

Aqua

Generated by Doxygen 1.9.1

1 Class Index	1
1.1 Class List	1
2 Class Documentation	3
2.1 aq::Fish Class Reference	3
2.2 aq::Force Class Reference	4
2.3 GLSL::PerlinNoise Class Reference	5
2.3.1 Detailed Description	7
2.3.2 Member Function Documentation	7
2.3.2.1 colorFromHeight()	7
2.3.2.2 fractalNoise()	7
2.3.2.3 perlin()	7
2.3.2.4 randomGradient()	8
Index	9

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

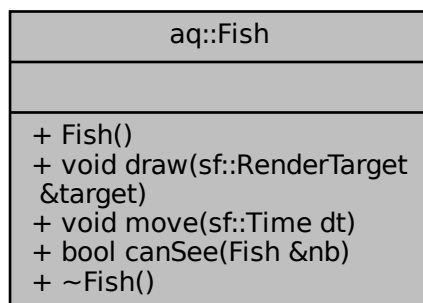
aq::Fish	3
aq::Force	4
GLSL::PerlinNoise Simple 2D perlin noise shader	5

Chapter 2

Class Documentation

2.1 aq::Fish Class Reference

Collaboration diagram for aq::Fish:



Public Member Functions

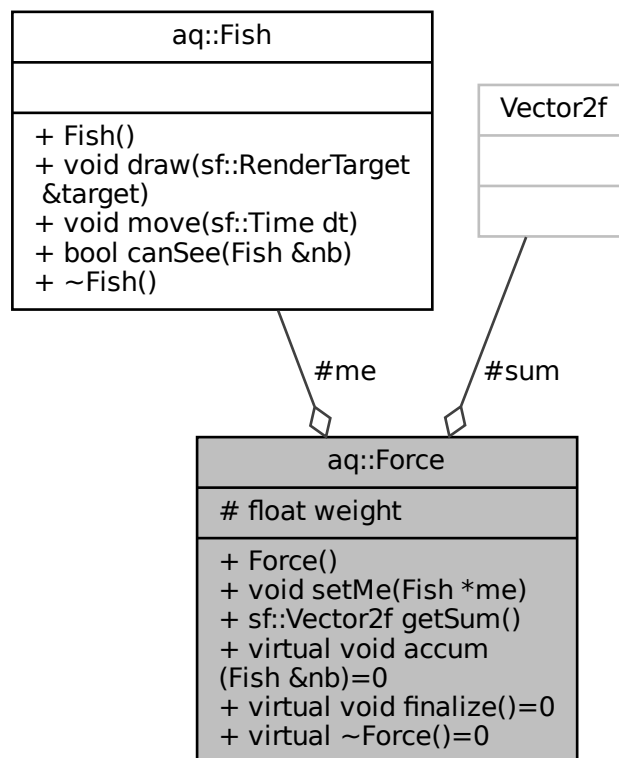
- void **draw** (sf::RenderTarget &target)
- void **move** (sf::Time dt)
- bool **canSee** ([Fish](#) &nb)

The documentation for this class was generated from the following file:

- inc/fish.hpp

2.2 aq::Force Class Reference

Collaboration diagram for aq::Force:



Public Member Functions

- void **setMe** ([Fish](#) *me)
- sf::Vector2f **getSum** ()
- virtual void **accum** ([Fish](#) &nb)=0
- virtual void **finalize** ()=0

Protected Attributes

- [Fish](#) * **me**
- sf::Vector2f **sum**
- float **weight**

The documentation for this class was generated from the following files:

- inc/force.hpp
- src/force.cpp

2.3 GLSL::PerlinNoise Class Reference

Simple 2D perlin noise shader.

Collaboration diagram for GLSL::PerlinNoise:

GLSL::PerlinNoise
<ul style="list-style-type: none"> + uniform vec2 u_seed + uniform int u_octaves + uniform float u_gridSize + uniform float u_amplitude + uniform float u_water_level + uniform float u_sand_level + uniform vec4 col_low_water + uniform vec4 col_high_water + uniform vec4 col_low_sand + uniform vec4 col_high_sand + uniform vec4 col_low_grass + uniform vec4 col_high_grass + uniform vec2 u_resolution + uniform vec2 u_top_left + uniform vec2 u_bottom_right
<ul style="list-style-type: none"> + float interpolate(float a, float b, float w) + float cap(float value) + vec2 randomGradient(ivec2 cord) + float dotGridGradient(ivec2 cord, vec2 pos) + float perlin(vec2 pos) + float fractalNoise(vec2 pos) + vec4 colorFromHeight(float height) + void main()

Public Member Functions

- float [interpolate](#) (float a, float b, float w)
Smoothly interpolates between two values.
- float [cap](#) (float value)
Caps a value between [0, 1].

- vec2 [randomGradient](#) (ivec2 cord)
Computes a pseudo random gradient vector for a given integer coordinate.
- float [dotGridGradient](#) (ivec2 cord, vec2 pos)
Computes the dot product of a random gradient vector and a given position.
- float [perlin](#) (vec2 pos)
2D Perlin noise
- float [fractalNoise](#) (vec2 pos)
Computes a fractal sum of perlin noise.
- vec4 [colorFromHeight](#) (float height)
Computes a color based on the height.
- void [main](#) ()
Main function.

Public Attributes

- uniform vec2 [u_seed](#)
Seed used as offset.
- uniform int [u_octaves](#)
Number of patterns to sum.
- uniform float [u_gridSize](#)
Size of the grid.
- uniform float [u_amplitude](#)
Start amplitude of the noise.
- uniform float [u_water_level](#)
Threshold for water [0, 1].
- uniform float [u_sand_level](#)
Threshold for sand [0, 1].
- uniform vec4 [col_low_water](#)
Color for deep water.
- uniform vec4 [col_high_water](#)
Color for shallow water.
- uniform vec4 [col_low_sand](#)
Color for low sand.
- uniform vec4 [col_high_sand](#)
Color for high sand.
- uniform vec4 [col_low_grass](#)
Color for low grass.
- uniform vec4 [col_high_grass](#)
Color for high grass.
- uniform vec2 [u_resolution](#)
Size of the window.
- uniform vec2 [u_top_left](#)
Top left corner of the visible area.
- uniform vec2 [u_bottom_right](#)
Bottom right corner of the visible area.

2.3.1 Detailed Description

Simple 2D perlin noise shader.

Code based on the the Perlin noise wikipedia page: https://en.wikipedia.org/wiki/Perlin_noise

Remarks

Fragment-Shader

2.3.2 Member Function Documentation

2.3.2.1 colorFromHeight()

```
vec4 GLSL::PerlinNoise::colorFromHeight (
    float height ) [inline]
```

Computes a color based on the height.

Parameters

<i>height</i>	in [0, 1]
---------------	-----------

2.3.2.2 fractalNoise()

```
float GLSL::PerlinNoise::fractalNoise (
    vec2 pos ) [inline]
```

Computes a fractal sum of perlin noise.

Returns

[0, 1]

2.3.2.3 perlin()

```
float GLSL::PerlinNoise::perlin (
    vec2 pos ) [inline]
```

2D Perlin noise

Parameters

<i>pos</i>	Position in 2D space
------------	----------------------

Returns

[-1, 1]

2.3.2.4 randomGradient()

```
vec2 GLSL::PerlinNoise::randomGradient (
    ivec2 cord ) [inline]
```

Computes a pseudo random gradient vector for a given integer coordinate.

Returns

Vector with length 1

The documentation for this class was generated from the following file:

- src/perlin.frag

Index

aq::Fish, [3](#)
aq::Force, [4](#)

colorFromHeight
 GLSL::PerlinNoise, [7](#)

fractalNoise
 GLSL::PerlinNoise, [7](#)

GLSL::PerlinNoise, [5](#)
 colorFromHeight, [7](#)
 fractalNoise, [7](#)
 perlin, [7](#)
 randomGradient, [8](#)

perlin
 GLSL::PerlinNoise, [7](#)

randomGradient
 GLSL::PerlinNoise, [8](#)