
Subject: Get OpenGL Version

Posted by [BlackEric](#) on Thu, 29 Sep 2022 06:48:24 GMT

[View Forum Message](#) <> [Reply to Message](#)

unit Unit1;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
Dialogs, OpenGL, StdCtrls, ExtCtrls;

type

TForm1 = class(TForm)
 Memo1: TMemo;
 Panel1: TPanel;
 Button1: TButton;
 procedure Button1Click(Sender: TObject);
 procedure FormCreate(Sender: TObject);
 procedure FormDestroy(Sender: TObject);
 procedure FormPaint(Sender: TObject);

private

 { Private declarations }
 glContext: HGLRC;
 glDC: HDC;
 errorCode: GLenum;
 openGLReady: Boolean;

public

 { Public declarations }
end;

var

 Form1: TForm1;

implementation

{ \$R *.dfm }

procedure TForm1.Button1Click(Sender: TObject);

begin

 Memo1.Lines.Clear;

 Memo1.Lines.Add('OpenGL vendor: ' + glGetString(GL_VENDOR));

 Memo1.Lines.Add('OpenGL renderer: ' + glGetString(GL_RENDERER));

 Memo1.Lines.Add('OpenGL version: ' + glGetString(GL_VERSION));

```

Memo1.Lines.Add('Supported OpenGL extensions: ' + glGetString(GL_EXTENSIONS));
end;

procedure TForm1.FormCreate(Sender: TObject);
var
  pfd: TPixelFormatDescriptor;
  FormatIndex: integer;
begin
  fillchar(pfd, SizeOf(pfd), 0);
  with pfd do
  begin
    nSize := SizeOf(pfd);
    nVersion := 1; { The current version of the descriptor is 1 }
    dwFlags := PFD_DRAW_TO_WINDOW or PFD_SUPPORT_OPENGL;
    iPixelFormat := PFD_TYPE_RGBA;
    cColorBits := 24; { support 24-bit color }
    cDepthBits := 32; { depth of z-axis }
    iLayerType := PFD_MAIN_PLANE;
  end; { with }
  glDC := getDC(handle);
  FormatIndex := ChoosePixelFormat(glDC, @pfd);
  if FormatIndex = 0 then
    raise Exception.Create('ChoosePixelFormat failed ' + IntToStr(GetLastError)
    );
  if not SetPixelFormat(glDC, FormatIndex, @pfd) then
    raise Exception.Create('SetPixelFormat failed ' + IntToStr(GetLastError));
  glContext := wglCreateContext(glDC);
  if glContext = 0 then
    raise Exception.Create('wglCreateContext failed ' + IntToStr(GetLastError));
  if not wglMakeCurrent(glDC, glContext) then
    raise Exception.Create('wglMakeCurrent failed ' + IntToStr(GetLastError));
  openGLReady := true;

end;

procedure TForm1.FormDestroy(Sender: TObject);
begin
  wglMakeCurrent(Canvas.handle, 0);
  wglDeleteContext(glContext);
end;

procedure TForm1.FormPaint(Sender: TObject);
begin
  { if not openGLReady then
    exit;
  //background
  glClearColor(0.1,0.4,0.0,0.0);
  glClear(GL_COLOR_BUFFER_BIT);

```

```
//error checking
errorCode:=glGetError;
if errorCode<>GL_NO_ERROR then
raise Exception.Create('Error in Paint'#13+gluErrorString(errorCode));
glFlush; }
end;

end.
```
