  GLsizei height,  GLenum format,  GLenum type,  const GLvoid \*pixels); |
|  | |  |  |
| void | | ***EdgeFlag*** | ([in] GLboolean flag); |
|  | |  |  |
| void | | ***EdgeFlagv*** | ([in] const GLboolean \*flag); |
|  | |  |  |
| void | | ***Enable*** | ([in] GLenum cap); |
|  | |  |  |
| void | | ***End*** | ([in\*/ void); |
|  | |  |  |
| void | | ***EndList*** | ([in\*/ void); |
|  | |  |  |
| void | | ***EvalCoord1d*** | ([in] GLdouble u); |
|  | |  |  |
| void | | ***EvalCoord1dv*** | ([in] const GLdouble \*u); |
|  | |  |  |
| void | | ***EvalCoord1f*** | ([in] GLfloat u); |
|  | |  |  |
| void | | ***EvalCoord1fv*** | ([in] const GLfloat \*u); |
|  | |  |  |
| void | | ***EvalCoord2d*** | ([in] GLdouble u,  GLdouble v); |
|  | |  |  |
| void | | ***EvalCoord2dv*** | ([in] const GLdouble \*u); |
|  | |  |  |
| void | | ***EvalCoord2f*** | ([in] GLfloat u,  GLfloat v); |
|  | |  |  |
| void | | ***EvalCoord2fv*** | ([in] const GLfloat \*u); |
|  | |  |  |
| void | | ***EvalMesh1*** | ([in] GLenum mode,  GLint i1,  GLint i2); |
|  | |  |  |
| void | | ***EvalMesh2*** | ([in] GLenum mode,  GLint i1,  GLint i2,  GLint j1,  GLint j2); |
|  | |  |  |
| void | | ***EvalPoint1*** | ([in] GLint i); |
|  | |  |  |
| void | | ***EvalPoint2*** | ([in] GLint i,  GLint j); |
|  | |  |  |
| void | | ***FeedbackBuffer*** | ([in] GLsizei size,  GLenum type,  GLfloat \*buffer); |
|  | |  |  |
| void | | ***Finish*** | ([in\*/ void); |
|  | |  |  |
| void | | ***Flush*** | ([in\*/ void); |
|  | |  |  |
| void | | ***Fogf*** | ([in] GLenum pname,  GLfloat param); |
|  | |  |  |
| void | | ***Fogfv*** | ([in] GLenum pname,  const GLfloat \*params); |
|  | |  |  |
| void | | ***Fogi*** | ([in] GLenum pname,  GLint param); |
|  | |  |  |
| void | | ***Fogiv*** | ([in] GLenum pname,  const GLint \*params); |
|  | |  |  |
| void | | ***FrontFace*** | ([in] GLenum mode); |
|  | |  |  |
| void | | ***Frustum*** | ([in] GLdouble left,  GLdouble right,  GLdouble bottom,  GLdouble top,  GLdouble zNear,  GLdouble zFar); |
|  | |  |  |
| GLuint | | ***GenLists*** | ([in] GLsizei range); |
|  | |  |  |
| void | | ***GetBooleanv*** | ([in] GLenum pname,   GLboolean \*params); |
|  | |  |  |
| void | | ***GetClipPlane*** | ([in] GLenum plane,   GLdouble \*equation); |
| void | | ***GetDoublev*** | ([in] GLenum pname,   GLdouble \*params); |
|  | |  |  |
| GLenum | | ***GetError*** | ([in\*/ void); |
|  | |  |  |
| void | | ***GetFloatv*** | ([in] GLenum pname,   GLfloat \*params); |
|  | |  |  |
| void | | ***GetIntegerv*** | ([in] GLenum pname,   GLint \*params); |
|  | |  |  |
| void | | ***GetLightfv*** | ([in] GLenum light,   GLenum pname,   GLfloat \*params); |
|  | |  |  |
| void | | ***GetLightiv*** | ([in] GLenum light,   GLenum pname,   GLint \*params); |
|  | |  |  |
| void | | ***GetMapdv*** | ([in] GLenum target,   GLenum query,   GLdouble \*v); |
|  | |  |  |
| void | | ***GetMapfv*** | ([in] GLenum target,   GLenum query,   GLfloat \*v); |
|  | |  |  |
| void | | ***GetMapiv*** | ([in] GLenum target,   GLenum query,   GLint \*v); |
|  | |  |  |
| void | | ***GetMaterialfv*** | ([in] GLenum face,   GLenum pname,   GLfloat \*params); |
|  | |  |  |
| void | | ***GetMaterialiv*** | ([in] GLenum face,   GLenum pname,   GLint \*params); |
|  | |  |  |
| void | | ***GetPixelMapfv*** | ([in] GLenum map,   GLfloat \*values); |
|  | |  |  |
| void | | ***GetPixelMapuiv*** | ([in] GLenum map,   GLuint \*values); |
|  | |  |  |
| void | | ***GetPixelMapusv*** | ([in] GLenum map,   GLushort \*values); |
|  | |  |  |
| void | | ***GetPolygonStipple*** | ([in] GLubyte \*mask); |
|  | |  |  |
| const GLubyte\* | | ***GetString*** | ([in] GLenum name); |
|  | |  |  |
| void | | ***GetTexEnvfv*** | ([in] GLenum target,   GLenum pname,   GLfloat \*params); |
|  | |  |  |
| void | | ***GetTexEnviv*** | ([in] GLenum target,   GLenum pname,   GLint \*params); |
|  | |  |  |
| void | | ***GetTexGendv*** | ([in] GLenum coord,   GLenum pname,   GLdouble \*params); |
|  | |  |  |
| void | | ***GetTexGenfv*** | ([in] GLenum coord,   GLenum pname,   GLfloat \*params); |
|  | |  |  |
| void | | ***GetTexGeniv*** | ([in] GLenum coord,   GLenum pname,   GLint \*params); |
|  | |  |  |
| void | | ***GetTexImage*** | ([in] GLenum target,   GLint level,   GLenum format,   GLenum type,   GLvoid \*pixels); |
|  | |  |  |
| void | | ***GetTexLevelParameterfv*** | ([in] GLenum target,   GLint level,   GLenum pname,   GLfloat \*params); |
|  | |  |  |
| void | | ***GetTexLevelParameteriv*** | ([in] GLenum target,   GLint level,   GLenum pname,   GLint \*params); |
|  | |  |  |
| void | | ***GetTexParameterfv*** | ([in] GLenum target,   GLenum pname,   GLfloat \*params); |
|  | |  |  |
| void | | ***GetTexParameteriv*** | ([in] GLenum target,   GLenum pname,   GLint \*params); |
|  | |  |  |
| void | | ***Hint*** | ([in] GLenum target,   GLenum mode); |
|  | |  |  |
| void | | ***IndexMask*** | ([in] GLuint mask); |
|  | |  |  |
| void | | ***Indexd*** | ([in] GLdouble c); |
|  | |  |  |
| void | | ***Indexdv*** | ([in] const GLdouble \*c); |
|  | |  |  |
| void | | ***Indexf*** | ([in] GLfloat c); |
|  | |  |  |
| void | | ***Indexfv*** | ([in] const GLfloat \*c); |
|  | |  |  |
| void | | ***Indexi*** | ([in] GLint c); |
|  | |  |  |
| void | | ***Indexiv*** | ([in] const GLint \*c); |
|  | |  |  |
| void | | ***Indexs*** | ([in] GLshort c); |
|  | |  |  |
| void | | ***Indexsv*** | ([in] const GLshort \*c); |
|  | |  |  |
| void | | ***InitNames*** | ([in\*/ void); |
|  | |  |  |
| GLboolean | | ***IsEnabled*** | ([in] GLenum cap); |
|  | |  |  |
| GLboolean | | ***IsList*** | ([in] GLuint list); |
|  | |  |  |
| void | | ***LightModelf*** | ([in] GLenum pname,   GLfloat param); |
|  | |  |  |
| void | | ***LightModelfv*** | ([in] GLenum pname,   const GLfloat \*params); |
|  | |  |  |
| void | | ***LightModeli*** | ([in] GLenum pname,   GLint param); |
|  | |  |  |
| void | | ***LightModeliv*** | ([in] GLenum pname,   const GLint \*params); |
|  | |  |  |
| void | | ***Lightf*** | ([in] GLenum light,   GLenum pname,   GLfloat param); |
|  | |  |  |
| void | | ***Lightfv*** | ([in] GLenum light,   GLenum pname,   const GLfloat \*params); |
|  | |  |  |
| void | | ***Lighti*** | ([in] GLenum light,   GLenum pname,   GLint param); |
|  | |  |  |
| void | | ***Lightiv*** | ([in] GLenum light,   GLenum pname,   const GLint \*params); |
|  | |  |  |
| void | | ***LineWidth*** | ([in] GLfloat width); |
|  | |  |  |
| void | | ***ListBase*** | ([in] GLuint base); |
|  | |  |  |
| void | | ***LoadIdentity*** | ([in\*/ void); |
|  | |  |  |
| void | | ***LoadMatrixd*** | ([in] const GLdouble \*m); |
|  | |  |  |
| void | | ***LoadMatrixf*** | ([in] const GLfloat \*m); |
|  | |  |  |
| void | | ***LoadName*** | ([in] GLuint name); |
|  | |  |  |
| void | | ***LogicOp*** | ([in] GLenum opcode); |
|  | |  |  |
| void | | ***Map1d*** | ([in] GLenum target,   GLdouble u1,   GLdouble u2,   GLint stride,   GLint order,   const GLdouble \*points); |
|  | |  |  |
| void | | ***Map1f*** | ([in] GLenum target,   GLfloat u1,   GLfloat u2,   GLint stride,   GLint order,   const GLfloat \*points); |
|  | |  |  |
| void | | ***Map2d*** | ([in] GLenum target,   GLdouble u1,   GLdouble u2,   GLint ustride,   GLint uorder,   GLdouble v1,   GLdouble v2,   GLint vstride,   GLint vorder,   const GLdouble \*points); |
|  | |  |  |
| void | | ***Map2f*** | ([in] GLenum target,   GLfloat u1,   GLfloat u2,   GLint ustride,   GLint uorder,   GLfloat v1,   GLfloat v2,   GLint vstride,   GLint vorder,   const GLfloat \*points); |
|  | |  |  |
| void | | ***MapGrid1d*** | ([in] GLint un,   GLdouble u1,   GLdouble u2); |
|  | |  |  |
| void | | ***MapGrid1f*** | ([in] GLint un,   GLfloat u1,   GLfloat u2); |
|  | |  |  |
| void | | ***MapGrid2d*** | ([in] GLint un,   GLdouble u1,   GLdouble u2,   GLint vn,   GLdouble v1,   GLdouble v2); |
|  | |  |  |
| void | | ***MapGrid2f*** | ([in] GLint un,   GLfloat u1,   GLfloat u2,   GLint vn,   GLfloat v1,   GLfloat v2); |
|  | |  |  |
| void | | ***Materialf*** | ([in] GLenum face,   GLenum pname,   GLfloat param); |
|  | |  |  |
| void | | ***Materialfv*** | ([in] GLenum face,   GLenum pname,   const GLfloat \*params); |
|  | |  |  |
| void | | ***Materiali*** | ([in] GLenum face,   GLenum pname,   GLint param); |
|  | |  |  |
| void | | ***Materialiv*** | ([in] GLenum face,   GLenum pname,   const GLint \*params); |
|  | |  |  |
| void | | ***MatrixMode*** | ([in] GLenum mode); |
|  | |  |  |
| void | | ***MultMatrixd*** | ([in] const GLdouble \*m); |
|  | |  |  |
| void | | ***MultMatrixf*** | ([in] const GLfloat \*m); |
|  | |  |  |
| void | | ***NewList*** | ([in] GLuint list,   GLenum mode); |
|  | |  |  |
| void | | ***Normal3b*** | ([in] GLbyte nx,   GLbyte ny,   GLbyte nz); |
|  | |  |  |
| void | | ***Normal3bv*** | ([in] const GLbyte \*v); |
|  | |  |  |
| void | | ***Normal3d*** | ([in] GLdouble nx,   GLdouble ny,   GLdouble nz); |
|  | |  |  |
| void | | ***Normal3dv*** | ([in] const GLdouble \*v); |
|  | |  |  |
| void | | ***Normal3f*** | ([in] GLfloat nx,   GLfloat ny,   GLfloat nz); |
|  | |  |  |
| void | | ***Normal3fv*** | ([in] const GLfloat \*v); |
|  | |  |  |
| void | | ***Normal3i*** | ([in] GLint nx,   GLint ny,   GLint nz); |
|  | |  |  |
| void | | ***Normal3iv*** | ([in] const GLint \*v); |
|  | |  |  |
| void | | ***Normal3s*** | ([in] GLshort nx,   GLshort ny,   GLshort nz); |
|  | |  |  |
| void | | ***Normal3sv*** | ([in] const GLshort \*v); |
|  | |  |  |
| void | | ***Ortho*** | ([in] GLdouble left,   GLdouble right,   GLdouble bottom,   GLdouble top,   GLdouble zNear,   GLdouble zFar); |
|  | |  |  |
| void | | ***PassThrough*** | ([in] GLfloat token); |
|  | |  |  |
| void | | ***PixelMapfv*** | ([in] GLenum map,   GLsizei mapsize,   const GLfloat \*values); |
|  | |  |  |
| void | | ***PixelMapuiv*** | ([in] GLenum map,   GLsizei mapsize,   const GLuint \*values); |
|  | |  |  |
| void | | ***PixelMapusv*** | ([in] GLenum map,   GLsizei mapsize,   const GLushort \*values); |
|  | |  |  |
| void | | ***PixelStoref*** | ([in] GLenum pname,   GLfloat param); |
|  | |  |  |
| void | | ***PixelStorei*** | ([in] GLenum pname,   GLint param); |
|  | |  |  |
| void | | ***PixelTransferf*** | ([in] GLenum pname,   GLfloat param); |
|  | |  |  |
| void | | ***PixelTransferi*** | ([in] GLenum pname,   GLint param); |
|  | |  |  |
| void | | ***PixelZoom*** | ([in] GLfloat xfactor,   GLfloat yfactor); |
|  | |  |  |
| void | | ***PointSize*** | ([in] GLfloat size); |
|  | |  |  |
| void | | ***PolygonMode*** | ([in] GLenum face,   GLenum mode); |
|  | |  |  |
| void | | ***PolygonStipple*** | ([in] const GLubyte \*mask); |
|  | |  |  |
| void | | ***PopAttrib*** | ([in\*/ void); |
|  | |  |  |
| void | | ***PopMatrix*** | ([in\*/ void); |
|  | |  |  |
| void | | ***PopName*** | ([in\*/ void); |
|  | |  |  |
| void | | ***PushAttrib*** | ([in] GLbitfield mask); |
|  | |  |  |
| void | | ***PushMatrix*** | ([in\*/ void); |
|  | |  |  |
| void | | ***PushName*** | ([in] GLuint name); |
|  | |  |  |
| void | | ***RasterPos2d*** | ([in] GLdouble x,   GLdouble y); |
|  | |  |  |
| void | | ***RasterPos2dv*** | ([in] const GLdouble \*v); |
|  | |  |  |
| void | | ***RasterPos2f*** | ([in] GLfloat x,   GLfloat y); |
|  | |  |  |
| void | | ***RasterPos2fv*** | ([in] const GLfloat \*v); |
|  | |  |  |
| void | | ***RasterPos2i*** | ([in] GLint x,   GLint y); |
|  | |  |  |
| void | | ***RasterPos2iv*** | ([in] const GLint \*v); |
|  | |  |  |
| void | | ***RasterPos2s*** | ([in] GLshort x,   GLshort y); |
|  | |  |  |
| void | | ***RasterPos2sv*** | ([in] const GLshort \*v); |
|  | |  |  |
| void | | ***RasterPos3d*** | ([in] GLdouble x,   GLdouble y,   GLdouble z); |
|  | |  |  |
| void | | ***RasterPos3dv*** | ([in] const GLdouble \*v); |
|  | |  |  |
| void | | ***RasterPos3f*** | ([in] GLfloat x,   GLfloat y,   GLfloat z); |
|  | |  |  |
| void | | ***RasterPos3fv*** | ([in] const GLfloat \*v); |
|  | |  |  |
| void | | ***RasterPos3i*** | ([in] GLint x,   GLint y,   GLint z); |
|  | |  |  |
| void | | ***RasterPos3iv*** | ([in] const GLint \*v); |
|  | |  |  |
| void | | ***RasterPos3s*** | ([in] GLshort x,   GLshort y,   GLshort z); |
|  | |  |  |
| void | | ***RasterPos3sv*** | ([in] const GLshort \*v); |
|  | |  |  |
| void | | ***RasterPos4d*** | ([in] GLdouble x,   GLdouble y,   GLdouble z,   GLdouble w); |
|  | |  |  |
| void | | ***RasterPos4dv*** | ([in] const GLdouble \*v); |
|  | |  |  |
| void | | ***RasterPos4f*** | ([in] GLfloat x,   GLfloat y,   GLfloat z,   GLfloat w); |
|  | |  |  |
| void | | ***RasterPos4fv*** | ([in] const GLfloat \*v); |
|  | |  |  |
| void | | ***RasterPos4i*** | ([in] GLint x,   GLint y,   GLint z,   GLint w); |
|  | |  |  |
| void | | ***RasterPos4iv*** | ([in] const GLint \*v); |
|  | |  |  |
| void | | ***RasterPos4s*** | ([in] GLshort x,   GLshort y,   GLshort z,   GLshort w); |
|  | |  |  |
| void | | ***RasterPos4sv*** | ([in] const GLshort \*v); |
|  | |  |  |
| void | | ***ReadBuffer*** | ([in] GLenum mode); |
|  | |  |  |
| void | | ***ReadPixels*** | ([in] GLint x,   GLint y,   GLsizei width,   GLsizei height,   GLenum format,   GLenum type,   GLvoid \*pixels); |
|  | |  |  |
| void | | ***Rectd*** | ([in] GLdouble x1,   GLdouble y1,   GLdouble x2,   GLdouble y2); |
|  | |  |  |
| void | | ***Rectdv*** | ([in] const GLdouble \*v1,   const GLdouble \*v2); |
|  | |  |  |
| void | | ***Rectf*** | ([in] GLfloat x1,   GLfloat y1,   GLfloat x2,   GLfloat y2); |
|  | |  |  |
| void | | ***Rectfv*** | ([in] const GLfloat \*v1,   const GLfloat \*v2); |
|  | |  |  |
| void | | ***Recti*** | ([in] GLint x1,   GLint y1,   GLint x2,   GLint y2); |
|  | |  |  |
| void | | ***Rectiv*** | ([in] const GLint \*v1,   const GLint \*v2); |
|  | |  |  |
| void | | ***Rects*** | ([in] GLshort x1,   GLshort y1,   GLshort x2,   GLshort y2); |
|  | |  |  |
| void | | ***Rectsv*** | ([in] const GLshort \*v1,   const GLshort \*v2); |
|  | |  |  |
| GLint | | ***RenderMode*** | ([in] GLenum mode); |
|  | |  |  |
| void | | ***Rotated*** | ([in] GLdouble angle,   GLdouble x,   GLdouble y,   GLdouble z); |
|  | |  |  |
| void | | ***Rotatef*** | ([in] GLfloat angle,   GLfloat x,   GLfloat y,   GLfloat z); |
|  | |  |  |
| void | | ***Scaled*** | ([in] GLdouble x,   GLdouble y,   GLdouble z); |
|  | |  |  |
| void | | ***Scalef*** | ([in] GLfloat x,   GLfloat y,   GLfloat z); |
|  | |  |  |
| void | | ***Scissor*** | ([in] GLint x,   GLint y,   GLsizei width,   GLsizei height); |
|  | |  |  |
| void | | ***SelectBuffer*** | ([in] GLsizei size,   GLuint \*buffer); |
|  | |  |  |
| void | | ***ShadeModel*** | ([in] GLenum mode); |
|  | |  |  |
| void | | ***StencilFunc*** | ([in] GLenum func,   GLint ref,   GLuint mask); |
|  | |  |  |
| void | | ***StencilOp*** | ([in] GLenum fail,   GLenum zfail,   GLenum zpass); |
|  | |  |  |
| void | | ***TexCoord1d*** | ([in] GLdouble s); |
|  | |  |  |
| void | | ***TexCoord1dv*** | ([in] const GLdouble \*v); |
|  | |  |  |
| void | | ***TexCoord1f*** | ([in] GLfloat s); |
|  | |  |  |
| void | | ***TexCoord1fv*** | ([in] const GLfloat \*v); |
|  | |  |  |
| void | | ***TexCoord1i*** | ([in] GLint s); |
|  | |  |  |
| void | | ***TexCoord1iv*** | ([in] const GLint \*v); |
|  | |  |  |
| void | | ***TexCoord1s*** | ([in] GLshort s); |
|  | |  |  |
| void | | ***TexCoord1sv*** | ([in] const GLshort \*v); |
|  | |  |  |
| void | | ***TexCoord2d*** | ([in] GLdouble s,   GLdouble t); |
|  | |  |  |
| void | | ***TexCoord2dv*** | ([in] const GLdouble \*v); |
|  | |  |  |
| void | | ***TexCoord2f*** | ([in] GLfloat s,   GLfloat t); |
|  | |  |  |
| void | | ***TexCoord2fv*** | ([in] const GLfloat \*v); |
|  | |  |  |
| void | | ***TexCoord2i*** | ([in] GLint s,   GLint t); |
|  | |  |  |
| void | | ***TexCoord2iv*** | ([in] const GLint \*v); |
|  | |  |  |
| void | | ***TexCoord2s*** | ([in] GLshort s,   GLshort t); |
|  | |  |  |
| void | | ***TexCoord2sv*** | ([in] const GLshort \*v); |
|  | |  |  |
| void | | ***TexCoord3d*** | ([in] GLdouble s,   GLdouble t,   GLdouble r); |
|  | |  |  |
| void | | ***TexCoord3dv*** | ([in] const GLdouble \*v); |
|  | |  |  |
| void | | ***TexCoord3f*** | ([in] GLfloat s,   GLfloat t,   GLfloat r); |
|  | |  |  |
| void | | ***TexCoord3fv*** | ([in] const GLfloat \*v); |
|  | |  |  |
| void | | ***TexCoord3i*** | ([in] GLint s,   GLint t,   GLint r); |
|  | |  |  |
| void | | ***TexCoord3iv*** | ([in] const GLint \*v); |
|  | |  |  |
| void | | ***TexCoord3s*** | ([in] GLshort s,   GLshort t,   GLshort r); |
|  | |  |  |
| void | | ***TexCoord3sv*** | ([in] const GLshort \*v); |
|  | |  |  |
| void | | ***TexCoord4d*** | ([in] GLdouble s,   GLdouble t,   GLdouble r,   GLdouble q); |
|  | |  |  |
| void | | ***TexCoord4dv*** | ([in] const GLdouble \*v); |
|  | |  |  |
| void | | ***TexCoord4f*** | ([in] GLfloat s,   GLfloat t,   GLfloat r,   GLfloat q); |
|  | |  |  |
| void | | ***TexCoord4fv*** | ([in] const GLfloat \*v); |
|  | |  |  |
| void | | ***TexCoord4i*** | ([in] GLint s,   GLint t,   GLint r,   GLint q); |
|  | |  |  |
| void | | ***TexCoord4iv*** | ([in] const GLint \*v); |
|  | |  |  |
| void | | ***TexCoord4s*** | ([in] GLshort s,   GLshort t,   GLshort r,   GLshort q); |
|  | |  |  |
| void | | ***TexCoord4sv*** | ([in] const GLshort \*v); |
|  | |  |  |
| void | | ***TexEnvf*** | ([in] GLenum target,   GLenum pname,   GLfloat param); |
|  | |  |  |
| void | | ***TexEnvfv*** | ([in] GLenum target,   GLenum pname,   const GLfloat \*params); |
|  | |  |  |
| void | | ***TexEnvi*** | ([in] GLenum target,   GLenum pname,   GLint param); |
|  | |  |  |
| void | | ***TexEnviv*** | ([in] GLenum target,   GLenum pname,   const GLint \*params); |
|  | |  |  |
| void | | ***TexGend*** | ([in] GLenum coord,   GLenum pname,   GLdouble param); |
|  | |  |  |
| void | | ***TexGendv*** | ([in] GLenum coord,   GLenum pname,   const GLdouble \*params); |
|  | |  |  |
| void | | ***TexGenf*** | ([in] GLenum coord,   GLenum pname,   GLfloat param); |
|  | |  |  |
| void | | ***TexGenfv*** | ([in] GLenum coord,   GLenum pname,   const GLfloat \*params); |
|  | |  |  |
| void | | ***TexGeni*** | ([in] GLenum coord,   GLenum pname,   GLint param); |
|  | |  |  |
| void | | ***TexGeniv*** | ([in] GLenum coord,   GLenum pname,   const GLint \*params); |
|  | |  |  |
| void | | ***TexImage1D*** | ([in] GLenum target,   GLint level,   GLint internalformat,   GLsizei width,   GLint border,   GLenum format,   GLenum type,   const GLvoid \*pixels); |
|  | |  |  |
| void | | ***TexParameterf*** | ([in] GLenum target,   GLenum pname,   GLfloat param); |
|  | |  |  |
| void | | ***TexParameteri*** | ([in] GLenum target,   GLenum pname,   GLint param); |
|  | |  |  |
| void | | ***TexParameteriv*** | ([in] GLenum target,   GLenum pname,   const GLint \*params); |
|  | |  |  |
| void | | ***Translated*** | ([in] GLdouble x,   GLdouble y,   GLdouble z); |
|  | |  |  |
| void | | ***Translatef*** | ([in] GLfloat x,   GLfloat y,   GLfloat z); |
|  | |  |  |
| void | | ***Vertex2d*** | ([in] GLdouble x,   GLdouble y); |
|  | |  |  |
| void | | ***Vertex2dv*** | ([in] const GLdouble \*v); |
|  | |  |  |
| void | | ***Vertex2f*** | ([in] GLfloat x,   GLfloat y); |
|  | |  |  |
| void | | ***Vertex2fv*** | ([in] const GLfloat \*v); |
|  | |  |  |
| void | | ***Vertex2i*** | ([in] GLint x,   GLint y); |
|  | |  |  |
| void | | ***Vertex2iv*** | ([in] const GLint \*v); |
|  | |  |  |
| void | | ***Vertex2s*** | ([in] GLshort x,   GLshort y); |
|  | |  |  |
| void | | ***Vertex2sv*** | ([in] const GLshort \*v); |
|  | |  |  |
| void | | ***Vertex3d*** | ([in] GLdouble x,   GLdouble y,   GLdouble z); |
|  | |  |  |
| void | | ***Vertex3dv*** | ([in] const GLdouble \*v); |
|  | |  |  |
| void | | ***Vertex3f*** | ([in] GLfloat x,   GLfloat y,   GLfloat z); |
|  | |  |  |
| void | | ***Vertex3fv*** | ([in] const GLfloat \*v); |
|  | |  |  |
| void | | ***Vertex3i*** | ([in] GLint x,   GLint y,   GLint z); |
|  | |  |  |
| void | | ***Vertex3iv*** | ([in] const GLint \*v); |
|  | |  |  |
| void | | ***Vertex3s*** | ([in] GLshort x,   GLshort y,   GLshort z); |
|  | |  |  |
| void | | ***Vertex3sv*** | ([in] const GLshort \*v); |
|  | |  |  |
| void | | ***Vertex4d*** | ([in] GLdouble x,   GLdouble y,   GLdouble z,   GLdouble w); |
|  | |  |  |
| void | | ***Vertex4dv*** | ([in] const GLdouble \*v); |
|  | |  |  |
| void | | ***Vertex4f*** | ([in] GLfloat x,   GLfloat y,   GLfloat z,   GLfloat w); |
|  | |  |  |
| void | | ***Vertex4fv*** | ([in] const GLfloat \*v); |
|  | |  |  |
| void | | ***Vertex4i*** | ([in] GLint x,   GLint y,   GLint z,   GLint w); |
|  | |  |  |
| void | | ***Vertex4iv*** | ([in] const GLint \*v); |
|  | |  |  |
| void | | ***Vertex4s*** | ([in] GLshort x,   GLshort y,   GLshort z,   GLshort w); |
|  | |  |  |
| void | | ***Vertex4sv*** | ([in] const GLshort \*v); |
|  | |  |  |
| void | | ***Viewport*** | ([in] GLint x,   GLint y,   GLsizei width,   GLsizei height); |
| } | |  |  |

* + 1. **Accum function**

The function takes four floating-point arguments that are the values in order to which to set the RGB and A values of the accumulation buffer.

* + 1. **AlphaFunc function**

The function is a symbolic constant indicating the alpha test function.

* + 1. **Begin function**

The function

* + 1. **Bitmap function**

The function

* + 1. **BlendFunc function**

The function **src** indicates how to compute a source blending factor while **dst** indicates how to compute a destination factor

* + 1. **CallList function**

The function calls the list by index **n.**

* + 1. **CallLists function**

The function provides an efficient means for executing a number of display lists.

* + 1. **Clear function**

The function is the bitwise OR of a number of values indicating which buffers

are to be cleared.

* + 1. **ClearAccum function**

The function

* + 1. **ClearColor function**

The function sets the clear value for the color buffers in RGBA mode.

* + 1. **ClearDepth function**

The function

* + 1. **ClearIndex function**

The function sets the clear color index.

* + 1. **ClearStencil function**

The function takes a single integer argument that is the value to which to clear the stencil buffer.

* + 1. **ClipPlane function**

The function

* + 1. **Color3b function**

The function

* + 1. **Color3bv function**

The function

* + 1. **Color3d function**

The function

* + 1. **Color3dv function**

The function

* + 1. **Color3f function**

The function

* + 1. **Color3fv function**

The function

* + 1. **Color3i function**

The function

* + 1. **Color3iv function**

The function

* + 1. **Color3s function**

The function

* + 1. **Color3sv function**

The function

* + 1. **Color3ub function**

The function

* + 1. **Color3ubv function**

The function

* + 1. **Color3ui function**

The function

* + 1. **Color3uiv function**

The function

* + 1. **Color3us function**

The function

* + 1. **Color3usv function**

The function

* + 1. **Color4b function**

The function

* + 1. **Color4bv function**

The function

* + 1. **Color4d function**

The function

* + 1. **Color4dv function**

The function

* + 1. **Color4f function**

The function

* + 1. **Color4fv function**

The function

* + 1. **Color4i function**

The function

* + 1. **Color4iv function**

The function

* + 1. **Color4s function**

The function

* + 1. **Color4sv function**

The function

* + 1. **Color4ub function**

The function

* + 1. **Color4ubv function**

The function

* + 1. **Color4ui function**

The function

* + 1. **Color4uiv function**

The function

* + 1. **Color4us function**

The function

* + 1. **Color4usv function**

The function

* + 1. **ColorMask function**

The function control the color buffer or buffers depending on which bu ers are currently indicated for writing.

* + 1. **ColorMaterial function**

The function

* + 1. **CopyPixels function**

The function CopyPixels transfers a rectangle of pixel values from one region of the framebuffer to another.

* + 1. **CullFace function**

The function

* + 1. **DeleteLists function**

The function where list is the index of the first display list to be deleted and range is the number of display lists to be deleted.

* + 1. **DepthFunc function**

The function takes a single symbolic constant: one of NEVER, ALWAYS, LESS, LEQUAL, EQUAL, GREATER, GEQUAL, NOTEQUAL.

* + 1. **DepthMask function**

The function

* + 1. **DepthRange function**

The function if **mask** is nonzero the depth buffer is enabled for writing otherwise it is disabled. In the initial state the depth buffer is enabled for writing.

* + 1. **Disable function**

The function

* + 1. **DrawBuffer function**

The function

* + 1. **DrawPixels function**

The function

* + 1. **EdgeFlag function**

The function to change the value of a flag bit. If flag is zero then the flag bit is set to FALSE if flag is non-zero then the flag bit is set to TRUE.

* + 1. **EdgeFlagv function**

The function to change the value of a flag bit. If flag is zero then the flag bit is set to FALSE if flag is non-zero then the flag bit is set to TRUE.

* + 1. **Enable function**

The function

* + 1. **End function**

The function

* + 1. **EndList function**

The function

* + 1. **EvalCoord1d function**

The function

* + 1. **EvalCoord1dv function**

The function

* + 1. **EvalCoord1f function**

The function

* + 1. **EvalCoord1fv function**

The function

* + 1. **EvalCoord2d function**

The function

* + 1. **EvalCoord2dv function**

The function

* + 1. **EvalCoord2f function**

The function

* + 1. **EvalCoord2fv function**

The function

* + 1. **EvalMesh1 function**

The function

* + 1. **EvalMesh2 function**

The function

* + 1. **EvalPoint1 function**

The function

* + 1. **EvalPoint2 function**

The function

* + 1. **FeedbackBuffer function**

The function **buffer** is a pointer to an array of floating-point values into which feedback information will be placed and **n** is a number indicating the maximum numberof values that can be written to that array. **type** is a symbolic constant describing the information to be fed back for each vertex

* + 1. **Finish function**

The function forces all previous GL commands to complete.

* + 1. **Flush function**

The function indicates that all commands that have previously been sent to the GL must complete in finite time.

* + 1. **Fogf function**

The function

* + 1. **Fogfv function**

The function

* + 1. **Fogi function**

The function

* + 1. **Fogiv function**

The function

* + 1. **FrontFace function**

The function

* + 1. **Frustum function**

The function

* + 1. **GenLists function**

The function

* + 1. **GetBooleanv function**

The function

* + 1. **GetClipPlane function**

The function always returns four double-precision values in **eqn.**

* + 1. **GetDoublev function**

The function

* + 1. **GetError function**

The function is used to obtain error information.

* + 1. **GetFloatv function**

The function

* + 1. **GetIntegerv function**

The function

* + 1. **GetLightfv function**

The function

* + 1. **GetLightiv function**

The function

* + 1. **GetMapdv function**

The function

* + 1. **GetMapfv function**

The function

* + 1. **GetMapiv function**

The function

* + 1. **GetMaterialfv function**

The function

* + 1. **GetMaterialiv function**

The function

* + 1. **GetPixelMapfv function**

The function

* + 1. **GetPixelMapuiv function**

The function

* + 1. **GetPixelMapusv function**

The function

* + 1. **GetPolygonStipple function**

The function

* + 1. **GetString function**

The function

* + 1. **GetTexEnvfv function**

The function

* + 1. **GetTexEnviv function**

The function

* + 1. **GetTexGendv function**

The function

* + 1. **GetTexGenfv function**

The function

* + 1. **GetTexGeniv function**

The function

* + 1. **GetTexImage function**

The function

* + 1. **GetTexLevelParameterfv function**

The function

* + 1. **GetTexLevelParameteriv function**

The function

* + 1. **GetTexParameterfv function**

The function

* + 1. **GetTexParameteriv function**

The function

* + 1. **Hint function**

The function **target** is a symbolic constant indicating the behavior to be controlled and **hint** is a symbolic constant indicating what type of behavior is desired.

* + 1. **IndexMask function**

The function

* + 1. **Indexd function**

The function

* + 1. **Indexdv function**

The function

* + 1. **Indexf function**

The function

* + 1. **Indexfv function**

The function

* + 1. **Indexi function**

The function

* + 1. **Indexiv function**

The function

* + 1. **Indexs function**

The function

* + 1. **Indexsv function**

The function

* + 1. **InitNames function**

The function empties clears the name stack.

* + 1. **IsEnabled function**

The function

* + 1. **IsList function**

The function returns TRUE if **list** is the index of some display list.

* + 1. **LightModelf function**

The function

* + 1. **LightModelfv function**

The function

* + 1. **LightModeli function**

The function

* + 1. **LightModeliv function**

The function

* + 1. **Lightf function**

The function

* + 1. **Lightfv function**

The function

* + 1. **Lighti function**

The function

* + 1. **Lightiv function**

The function

* + 1. **LineWidth function**

The function

* + 1. **ListBase function**

The function indicating a display list index that does not correspond to any display list has no effect.

* + 1. **LoadIdentity function**

The function effectively calls **LoadMatrix** with the identity matrix.

* + 1. **LoadMatrixd function**

The function

* + 1. **LoadMatrixf function**

The function

* + 1. **LoadName function**

The function replaces the value on the top of the stackwith **name.**

* + 1. **LogicOp function**

The function

* + 1. **Map1d function**

The function

* + 1. **Map1f function**

The function

* + 1. **Map2d function**

The function

* + 1. **Map2f function**

The function

* + 1. **MapGrid1d function**

The function

* + 1. **MapGrid1f function**

The function

* + 1. **MapGrid2d function**

The function

* + 1. **MapGrid2f function**

The function

* + 1. **Materialf function**

The function

* + 1. **Materialfv function**

The function

* + 1. **Materiali function**

The function

* + 1. **Materialiv function**

The function

* + 1. **MatrixMode function**

The function which takes one of the three pre-fefined constants **TEXTURE,** **MODELVIEW** or **PROJECTION** as the argument value.

* + 1. **MultMatrixd function**

The function

* + 1. **MultMatrixf function**

The function

* + 1. **NewList function**

The function

* + 1. **Normal3b function**

The function

* + 1. **Normal3bv function**

The function

* + 1. **Normal3d function**

The function

* + 1. **Normal3dv function**

The function

* + 1. **Normal3f function**

The function

* + 1. **Normal3fv function**

The function

* + 1. **Normal3i function**

The function

* + 1. **Normal3iv function**

The function

* + 1. **Normal3s function**

The function

* + 1. **Normal3sv function**

The function

* + 1. **Ortho function**

The function

* + 1. **PassThrough function**

The function may be used as a marker in feedback mode.

* + 1. **PixelMapfv function**

The function

* + 1. **PixelMapuiv function**

The function

* + 1. **PixelMapusv function**

The function

* + 1. **PixelStoref function**

The function

* + 1. **PixelStorei function**

The function

* + 1. **PixelTransferf function**

The function

* + 1. **PixelTransferi function**

The function

* + 1. **PixelZoom function**

The function

* + 1. **PointSize function**

The function **size** specifies the width or diameter of a point.

* + 1. **PolygonMode function**

The function

* + 1. **PolygonStipple function**

The function obtains the polygon stipple.

* + 1. **PopAttrib function**

The function resets the values of those state variables that were saved with the last **PushAttrib**.

* + 1. **PopMatrix function**

The function pops the top entry off the stack replacing the current matrix with the matrix that was the second entry in the stack.

* + 1. **PopName function**

The function pops one name o the top of the name stack.

* + 1. **PushAttrib function**

The function takes a bitwise OR of symbolic constants indicating which groups of state variables to push onto an attribute stack.

* + 1. **PushMatrix function**

The function pushes the stack down by one duplicating the current matrix in both the top of the stack and the entry below it,

* + 1. **PushName function**

The function causes name to be pushed onto the name stack.

* + 1. **RasterPos2d function**

The function

* + 1. **RasterPos2dv function**

The function

* + 1. **RasterPos2f function**

The function

* + 1. **RasterPos2fv function**

The function

* + 1. **RasterPos2i function**

The function

* + 1. **RasterPos2iv function**

The function

* + 1. **RasterPos2s function**

The function

* + 1. **RasterPos2sv function**

The function

* + 1. **RasterPos3d function**

The function

* + 1. **RasterPos3dv function**

The function

* + 1. **RasterPos3f function**

The function

* + 1. **RasterPos3fv function**

The function

* + 1. **RasterPos3i function**

The function

* + 1. **RasterPos3iv function**

The function

* + 1. **RasterPos3s function**

The function

* + 1. **RasterPos3sv function**

The function

* + 1. **RasterPos4d function**

The function

* + 1. **RasterPos4dv function**

The function

* + 1. **RasterPos4f function**

The function

* + 1. **RasterPos4fv function**

The function

* + 1. **RasterPos4i function**

The function

* + 1. **RasterPos4iv function**

The function

* + 1. **RasterPos4s function**

The function

* + 1. **RasterPos4sv function**

The function

* + 1. **ReadBuffer function**

The function

* + 1. **ReadPixels function**

The function

* + 1. **Rectd function**

The function

* + 1. **Rectdv function**

The function

* + 1. **Rectf function**

The function

* + 1. **Rectfv function**

The function

* + 1. **Recti function**

The function

* + 1. **Rectiv function**

The function

* + 1. **Rects function**

The function

* + 1. **Rectsv function**

The function

* + 1. **RenderMode function**

The function

* + 1. **Rotated function**

The function

* + 1. **Rotatef function**

The function

* + 1. **Scaled function**

The function

* + 1. **Scalef function**

The function

* + 1. **Scissor function**

The function

* + 1. **SelectBuffer function**

The function

* + 1. **ShadeModel function**

The function **mode** value must be either of the symbolic constants **SMOOTH** or **FLAT.**

* + 1. **StencilFunc function**

The function

* + 1. **StencilOp function**

The function

* + 1. **TexCoord1d function**

The function

* + 1. **TexCoord1dv function**

The function

* + 1. **TexCoord1f function**

The function

* + 1. **TexCoord1fv function**

The function

* + 1. **TexCoord1i function**

The function

* + 1. **TexCoord1iv function**

The function

* + 1. **TexCoord1s function**

The function

* + 1. **TexCoord1sv function**

The function

* + 1. **TexCoord2d function**

The function

* + 1. **TexCoord2dv function**

The function

* + 1. **TexCoord2f function**

The function

* + 1. **TexCoord2fv function**

The function

* + 1. **TexCoord2i function**

The function

* + 1. **TexCoord2iv function**

The function

* + 1. **TexCoord2s function**

The function

* + 1. **TexCoord2sv function**

The function

* + 1. **TexCoord3d function**

The function

* + 1. **TexCoord3dv function**

The function

* + 1. **TexCoord3f function**

The function

* + 1. **TexCoord3fv function**

The function

* + 1. **TexCoord3i function**

The function

* + 1. **TexCoord3iv function**

The function

* + 1. **TexCoord3s function**

The function

* + 1. **TexCoord3sv function**

The function

* + 1. **TexCoord4d function**

The function

* + 1. **TexCoord4dv function**

The function

* + 1. **TexCoord4f function**

The function

* + 1. **TexCoord4fv function**

The function

* + 1. **TexCoord4i function**

The function

* + 1. **TexCoord4iv function**

The function

* + 1. **TexCoord4s function**

The function

* + 1. **TexCoord4sv function**

The function

* + 1. **TexEnvf function**

The function

* + 1. **TexEnvfv function**

The function

* + 1. **TexEnvi function**

The function

* + 1. **TexEnviv function**

The function

* + 1. **TexGend function**

The function

* + 1. **TexGendv function**

The function

* + 1. **TexGenf function**

The function

* + 1. **TexGenfv function**

The function

* + 1. **TexGeni function**

The function

* + 1. **TexGeniv function**

The function

* + 1. **TexImage1D function**

The function

* + 1. **TexParameterf function**

The function

* + 1. **TexParameteri function**

The function

* + 1. **TexParameteriv function**

The function

* + 1. **Translated function**

The function

* + 1. **Translatef function**

The function give the coordinates of a translation vector as (x y z)^T.

* + 1. **Vertex2d function**

The function

* + 1. **Vertex2dv function**

The function

* + 1. **Vertex2f function**

The function

* + 1. **Vertex2fv function**

The function

* + 1. **Vertex2i function**

The function

* + 1. **Vertex2iv function**

The function

* + 1. **Vertex2s function**

The function

* + 1. **Vertex2sv function**

The function

* + 1. **Vertex3d function**

The function

* + 1. **Vertex3dv function**

The function

* + 1. **Vertex3f function**

The function

* + 1. **Vertex3fv function**

The function

* + 1. **Vertex3i function**

The function

* + 1. **Vertex3iv function**

The function

* + 1. **Vertex3s function**

The function

* + 1. **Vertex3sv function**

The function

* + 1. **Vertex4d function**

The function

* + 1. **Vertex4dv function**

The function

* + 1. **Vertex4f function**

The function

* + 1. **Vertex4fv function**

The function

* + 1. **Vertex4i function**

The function

* + 1. **Vertex4iv function**

The function

* + 1. **Vertex4s function**

The function

* + 1. **Vertex4sv function**

The function

* + 1. **Viewport function**

The function

1. **Error codes**

The following table contains the error codes.

|  |  |  |
| --- | --- | --- |
| **Error code** | **Value** | **Description** |
| ERR\_ECO\_SUCCESES | 0x0000 | Operation successful. |
| ERR\_ECO\_UNEXPECTED | 0xFFFF | Unexpected condition. |
| ERR\_ECO\_POINTER | 0xFFEE | NULL was passed incorrectly for a pointer value. |
| ERR\_ECO\_NOINTERFACE | 0xFFED | No such interface supported. |
| ERR\_ECO\_COMPONENT\_NOTFOUND | 0xFFE9 | The component was not found. |
|  |  |  |
|  |  |  |

# **Appendix A: Training programs**