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)
* [Graphics](http://docs.google.com/dir_aaa96c3797a59111c2945d0d638ce5cf.htm)

CircleShape.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\_CIRCLESHAPE\_HPP

26 #define SFML\_CIRCLESHAPE\_HPP

27

29 // Headers

31 #include <SFML/Graphics/Export.hpp>

32 #include <SFML/Graphics/Shape.hpp>

33

34

35 namespace sf

36 {

[41](http://docs.google.com/classsf_1_1CircleShape.htm) class SFML\_GRAPHICS\_API [CircleShape](http://docs.google.com/classsf_1_1CircleShape.htm) : public [Shape](http://docs.google.com/classsf_1_1Shape.htm)

42 {

43 public :

44

52  explicit [CircleShape](http://docs.google.com/classsf_1_1CircleShape.htm)(float radius = 0, unsigned int pointCount = 30);

53

62  void setRadius(float radius);

63

72  float getRadius() const;

73

82  void setPointCount(unsigned int count);

83

92  virtual unsigned int getPointCount() const;

93

104  virtual [Vector2f](http://docs.google.com/classsf_1_1Vector2.htm) getPoint(unsigned int index) const;

105

106 private :

107

109  // Member data

111  float m\_radius;

112  unsigned int m\_pointCount;

113 };

114

115 } // namespace sf

116

117

118 #endif // SFML\_CIRCLESHAPE\_HPP

119

120

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