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)
* [File List](http://docs.google.com/files.htm)
* [include](http://docs.google.com/dir_f3190241575fd2bd132a392ae6942f4a.htm)
* [SFML](http://docs.google.com/dir_692f376662c82a26cfe4cfa3aceebe24.htm)
* [Network](http://docs.google.com/dir_b9ac88db2949395b3130dd4ffb1be4e1.htm)

SocketSelector.hpp

1

2 //

3 // SFML - Simple and Fast Multimedia Library

4 // Copyright (C) 2007-2013 Laurent Gomila (laurent.gom@gmail.com)

5 //

6 // This software is provided 'as-is', without any express or implied warranty.

7 // In no event will the authors be held liable for any damages arising from the use of this software.

8 //

9 // Permission is granted to anyone to use this software for any purpose,

10 // including commercial applications, and to alter it and redistribute it freely,

11 // subject to the following restrictions:

12 //

13 // 1. The origin of this software must not be misrepresented;

14 // you must not claim that you wrote the original software.

15 // If you use this software in a product, an acknowledgment

16 // in the product documentation would be appreciated but is not required.

17 //

18 // 2. Altered source versions must be plainly marked as such,

19 // and must not be misrepresented as being the original software.

20 //

21 // 3. This notice may not be removed or altered from any source distribution.

22 //

24

25 #ifndef SFML\_SOCKETSELECTOR\_HPP

26 #define SFML\_SOCKETSELECTOR\_HPP

27

29 // Headers

31 #include <SFML/Network/Export.hpp>

32 #include <SFML/System/Time.hpp>

33

34

35 namespace sf

36 {

37 class Socket;

38

[43](http://docs.google.com/classsf_1_1SocketSelector.htm) class SFML\_NETWORK\_API [SocketSelector](http://docs.google.com/classsf_1_1SocketSelector.htm)

44 {

45 public :

46

51  [SocketSelector](http://docs.google.com/classsf_1_1SocketSelector.htm)();

52

59  [SocketSelector](http://docs.google.com/classsf_1_1SocketSelector.htm)(const [SocketSelector](http://docs.google.com/classsf_1_1SocketSelector.htm)& copy);

60

65  ~[SocketSelector](http://docs.google.com/classsf_1_1SocketSelector.htm)();

66

80  void add([Socket](http://docs.google.com/classsf_1_1Socket.htm)& socket);

81

93  void remove([Socket](http://docs.google.com/classsf_1_1Socket.htm)& socket);

94

105  void clear();

106

123  bool wait([Time](http://docs.google.com/classsf_1_1Time.htm) timeout = [Time::Zero](http://docs.google.com/classsf_1_1Time.htm#a8db127b632fa8da21550e7282af11fa0));

124

142  bool isReady([Socket](http://docs.google.com/classsf_1_1Socket.htm)& socket) const;

143

152  [SocketSelector](http://docs.google.com/classsf_1_1SocketSelector.htm)& operator =(const [SocketSelector](http://docs.google.com/classsf_1_1SocketSelector.htm)& right);

153

154 private :

155

156  struct SocketSelectorImpl;

157

159  // Member data

161  SocketSelectorImpl\* m\_impl;

162 };

163

164 } // namespace sf

165

166

167 #endif // SFML\_SOCKETSELECTOR\_HPP

168

169

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