SFML

* [Main Page](http://docs.google.com/index.htm)
* [Modules](http://docs.google.com/modules.htm)
* [Classes](http://docs.google.com/annotated.htm)
* [Files](http://docs.google.com/files.htm)
* [Class List](http://docs.google.com/annotated.htm)
* [Class Index](http://docs.google.com/classes.htm)
* [Class Hierarchy](http://docs.google.com/hierarchy.htm)
* [Class Members](http://docs.google.com/functions.htm)
* **sf**
* [View](http://docs.google.com/classsf_1_1View.htm)

[Public Member Functions](#_gjdgxs) | [List of all members](http://docs.google.com/classsf_1_1View-members.htm)

sf::View Class Reference

[Graphics module](http://docs.google.com/group__graphics.htm)

2D camera that defines what region is shown on screen [More...](http://docs.google.com/classsf_1_1View.htm#details)

#include <[View.hpp](http://docs.google.com/View_8hpp_source.htm)>

| Public Member Functions | |
| --- | --- |
|  | [View](http://docs.google.com/classsf_1_1View.htm#a28c38308ff089ae5bdacd001d12286d3) () |
|  | Default constructor. |
|  | |
|  | [View](http://docs.google.com/classsf_1_1View.htm#a1d63bc49e041b3b1ff992bb6430e1326) (const [FloatRect](http://docs.google.com/classsf_1_1Rect.htm) &rectangle) |
|  | Construct the view from a rectangle. |
|  | |
|  | [View](http://docs.google.com/classsf_1_1View.htm#afdaf84cfc910ef160450d63603457ea4) (const [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm) &center, const [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm) &size) |
|  | Construct the view from its center and size. |
|  | |
| void | [setCenter](http://docs.google.com/classsf_1_1View.htm#aa8e3fedb008306ff9811163545fb75f2) (float x, float y) |
|  | Set the center of the view. |
|  | |
| void | [setCenter](http://docs.google.com/classsf_1_1View.htm#ab0296b03793e0873e6ae9e15311f3e78) (const [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm) &center) |
|  | Set the center of the view. |
|  | |
| void | [setSize](http://docs.google.com/classsf_1_1View.htm#a9525b73fe9fbaceb9568faf56b399dab) (float width, float height) |
|  | Set the size of the view. |
|  | |
| void | [setSize](http://docs.google.com/classsf_1_1View.htm#a9e08d471ce21aa0e69ce55ff9de66d29) (const [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm) &size) |
|  | Set the size of the view. |
|  | |
| void | [setRotation](http://docs.google.com/classsf_1_1View.htm#a24d0503c9c51f5ef5918612786d325c1) (float angle) |
|  | Set the orientation of the view. |
|  | |
| void | [setViewport](http://docs.google.com/classsf_1_1View.htm#a8eaec46b7d332fe834f016d0187d4b4a) (const [FloatRect](http://docs.google.com/classsf_1_1Rect.htm) &viewport) |
|  | Set the target viewport. |
|  | |
| void | [reset](http://docs.google.com/classsf_1_1View.htm#ac95b636eafab3922b7e8304fb6c00d7d) (const [FloatRect](http://docs.google.com/classsf_1_1Rect.htm) &rectangle) |
|  | Reset the view to the given rectangle. |
|  | |
| const [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm) & | [getCenter](http://docs.google.com/classsf_1_1View.htm#adae81dede405b91bb3c487e28f536fe9) () const |
|  | Get the center of the view. |
|  | |
| const [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm) & | [getSize](http://docs.google.com/classsf_1_1View.htm#aa130cf34676d715242bee661537a6257) () const |
|  | Get the size of the view. |
|  | |
| float | [getRotation](http://docs.google.com/classsf_1_1View.htm#a8ad320469a27f96f6f49de1c14f0978d) () const |
|  | Get the current orientation of the view. |
|  | |
| const [FloatRect](http://docs.google.com/classsf_1_1Rect.htm) & | [getViewport](http://docs.google.com/classsf_1_1View.htm#af1db1e4f21d104a9691fd38be9165758) () const |
|  | Get the target viewport rectangle of the view. |
|  | |
| void | [move](http://docs.google.com/classsf_1_1View.htm#a0c82144b837caf812f7cb25a43d80c41) (float offsetX, float offsetY) |
|  | Move the view relatively to its current position. |
|  | |
| void | [move](http://docs.google.com/classsf_1_1View.htm#a4c98a6e04fed756dfaff8f629de50862) (const [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm) &offset) |
|  | Move the view relatively to its current position. |
|  | |
| void | [rotate](http://docs.google.com/classsf_1_1View.htm#a5fd3901aae1845586ca40add94faa378) (float angle) |
|  | Rotate the view relatively to its current orientation. |
|  | |
| void | [zoom](http://docs.google.com/classsf_1_1View.htm#a4a72a360a5792fbe4e99cd6feaf7726e) (float factor) |
|  | Resize the view rectangle relatively to its current size. |
|  | |
| const [Transform](http://docs.google.com/classsf_1_1Transform.htm) & | [getTransform](http://docs.google.com/classsf_1_1View.htm#a0109dba48aee769126f670a212b3ed7f) () const |
|  | Get the projection transform of the view. |
|  | |
| const [Transform](http://docs.google.com/classsf_1_1Transform.htm) & | [getInverseTransform](http://docs.google.com/classsf_1_1View.htm#ae7643324b2d8807cf0d9efe9dadfc8cc) () const |
|  | Get the inverse projection transform of the view. |
|  | |

## Detailed Description

2D camera that defines what region is shown on screen

[sf::View](http://docs.google.com/classsf_1_1View.htm) defines a camera in the 2D scene.

This is a very powerful concept: you can scroll, rotate or zoom the entire scene without altering the way that your drawable objects are drawn.

A view is composed of a source rectangle, which defines what part of the 2D scene is shown, and a target viewport, which defines where the contents of the source rectangle will be displayed on the render target (window or texture).

The viewport allows to map the scene to a custom part of the render target, and can be used for split-screen or for displaying a minimap, for example. If the source rectangle has not the same size as the viewport, its contents will be stretched to fit in.

To apply a view, you have to assign it to the render target. Then, every objects drawn in this render target will be affected by the view until you use another view.

Usage example:

[sf::RenderWindow](http://docs.google.com/classsf_1_1RenderWindow.htm) window;

[sf::View](http://docs.google.com/classsf_1_1View.htm) view;

// Initialize the view to a rectangle located at (100, 100) and with a size of 400x200

view.[reset](http://docs.google.com/classsf_1_1View.htm#ac95b636eafab3922b7e8304fb6c00d7d)([sf::FloatRect](http://docs.google.com/classsf_1_1Rect.htm)(100, 100, 400, 200));

// Rotate it by 45 degrees

view.[rotate](http://docs.google.com/classsf_1_1View.htm#a5fd3901aae1845586ca40add94faa378)(45);

// Set its target viewport to be half of the window

view.[setViewport](http://docs.google.com/classsf_1_1View.htm#a8eaec46b7d332fe834f016d0187d4b4a)([sf::FloatRect](http://docs.google.com/classsf_1_1Rect.htm)(0.f, 0.f, 0.5f, 1.f));

// Apply it

window.[setView](http://docs.google.com/classsf_1_1RenderTarget.htm#a063db6dd0a14913504af30e50cb6d946)(view);

// Render stuff

window.[draw](http://docs.google.com/classsf_1_1RenderTarget.htm#a12417a3bcc245c41d957b29583556f39)(someSprite);

// Set the default view back

window.[setView](http://docs.google.com/classsf_1_1RenderTarget.htm#a063db6dd0a14913504af30e50cb6d946)(window.[getDefaultView](http://docs.google.com/classsf_1_1RenderTarget.htm#a718b1aa6296bf855171699cc18251ced)());

// Render stuff not affected by the view

window.[draw](http://docs.google.com/classsf_1_1RenderTarget.htm#a12417a3bcc245c41d957b29583556f39)(someText);

See Also[sf::RenderWindow](http://docs.google.com/classsf_1_1RenderWindow.htm), [sf::RenderTexture](http://docs.google.com/classsf_1_1RenderTexture.htm)

Definition at line [43](http://docs.google.com/View_8hpp_source.htm#l00043) of file [View.hpp](http://docs.google.com/View_8hpp_source.htm).

## Constructor & Destructor Documentation

| sf::View::View | ( |  | ) |  |
| --- | --- | --- | --- | --- |

Default constructor.

This constructor creates a default view of (0, 0, 1000, 1000)

| | sf::View::View | ( | const [FloatRect](http://docs.google.com/classsf_1_1Rect.htm) & | *rectangle* | ) |  | | --- | --- | --- | --- | --- | --- | | explicit |
| --- | --- | --- | --- | --- | --- | --- | --- |

Construct the view from a rectangle.

Parameters

| rectangle | Rectangle defining the zone to display |
| --- | --- |

| sf::View::View | ( | const [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm) & | *center*, |
| --- | --- | --- | --- |
|  |  | const [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm) & | *size* |
|  | ) |  |  |

Construct the view from its center and size.

Parameters

| center | Center of the zone to display |
| --- | --- |
| size | Size of zone to display |

## Member Function Documentation

| const [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm)& sf::View::getCenter | ( |  | ) | const |
| --- | --- | --- | --- | --- |

Get the center of the view.

ReturnsCenter of the view See Also[getSize](http://docs.google.com/classsf_1_1View.htm#aa130cf34676d715242bee661537a6257), [setCenter](http://docs.google.com/classsf_1_1View.htm#aa8e3fedb008306ff9811163545fb75f2)

| const [Transform](http://docs.google.com/classsf_1_1Transform.htm)& sf::View::getInverseTransform | ( |  | ) | const |
| --- | --- | --- | --- | --- |

Get the inverse projection transform of the view.

This function is meant for internal use only.

ReturnsInverse of the projection transform defining the view See Also[getTransform](http://docs.google.com/classsf_1_1View.htm#a0109dba48aee769126f670a212b3ed7f)

| float sf::View::getRotation | ( |  | ) | const |
| --- | --- | --- | --- | --- |

Get the current orientation of the view.

ReturnsRotation angle of the view, in degrees See Also[setRotation](http://docs.google.com/classsf_1_1View.htm#a24d0503c9c51f5ef5918612786d325c1)

| const [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm)& sf::View::getSize | ( |  | ) | const |
| --- | --- | --- | --- | --- |

Get the size of the view.

ReturnsSize of the view See Also[getCenter](http://docs.google.com/classsf_1_1View.htm#adae81dede405b91bb3c487e28f536fe9), [setSize](http://docs.google.com/classsf_1_1View.htm#a9525b73fe9fbaceb9568faf56b399dab)

| const [Transform](http://docs.google.com/classsf_1_1Transform.htm)& sf::View::getTransform | ( |  | ) | const |
| --- | --- | --- | --- | --- |

Get the projection transform of the view.

This function is meant for internal use only.

ReturnsProjection transform defining the view See Also[getInverseTransform](http://docs.google.com/classsf_1_1View.htm#ae7643324b2d8807cf0d9efe9dadfc8cc)

| const [FloatRect](http://docs.google.com/classsf_1_1Rect.htm)& sf::View::getViewport | ( |  | ) | const |
| --- | --- | --- | --- | --- |

Get the target viewport rectangle of the view.

ReturnsViewport rectangle, expressed as a factor of the target size See Also[setViewport](http://docs.google.com/classsf_1_1View.htm#a8eaec46b7d332fe834f016d0187d4b4a)

| void sf::View::move | ( | float | *offsetX*, |
| --- | --- | --- | --- |
|  |  | float | *offsetY* |
|  | ) |  |  |

Move the view relatively to its current position.

Parameters

| offsetX | X coordinate of the move offset |
| --- | --- |
| offsetY | Y coordinate of the move offset |

See Also[setCenter](http://docs.google.com/classsf_1_1View.htm#aa8e3fedb008306ff9811163545fb75f2), [rotate](http://docs.google.com/classsf_1_1View.htm#a5fd3901aae1845586ca40add94faa378), [zoom](http://docs.google.com/classsf_1_1View.htm#a4a72a360a5792fbe4e99cd6feaf7726e)

| void sf::View::move | ( | const [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm) & | *offset* | ) |  |
| --- | --- | --- | --- | --- | --- |

Move the view relatively to its current position.

Parameters

| offset | Move offset |
| --- | --- |

See Also[setCenter](http://docs.google.com/classsf_1_1View.htm#aa8e3fedb008306ff9811163545fb75f2), [rotate](http://docs.google.com/classsf_1_1View.htm#a5fd3901aae1845586ca40add94faa378), [zoom](http://docs.google.com/classsf_1_1View.htm#a4a72a360a5792fbe4e99cd6feaf7726e)

| void sf::View::reset | ( | const [FloatRect](http://docs.google.com/classsf_1_1Rect.htm) & | *rectangle* | ) |  |
| --- | --- | --- | --- | --- | --- |

Reset the view to the given rectangle.

Note that this function resets the rotation angle to 0.

Parameters

| rectangle | Rectangle defining the zone to display |
| --- | --- |

See Also[setCenter](http://docs.google.com/classsf_1_1View.htm#aa8e3fedb008306ff9811163545fb75f2), [setSize](http://docs.google.com/classsf_1_1View.htm#a9525b73fe9fbaceb9568faf56b399dab), [setRotation](http://docs.google.com/classsf_1_1View.htm#a24d0503c9c51f5ef5918612786d325c1)

| void sf::View::rotate | ( | float | *angle* | ) |  |
| --- | --- | --- | --- | --- | --- |

Rotate the view relatively to its current orientation.

Parameters

| angle | Angle to rotate, in degrees |
| --- | --- |

See Also[setRotation](http://docs.google.com/classsf_1_1View.htm#a24d0503c9c51f5ef5918612786d325c1), [move](http://docs.google.com/classsf_1_1View.htm#a0c82144b837caf812f7cb25a43d80c41), [zoom](http://docs.google.com/classsf_1_1View.htm#a4a72a360a5792fbe4e99cd6feaf7726e)

| void sf::View::setCenter | ( | float | *x*, |
| --- | --- | --- | --- |
|  |  | float | *y* |
|  | ) |  |  |

Set the center of the view.

Parameters

| x | X coordinate of the new center |
| --- | --- |
| y | Y coordinate of the new center |

See Also[setSize](http://docs.google.com/classsf_1_1View.htm#a9525b73fe9fbaceb9568faf56b399dab), [getCenter](http://docs.google.com/classsf_1_1View.htm#adae81dede405b91bb3c487e28f536fe9)

| void sf::View::setCenter | ( | const [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm) & | *center* | ) |  |
| --- | --- | --- | --- | --- | --- |

Set the center of the view.

Parameters

| center | New center |
| --- | --- |

See Also[setSize](http://docs.google.com/classsf_1_1View.htm#a9525b73fe9fbaceb9568faf56b399dab), [getCenter](http://docs.google.com/classsf_1_1View.htm#adae81dede405b91bb3c487e28f536fe9)

| void sf::View::setRotation | ( | float | *angle* | ) |  |
| --- | --- | --- | --- | --- | --- |

Set the orientation of the view.

The default rotation of a view is 0 degree.

Parameters

| angle | New angle, in degrees |
| --- | --- |

See Also[getRotation](http://docs.google.com/classsf_1_1View.htm#a8ad320469a27f96f6f49de1c14f0978d)

| void sf::View::setSize | ( | float | *width*, |
| --- | --- | --- | --- |
|  |  | float | *height* |
|  | ) |  |  |

Set the size of the view.

Parameters

| width | New width of the view |
| --- | --- |
| height | New height of the view |

See Also[setCenter](http://docs.google.com/classsf_1_1View.htm#aa8e3fedb008306ff9811163545fb75f2), [getCenter](http://docs.google.com/classsf_1_1View.htm#adae81dede405b91bb3c487e28f536fe9)

| void sf::View::setSize | ( | const [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm) & | *size* | ) |  |
| --- | --- | --- | --- | --- | --- |

Set the size of the view.

Parameters

| size | New size |
| --- | --- |

See Also[setCenter](http://docs.google.com/classsf_1_1View.htm#aa8e3fedb008306ff9811163545fb75f2), [getCenter](http://docs.google.com/classsf_1_1View.htm#adae81dede405b91bb3c487e28f536fe9)

| void sf::View::setViewport | ( | const [FloatRect](http://docs.google.com/classsf_1_1Rect.htm) & | *viewport* | ) |  |
| --- | --- | --- | --- | --- | --- |

Set the target viewport.

The viewport is the rectangle into which the contents of the view are displayed, expressed as a factor (between 0 and 1) of the size of the [RenderTarget](http://docs.google.com/classsf_1_1RenderTarget.htm) to which the view is applied. For example, a view which takes the left side of the target would be defined with [View.setViewport](http://docs.google.com/classsf_1_1View.htm#a8eaec46b7d332fe834f016d0187d4b4a)(sf::FloatRect(0, 0, 0.5, 1)). By default, a view has a viewport which covers the entire target.

Parameters

| viewport | New viewport rectangle |
| --- | --- |

See Also[getViewport](http://docs.google.com/classsf_1_1View.htm#af1db1e4f21d104a9691fd38be9165758)

| void sf::View::zoom | ( | float | *factor* | ) |  |
| --- | --- | --- | --- | --- | --- |

Resize the view rectangle relatively to its current size.

Resizing the view simulates a zoom, as the zone displayed on screen grows or shrinks. *factor* is a multiplier:

* 1 keeps the size unchanged
* > 1 makes the view bigger (objects appear smaller)
* < 1 makes the view smaller (objects appear bigger)

Parameters

| factor | Zoom factor to apply |
| --- | --- |

See Also[setSize](http://docs.google.com/classsf_1_1View.htm#a9525b73fe9fbaceb9568faf56b399dab), [move](http://docs.google.com/classsf_1_1View.htm#a0c82144b837caf812f7cb25a43d80c41), [rotate](http://docs.google.com/classsf_1_1View.htm#a5fd3901aae1845586ca40add94faa378)

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

* [View.hpp](http://docs.google.com/View_8hpp_source.htm)

Copyright � Laurent Gomila  ::  Documentation generated by [doxygen](http://www.doxygen.org/)  ::