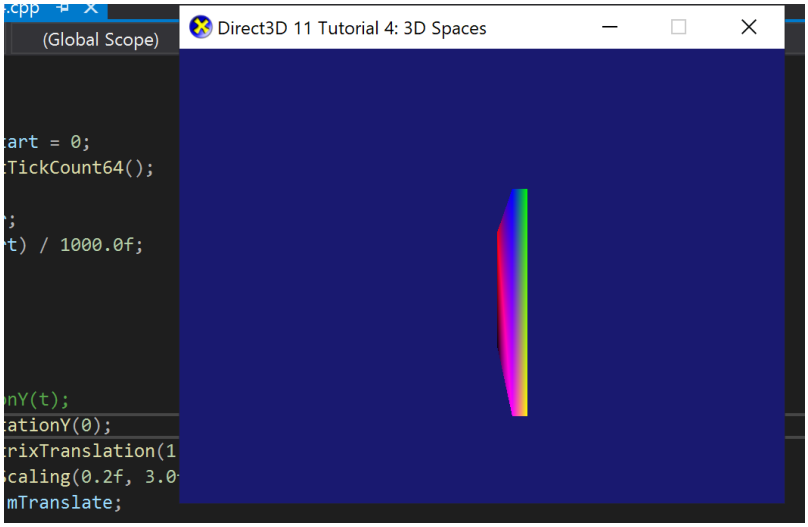
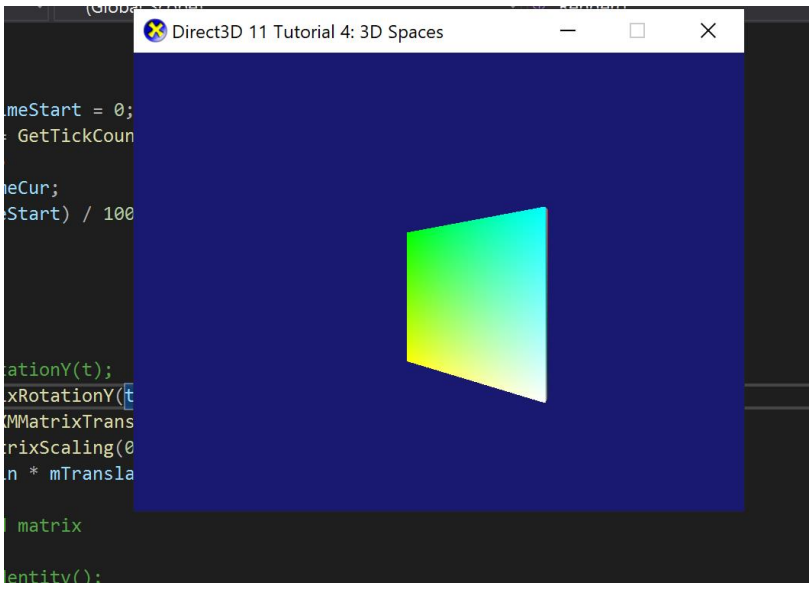
Real Time Graphics Lab D.

Week 4 – Lab D

Exercise 1. Create your own vertex shaders

Sample Output:





Test Data:

N/A

Solution:

```
VS_OUTPUT VS_main(float4 Pos : POSITION, float4 Color : COLOR)
{
    VS_OUTPUT output = (VS_OUTPUT)0;
    float4 inPos = Pos;
    float3 translation = float3(1.0f, 0.3f, 1.0f);
    float3 scale = float3(0.2f, 3.0f, 3.0f);

    float angle = 1.3748;

    float3x3 rotationMatrix = { cos(angle), 0.0f, sin(angle), 0.0f, 1.0f, 0.0f, -sin(angle), 0.0f, cos(angle) };

    inPos.xyz = (scale * (mul(inPos.xyz, rotationMatrix))) + translation;
    output.Pos = mul(Pos, World);
    output.Pos = mul(output.Pos, View);
    output.Pos = mul(output.Pos, Projection);
    output.Color = Color;
    return output;
}
```

```
if (g_pVertexShader_1) g_pVertexShader_1->Release();

//
// Renders a triangle
//
g_pImmediateContext->VSSetShader(g_pVertexShader_1, nullptr, 0);
```

```
hr = CompileShaderFromFile(L"Tutorial04.fxh", "VS_main", "vs_4_0", &pVSBlob);
if (FAILED(hr))
{
    MessageBox(nullptr,
        L"The FX file cannot be compiled.  Please run this executable from the directory that contains the FX file.", L"Error", MB_OK);
    return hr;
}

// Create the vertex shader
hr = g_pd3dDevice->CreateVertexShader(pVSBlob->GetBufferPointer(), pVSBlob->GetBufferSize(), nullptr, &g_pVertexShader_1);
if (FAILED(hr))
{
    pVSBlob->Release();
    return hr;
}
```

```
ID3D11VertexShader* g_pVertexShader_1 = nullptr;
```

Reflection:

This exercise was quite straight forward assuming I understood the task, after doing all that was stated in the exercise the resulting output is shown above.

Meta data:

Vertex Shader

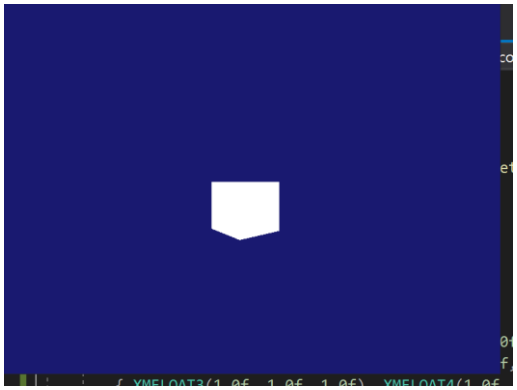
Further Information:

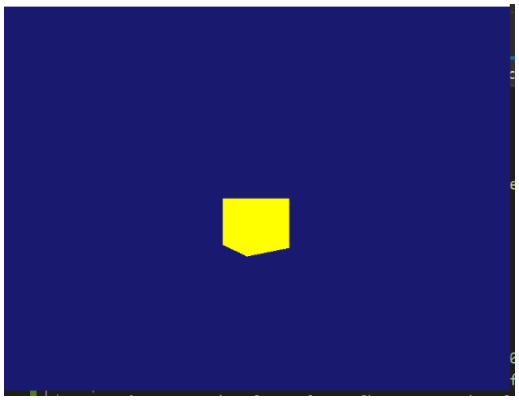
Exercise 2. Create your own pixel shaders

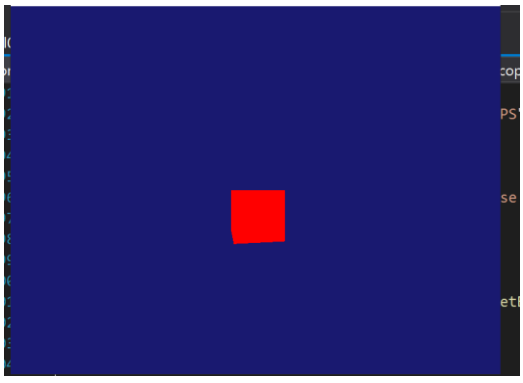
Solution:

Test Data:

Sample Output:







Reflection:

This was a bit tricky and confusing, I tried using the vertex shader to produce cubes of different colours but could not, I did it the other way around and saw results. I hope I did the right thing.

Meta data:

Pixel Shader

Further Information:

Exercise 3. Cornell box in vertex shaders

Sample Output:

Test Data:

Sample Output:

Reflection:

Meta data:

Cornell

Further Information:

Exercise 4. Define Model-View-Projection in vertex shader (optional)

Exercise 5. Vertex shader point cloud (Optional)ers

Exercise 6. Per-vertex diffuse lighting

Solution:

```
VS_OUTPUT VS_main(float4 Pos : POSITION, float4 Color : COLOR, float3 N : NORMAL)
{
    VS_OUTPUT output = (VS_OUTPUT)0;
    output.Pos = mul(Pos, World);
    output.Pos = mul(output.Pos, View);
    output.Pos = mul(output.Pos, Projection);

    float4 materialAmb = float4(0.1, 0.2, 0.2, 1.0);
    float4 materialDiff = float4(0.9, 0.7, 1.0, 1.0);
    float4 lightCol = float4(1.0, 0.6, 0.8, 1.0);
    float3 lightDir = normalize(lightPos.xyz - Pos.xyz);
    float3 normal = normalize(N);
    float diff = max(0.0, dot(lightDir, normal));
    output.Color = (materialAmb + diff * materialDiff) * lightCol;
    output.PosWorld = Pos.xyz;
    output.Norm = N.xyz;

   /* if (Norm.z == -1.0f)
    {
        output.Color = float4(1.0f, 1.0f, 1.0f, 1.0f);
```

Test Data:

Sample Output:

Reflection:

I was able to get the code but could not see the output, I guess somethings are mixed up and I cannot seem to know how to fix this..

Meta data:

Diffuse Lightning

Further Information:

Exercise 7. Per-pixel diffuse lighting

Solution:

```
float4 PS(VS_OUTPUT input) : SV_Target
{

    float4 finalColour = 0;
    float4 materialAmb = float4(0.1f, 0.2f, 0.2f, 1.0f);
    float4 materialDiff = float4(1.0f, 0.0f, 1.0f, 1.0f);
    float4 lightCol = float4(1.0f, 0.6f, 0.8f, 1.0f);

    float3 lightDir = normalize(lightPos.xyz - Pos.xyz);
    float3 normal = normalize(N);

    float diff = max(0.0f, dot(lightDir, normal));

    finalColour = aterialAmb + (diff * material) * lightCol;

        finalColour.a = 1;
    //return float4(1.0f, 0.0f, 0.0f, 1.0f);
    return input.Color;
}
```

Test Data:

Sample Output:

Reflection:

My code is not working after several hours of debugging, attached is the error response from the compiler.

Code	Description	Project	File	Line	Supp
X3004	undeclared identifier 'lightPos'	Tutorial04	Tutorial04.fxh	51	

Meta data:

Diffuse Lighting

Further Information:

Exercise 8. Per-pixel specular lighting

Sample Output:

Test Data:

Sample Output:

Reflection:

Meta data:

Further Information:

Exercise 9. Multiple materials and light sources

Sample Output:

Test Data:

Sample Output:

Reflection:

Meta data:

Further Information: