

# inspectHooks()



**This API is experimental**

Its behavior may change in a future version of p5.js.

Logs the hooks available in this shader, and their current implementation.

Each shader may let you override bits of its behavior. Each bit is called a *hook*. A hook is either for the *vertex* shader, if it affects the position of vertices, or in the *fragment* shader, if it affects the pixel color. This method logs those values to the console, letting you know what you are able to use in a call to `modify()`.

For example, this shader will produce the following output:

```
myShader = baseMaterialShader().modify({
  declarations: 'uniform float time;',
  'vec3 getWorldPosition': `(vec3 pos) {
    pos.y += 20. * sin(time * 0.001 + pos.x * 0.05);
    return pos;
  }`
});
myShader.inspectHooks();
```

```
==== Vertex shader hooks: ====
void beforeVertex() {}
vec3 getLocalPosition(vec3 position) { return position; }
[MODIFIED] vec3 getWorldPosition(vec3 pos) {
  pos.y += 20. * sin(time * 0.001 + pos.x * 0.05);
  return pos;
}
vec3 getLocalNormal(vec3 normal) { return normal; }
vec3 getWorldNormal(vec3 normal) { return normal; }
vec2 getUV(vec2 uv) { return uv; }
vec4 getVertexColor(vec4 color) { return color; }
void afterVertex() {}

==== Fragment shader hooks: ====
void beforeFragment() {}
Inputs getPixelInputs(Inputs inputs) { return inputs; }
vec4 combineColors(ColorComponents components) {
  vec4 color = vec4(0.);
  color.rgb += components.diffuse * components.baseColor;
  color.rgb += components.ambient * components.ambientColor;
  color.rgb += components.specular * components.specularColor;
  color.rgb += components.emissive;
  color.a = components.opacity;
  return color;
}
vec4 getFinalColor(vec4 color) { return color; }
void afterFragment() {}
```

This page is generated from the comments in `src/webgl/p5.Shader.js`. Please feel free to edit it and submit a pull request!

## Related References

### copyToContext

Copies the shader from one drawing context to another.

### inspectHooks

Logs the hooks available in this shader, and their current implementation.

### modify

Returns a new shader, based on the original, but with custom snippets of shader code replacing default behaviour.

### setUniform

Sets the shader's uniform (global) variables.

p5.js

Resources

Information

Socials

- Reference
- Tutorials
- Examples
- Contribute
- Community
- About
- Start Coding
- Donate

- Download
- Contact
- Copyright
- Privacy Policy
- Terms of Use

- GitHub ↗
- Instagram ↗
- X ↗
- YouTube ↗
- Discord ↗
- Forum ↗

