Fragment Shader Report

Code Summary:

The provided fragment shader is a simple program that defines a single function, main(), which determines the color output of a fragment (a pixel) on the screen.

Code Analysis:

Precision Declaration:

“precision mediump float;”

This line declares the precision of the floating-point values used in the shader. mediump is a medium level of precision, suitable for most graphics operations.

Main Function:

“void main() {

gl\_FragColor = vec4(0.0, 0.0, 1.0, 1.0);

}”

void main(): This is the entry point of the shader program. It's executed for each pixel (fragment) of the rendered object.

gl\_FragColor: This is a predefined variable that represents the output color of the fragment. It's a vec4 type, which is a 4-component vector representing red, green, blue, and alpha values.

“vec4(1.0, 0.0, 0.0, 1.0)”

This line assigns a color to gl\_FragColor.

It was vec4(1.0, 0.0, 0.0, 1.0) before and it was a red square. After changing it into vec4(0.0, 0.0, 1.0, 1.0) it's setting the color to full blue, with no red or green, and fully opaque (alpha value of 1.0).

Conclusion:

This shader is very straightforward. It sets the color of every fragment to blue.