My Project

Generated by Doxygen 1.8.5

Tue Mar 10 2015 19:04:05

Contents

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 2

Class Index

2.1 Class List

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

Chapter 3

Class Documentation

3.1 bloc Struct Reference

The bloc struct.

`#include <bloc.hpp>`

Public Member Functions

- bloc (vec2 origine, float Longeur, float Hauteur)

  bloc
- bool estToucher (circle const &balle)

  *estToucher /** *
- bool operator== (bloc const &A)

  operator ==

Public Attributes

- vec2 bas_gauche
- float L
- float **H**

3.1.1 Detailed Description

The bloc struct.

3.1.2 Constructor & Destructor Documentation

3.1.2.1 bloc::bloc (vec2 *origine,* float *Longeur,* float *Hauteur*)

bloc

Parameters

>p0.15	p0.805
origine
Longeur

Hauteur Constructeur d'un bloc

3.1.3 Member Function Documentation

3.1.3.1 bool bloc::estToucher (circle const & *balle*)

estToucher /** *

bloc::estToucher /** *

/** *

Parameters

|>p0.15|p0.805|
balle /** *

Returns

/**

/** *

Parameters

|>p0.15|p0.805|
balle /** *

Returns

permet de determiné si le bloc a été toucher par la balle /**

3.1.3.2 bool bloc::operator== (bloc const & *A*)

operator ==

Parameters

|>p0.15|p0.805|
A

Returns

3.1.4 Member Data Documentation

3.1.4.1 vec2 bloc::bas_gauche

Origine du bloc (en bas à gauche)

3.1.4.2 float bloc::L

L=largeur H=hauteur

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

- bloc.hpp
- bloc.cpp

3.2 cibles Struct Reference

Public Member Functions

- cibles (int N, int nombreLignes)

  cibles
- void **gestionCollision** (circle const &balle)

Public Attributes

- int **nombreCibles**
- std::list< bloc > **briques**

3.2.1 Constructor & Destructor Documentation

3.2.1.1 cibles::cibles (int *N,* int *nombreLignes*)

cibles

Parameters

>p0.15	p0.805
N	
nombreLignes	Constructeur des cibles crée N cibles repartient sur nombreLignes

Constructeur permettant de genérer une matrice de N element qui remplis toute la partie haute de l'élément

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

- cibles.hpp
- cibles.cpp

3.3 circle Struct Reference

`#include <circle.hpp>`

Public Member Functions

- circle ()
- circle (vec2 const &center_param, float radius_center)
- vec2 **getCoord** ()

Public Attributes

- vec2 center
- float radius

3.3.1 Detailed Description

A structure containing parameter of a geometric circle

3.3.2 Constructor & Destructor Documentation

3.3.2.1 circle::circle ( )

Constructor circle (0,0)

3.3.2.2 circle::circle (vec2 const & *center_param,* float *radius_center*)

Constructor circle ({x,y},R)

3.3.3 Member Data Documentation

3.3.3.1 vec2 circle::center

center coordinate

3.3.3.2 float circle::radius

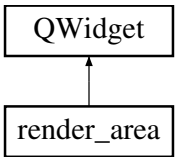
radius coordinate

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

- circle.hpp
- circle.cpp

3.4 render_area Class Reference

Inheritance diagram for render_area:



Public Member Functions

- **render_area** (QWidget ∗parent=0)
- void change_draw_circle_state ()
- void set_damping (float damping)
- void set_bounce_coeff (float bounce_coeff_value)
- void setup_bounce_number (QLabel ∗bounce_number_param)

Protected Member Functions

- void paintEvent (QPaintEvent ∗event)
- void mousePressEvent (QMouseEvent ∗event)
- void mouseMoveEvent (QMouseEvent ∗event)
- void mouseReleaseEvent (QMouseEvent ∗event)

3.4.1 Member Function Documentation

3.4.1.1 void render_area::change_draw_circle_state ( )

Draw or not the circle when called

3.4.1.2 void render_area::mouseMoveEvent (QMouseEvent ∗ *event*) [protected]

Function called when the mouse is moved

3.4.1.3 void render_area::mousePressEvent (QMouseEvent ∗ *event*) [protected]

Function called when the mouse is pressed

3.4.1.4 void render_area::mouseReleaseEvent (QMouseEvent ∗ *event*) [protected]

Function called when the button of the mouse is released

3.4.1.5 void render_area::paintEvent (QPaintEvent ∗ *event*) [protected]

Actual drawing function

3.4.1.6 void render_area::set_bounce_coeff (float *bounce_coeff_value*)

Set a new bouncing coefficient value

3.4.1.7 void render_area::set_damping (float *damping*)

Set a new damping value

3.4.1.8 void render_area::setup_bounce_number (QLabel ∗ *bounce_number_param*)

Pass the pointer of the label for the bouncing number

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

- render_area.hpp
- render_area.cpp

3.5 vec2 Struct Reference

```
#include <vec2.hpp>
```

Public Member Functions

- vec2 ()
- vec2 (float x_param, float y_param)

Public Attributes

- float x
- float y

3.5.1 Detailed Description

A 2D vector

3.5.2 Constructor & Destructor Documentation

3.5.2.1 vec2::vec2 ( )

Constructor vec (0,0)

3.5.2.2 vec2::vec2 (float *x_param,* float *y_param*)

Constructor vec (x,y)

3.5.3 Member Data Documentation

3.5.3.1 float vec2::x

x coordinate

3.5.3.2 float vec2::y

y coordinate

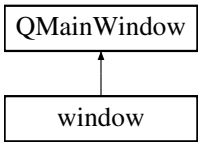
The documentation for this struct was generated from the following files:

- vec2.hpp
- vec2.cpp

3.6 window Class Reference

`#include <window.hpp>`

Inheritance diagram for window:



Public Member Functions

- **window** (QWidget ∗parent=nullptr)

3.6.1 Detailed Description

Declaration of the window class

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

- window.hpp
- window.cpp