```
};

// Output structure describes the vertex that is
// output from the shader.  Here the output
// vertex contains a position and color component.
struct VS_OUTPUT
{
    vector position : POSITION;
    vector diffuse  : COLOR;
};

//
// Main Entry Point, observe the main function
// receives a copy of the input vertex through
// its parameter and returns a copy of the output
// vertex it computes.
//

VS_OUTPUT Main(VS_INPUT input)
{
    // zero out members of output
    VS_OUTPUT output = (VS_OUTPUT)0;

    // transform to view space and project
    output.position  = mul(input.position, ViewProjMatrix);

    // set vertex diffuse color to blue
    output.diffuse = Blue;

    return output;
}
```

16.1.1: المتحولات العامة

أولاً نقوم بتعريف متحولين عامين:

```
matrix ViewProjMatrix;
vector Blue = {0.0f, 0.0f, 1.0f, 1.0f};
```

المتحول الأول ViewProjMatrix من النوع matrix (والذي هو نوع لمصفوفة 4×4 معرف ضمنياً في HLSL)، سيخزن هذا المتحول مصفوفة العرض والإسقاط المركبة بحيث يصف هذين التحويلين النقطيين معاً. بهذه الطريقة سيكون علينا إجراء عملية ضرب شعاع بمصفوفة واحدة بدلاً من عمليتين. لاحظ بأننا لا نهيء هذا المتحول في أي مكان