

libdx_f_2D

0.1

Generated by Doxygen 1.8.3.1

Mon Apr 29 2013 11:17:08

Contents

1	Main Page	1
2	Hierarchical Index	3
2.1	Class Hierarchy	3
3	Class Index	5
3.1	Class List	5
4	File Index	7
4.1	File List	7
5	Class Documentation	9
5.1	base Class Reference	9
5.1.1	Detailed Description	10
5.1.2	Member Function Documentation	10
5.1.2.1	read_Footer	10
5.1.2.2	read_Header	10
5.1.2.3	write_Hatch_End	10
5.1.2.4	write_Hatch_Start	10
5.1.3	Member Data Documentation	11
5.1.3.1	color	11
5.1.3.2	dxf_filename	11
5.1.3.3	edge_type	11
5.1.3.4	edges	11
5.1.3.5	flag	11
5.1.3.6	layer	11
5.1.3.7	name	11
5.1.3.8	pattern_name	11
5.1.3.9	radius	11
5.1.3.10	readFile	12
5.1.3.11	readwrite	12
5.1.3.12	writeFile	12
5.1.3.13	xEnd	12

5.1.3.14	xMid	12
5.1.3.15	xStart	12
5.1.3.16	yEnd	12
5.1.3.17	yMid	12
5.1.3.18	yStart	12
5.1.3.19	zEnd	13
5.1.3.20	zStart	13
5.2	circle Class Reference	13
5.2.1	Detailed Description	13
5.2.2	Constructor & Destructor Documentation	13
5.2.2.1	circle	14
5.2.2.2	circle	14
5.2.2.3	circle	14
5.3	dxf Class Reference	14
5.3.1	Detailed Description	15
5.3.2	Constructor & Destructor Documentation	15
5.3.2.1	dxf	15
5.3.3	Member Function Documentation	15
5.3.3.1	save	15
5.3.3.2	write_Circle	15
5.3.3.3	write_Circle	16
5.3.3.4	write_Circle	16
5.3.3.5	write_Hatch_Circle	16
5.3.3.6	write_Hatch_Rect	16
5.3.3.7	write_Line	16
5.3.3.8	write_Rect	16
5.3.3.9	write_Rect	16
5.3.3.10	write_Rect	17
5.4	line Class Reference	17
5.4.1	Detailed Description	17
5.4.2	Constructor & Destructor Documentation	17
5.4.2.1	line	17
5.5	point Class Reference	18
5.5.1	Detailed Description	18
5.5.2	Constructor & Destructor Documentation	18
5.5.2.1	point	18
5.5.3	Member Data Documentation	18
5.5.3.1	xCo	18
5.5.3.2	yCo	18
5.5.3.3	zCo	19

5.6	rectangle Class Reference	19
5.6.1	Detailed Description	19
5.6.2	Constructor & Destructor Documentation	19
5.6.2.1	rectangle	19
5.6.2.2	rectangle	20
6	File Documentation	21
6.1	include/dxf_2D.h File Reference	21
6.1.1	Detailed Description	21
6.2	src/dxf_base.cpp File Reference	22
6.2.1	Detailed Description	22
6.3	src/dxf_circle.cpp File Reference	22
6.3.1	Detailed Description	22
6.4	src/dxf_dxf.cpp File Reference	23
6.4.1	Detailed Description	23
6.5	src/dxf_line.cpp File Reference	23
6.5.1	Detailed Description	23
6.6	src/dxf_point.cpp File Reference	24
6.6.1	Detailed Description	24
6.7	src/dxf_rect.cpp File Reference	24
6.7.1	Detailed Description	25
	Index	25

Chapter 1

Main Page

libdxf_2D.so Library in C++

libdxf_2D.so is a dynamic shared library made in C++. This library is used to generate DXF file that can be opened in LibreCAD 1.0.2. It can generate DXF file using line, rectangle and circle entities. This library also provides hatching feature. You can fill either with solid fill or pattern fill.

REQUIREMENTS:

1) GNU G++ Compiler

Run following command in terminal to install

```
$ sudo apt-get install g++
```

INSTALLATION:

1) Open the terminal and type

```
$ git clone git://github.com/Akaur/testing.git
```

2) Go to this directory

```
$ cd testing
```

3) Run make

```
$ make
```


Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

base	9
circle	13
dxf	14
line	17
rectangle	19
point	18

Chapter 3

Class Index

3.1 Class List

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

base	Base class defines functions for DXF header, footer and hatching section	9
circle	Circle entity class	13
dxf	Draw entities in DXF file	14
line	Line entity class	17
point	Point entity class	18
rectangle	Rectangle entity class	19

Chapter 4

File Index

4.1 File List

Here is a list of all documented files with brief descriptions:

include/dxf_2D.h	21
src/dxf_base.cpp This file defines base class. base class is used to write header, footer and common hatching part to DXF file	22
src/dxf_circle.cpp This file defines circle class. circle class is used to create circle entity with or without hatching .	22
src/dxf_dxf.cpp This file defines dxf class. dxf class is used to draw multiple entites in a DXF file	23
src/dxf_line.cpp This file is defining line class. line class is used to create line entity	23
src/dxf_point.cpp This file defines point class. point class is used to initialize xCo, yCo and zCo coordinates of point	24
src/dxf_rect.cpp This file defines rectangle class. rectangle class is used to create rectangle with or without hatching	24
test/example.cpp	??

Chapter 5

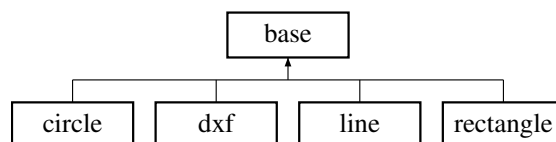
Class Documentation

5.1 base Class Reference

Base class defines functions for DXF header, footer and hatching section.

```
#include <dxfg_2D.h>
```

Inheritance diagram for base:



Public Member Functions

- void [read_Header](#) ()
Reads DXF header.
- void [read_Footer](#) ()
Reads DXF EOF section.
- void [write_Hatch_Start](#) (int, string, int)
Writes Hatch entity start part.
- void [write_Hatch_End](#) ()
Writes hatch entity end part.

Public Attributes

- ofstream [writeFile](#)

Protected Attributes

- double [xStart](#)
- double [yStart](#)
- double [zStart](#)
- double [xEnd](#)
- double [yEnd](#)
- double [zEnd](#)
- double [xMid](#)

- double `yMid`
- double `radius`
- ifstream `readFile`
- int `flag`
- int `edges`
- int `edge_type`
- int `color`
- string `dxf_filename`
- string `readwrite`
- string `pattern_name`
- string `name`
- string `layer`

5.1.1 Detailed Description

Base class defines functions for DXF header, footer and hatching section.

Definition at line 35 of file `dxf_2D.h`.

5.1.2 Member Function Documentation

5.1.2.1 void base::read_Footer ()

Reads DXF EOF section.

Opens "dxf_footer.txt" for reading and writes at the end of output DXF file.

Definition at line 56 of file `dxf_base.cpp`.

5.1.2.2 void base::read_Header ()

Reads DXF header.

Opens "dxf_header.txt" file for reading and writes at the start of DXF file.

Definition at line 30 of file `dxf_base.cpp`.

5.1.2.3 void base::write_Hatch_End ()

Writes hatch entity end part.

Writes the common end part of hatch entity to the entity section of output DXF file.

Definition at line 106 of file `dxf_base.cpp`.

5.1.2.4 void base::write_Hatch_Start (int *flag*, string *pattern_name*, int *color*)

Writes Hatch entity start part.

Writes the common start part of hatch entity to the the entity section of DXF output file.

Definition at line 82 of file `dxf_base.cpp`.

5.1.3 Member Data Documentation

5.1.3.1 `int base::color` [protected]

color-code[1-256]

Definition at line 51 of file dxf_2D.h.

5.1.3.2 `string base::dxf_filename` [protected]

output dxf filename

Definition at line 57 of file dxf_2D.h.

5.1.3.3 `int base::edge_type` [protected]

boundary edge type

Definition at line 51 of file dxf_2D.h.

5.1.3.4 `int base::edges` [protected]

edges of boundary path

Definition at line 51 of file dxf_2D.h.

5.1.3.5 `int base::flag` [protected]

flag for hatching

Definition at line 51 of file dxf_2D.h.

5.1.3.6 `string base::layer` [protected]

layer[0-5]

Definition at line 57 of file dxf_2D.h.

5.1.3.7 `string base::name` [protected]

input filename for reading

Definition at line 57 of file dxf_2D.h.

5.1.3.8 `string base::pattern_name` [protected]

pattern name for pattern hatching

Definition at line 57 of file dxf_2D.h.

5.1.3.9 `double base::radius` [protected]

radius of circle

Definition at line 39 of file dxf_2D.h.

5.1.3.10 ifstream base::readFile [protected]

object for reading files

Definition at line 49 of file dxf_2D.h.

5.1.3.11 string base::readwrite [protected]

reads from input file and writes to output file

Definition at line 57 of file dxf_2D.h.

5.1.3.12 ofstream base::writeFile

object for writing files

Definition at line 66 of file dxf_2D.h.

5.1.3.13 double base::xEnd [protected]

x-coordinate of ending point(x2, y2, z2)

Definition at line 39 of file dxf_2D.h.

5.1.3.14 double base::xMid [protected]

x-coordinate of center(x, y) for circle

Definition at line 39 of file dxf_2D.h.

5.1.3.15 double base::xStart [protected]

x-coordinate of starting point(x1, y1, z1)

Definition at line 39 of file dxf_2D.h.

5.1.3.16 double base::yEnd [protected]

y-coordinate of ending point(x2, y2, z2)

Definition at line 39 of file dxf_2D.h.

5.1.3.17 double base::yMid [protected]

y-coordinate of center(x, y) for circle

Definition at line 39 of file dxf_2D.h.

5.1.3.18 double base::yStart [protected]

y-coordinate of starting point(x1, y1, z1)

Definition at line 39 of file dxf_2D.h.

5.1.3.19 `double base::zEnd` `[protected]`

z-coordinate of ending point(x2, y2, z2)

Definition at line 39 of file dxf_2D.h.

5.1.3.20 `double base::zStart` `[protected]`

z-coordinate of starting point(x1, y1, z1)

Definition at line 39 of file dxf_2D.h.

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

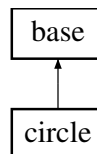
- [include/dxf_2D.h](#)
- [src/dxf_base.cpp](#)

5.2 circle Class Reference

Circle entity class.

```
#include <dxf_2D.h>
```

Inheritance diagram for circle:



Public Member Functions

- [circle](#) ()
Default constructor.
- [circle](#) (point &, double, string, dxf &)
Parameterized constructor.
- [circle](#) (point &, double, string, dxf &, int hflag, int hcolor=256)
Parameterized constructor for solid fill.
- [circle](#) (point &, double, string, dxf &, int hflag, string p_name, int hcolor=256)
Parameterized constructor for pattern fill.

Additional Inherited Members

5.2.1 Detailed Description

Circle entity class.

Definition at line 192 of file dxf_2D.h.

5.2.2 Constructor & Destructor Documentation

5.2.2.1 circle::circle (point & pt1, double r, string dlayer, dxf & d)

Parameterized constructor.

Initializes starting point and radius of circle, specify the layer and calls write_Circle(xStart, yStart, zStart, radius, layer) function of dxf class for writing circle. entity to DXF file.

Definition at line 45 of file dxf_circle.cpp.

5.2.2.2 circle::circle (point & pt1, double r, string dlayer, dxf & d, int hflag, int hcolor = 256)

Parameterized constructor for solid fill.

Initialize starting point and radius of circle, specify the layer, flag = 1 for solid fill and calls write_Circle(xStart, yStart, zStart, radius, layer, flag, color) function of dxf class for writing circle (solid fill) entity to DXF file.

Definition at line 74 of file dxf_circle.cpp.

5.2.2.3 circle::circle (point & pt1, double r, string dlayer, dxf & d, int hflag, string p_name, int hcolor = 256)

Parameterized constructor for pattern fill.

Initialize starting point and radius of circle, specify the layer, flag = 0 for pattern fill and calls write_Circle(xStart, yStart, zStart, radius, layer, flag, pattern_name, color) function of dxf class for writing circle (pattern fill) entity to DXF file.

Definition at line 107 of file dxf_circle.cpp.

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

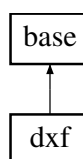
- [include/dxf_2D.h](#)
- [src/dxf_circle.cpp](#)

5.3 dxf Class Reference

Draw entities in DXF file.

```
#include <dxf_2D.h>
```

Inheritance diagram for dxf:



Public Member Functions

- [dxf \(\)](#)
Default constructor.
- [dxf \(string\)](#)
Parameterised constructor.
- void [save \(\)](#)
Save DXF file.
- void [write_Line](#) (double, double, double, double, double, double, double, string)
Line entity.

- void `write_Circle` (double, double, double, double, string)
Circle entity.
- void `write_Circle` (double, double, double, double, string, int, int)
Circle (solid fill) entity.
- void `write_Circle` (double, double, double, double, string, int, string, int)
Circle (pattern fill) entity.
- void `write_Hatch_Circle` (double, double, double, double)
Hatch circle entity.
- void `write_Rect` (double, double, double, double, double, double, string)
Rectangle entity.
- void `write_Rect` (double, double, double, double, double, double, string, int, int)
Rectangle (solid fill) entity.
- void `write_Rect` (double, double, double, double, double, double, string, int, string, int)
Rectangle (pattern fill) entity.
- void `write_Hatch_Rect` (double, double, double, double, double, double)
Hatch rectangle entity.

Additional Inherited Members

5.3.1 Detailed Description

Draw entities in DXF file.

Definition at line 114 of file `dxf_2D.h`.

5.3.2 Constructor & Destructor Documentation

5.3.2.1 `dxflib::dxf (string filename)`

Parameterised constructor.

Constructor opens the filename for writing and writes DXF header to it.

Definition at line 43 of file `dxf_dxf.cpp`.

5.3.3 Member Function Documentation

5.3.3.1 `void dxflib::save ()`

Save DXF file.

Constructor closes DXF file after writing DXF footer to it.

Definition at line 60 of file `dxf_dxf.cpp`.

5.3.3.2 `void dxflib::write_Circle (double xStart, double yStart, double zStart, double radius, string layer)`

Circle entity.

Writes circle entity to DXF file.

Definition at line 109 of file `dxf_dxf.cpp`.

5.3.3.3 void dxf::write_Circle (double xStart, double yStart, double zStart, double radius, string layer, int flag, int color)

Circle (solid fill) entity.

Writes circle (solid fill) entity to DXF file by calling write_Hatch_Start(flag, pattern_name, color), write_Hatch_Circle(xStart, yStart, zStart, radius), [write_Hatch_End\(\)](#) and write_Circle(xStart, yStart, zStart, radius, layer).

Definition at line 140 of file dxf_dxf.cpp.

5.3.3.4 void dxf::write_Circle (double xStart, double yStart, double zStart, double radius, string layer, int flag, string pattern_name, int color)

Circle (pattern fill) entity.

Writes circle (pattern fill) entity to DXF file by calling write_Hatch_Start(flag, pattern_name, color), write_Hatch_Circle(xStart, yStart, zStart, radius), [write_Hatch_End\(\)](#) and write_Circle(xStart, yStart, zStart, radius, layer).

Definition at line 169 of file dxf_dxf.cpp.

5.3.3.5 void dxf::write_Hatch_Circle (double xStart, double yStart, double zStart, double radius)

Hatch circle entity.

Writes hatch circle entity to DXF file.

Definition at line 192 of file dxf_dxf.cpp.

5.3.3.6 void dxf::write_Hatch_Rect (double xStart, double yStart, double zStart, double xEnd, double yEnd, double zEnd)

Hatch rectangle entity.

Writes hatch rectangle entity to DXF file.

Definition at line 303 of file dxf_dxf.cpp.

5.3.3.7 void dxf::write_Line (double xStart, double yStart, double zStart, double xEnd, double yEnd, double zEnd, string layer)

Line entity.

Writes line entity to DXF file.

Definition at line 78 of file dxf_dxf.cpp.

5.3.3.8 void dxf::write_Rect (double xStart, double yStart, double zStart, double xEnd, double yEnd, double zEnd, string layer)

Rectangle entity.

Writes rectangle entity to DXF file.

Definition at line 213 of file dxf_dxf.cpp.

5.3.3.9 void dxf::write_Rect (double xStart, double yStart, double zStart, double xEnd, double yEnd, double zEnd, string layer, int flag, int color)

Rectangle (solid fill) entity.

Writes rectangle (solid fill) entity to DXF file by calling write_Hatch_Start(flag, pattern_name, color), write_Hatch_Rect(xStart, yStart, zStart, xEnd, yEnd, zEnd), [write_Hatch_End\(\)](#) and write_Rect(xStart, yStart, zStart, xEnd, yEnd, zEnd, layer).

Definition at line 249 of file dxf_dxf.cpp.

```
5.3.3.10 void dxf::write_Rect ( double xStart, double yStart, double zStart, double xEnd, double yEnd, double zEnd, string
layer, int flag, string pattern_name, int color )
```

Rectangle (pattern fill) entity.

Writes rectangle (pattern fill) entity to DXF file by calling write_Hatch_Start(flag, pattern_name, color), write_Hatch_Rect(xStart, yStart, zStart, xEnd, yEnd, zEnd), [write_Hatch_End\(\)](#) and write_Rect(xStart, yStart, zStart, xEnd, yEnd, zEnd, layer).

Definition at line 279 of file dxf_dxf.cpp.

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

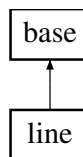
- [include/dxf_2D.h](#)
- [src/dxf_dxf.cpp](#)

5.4 line Class Reference

Line entity class.

```
#include <dxf_2D.h>
```

Inheritance diagram for line:



Public Member Functions

- [line](#) ()
Default constructor.
- [line](#) ([point](#) &, [point](#) &, string, [dxf](#) &)
Parameterized constructor.

Additional Inherited Members

5.4.1 Detailed Description

Line entity class.

Definition at line 172 of file dxf_2D.h.

5.4.2 Constructor & Destructor Documentation

```
5.4.2.1 line::line ( point & pt1, point & pt2, string dlayer, dxf & d )
```

Parameterized constructor.

Initializes starting and ending points of line, specify the layer and calls write_Line(xStart, yStart, zStart, xEnd, yEnd, zEnd, layer) function of dxf class for writing line entity to DXF file.

Definition at line 43 of file dxf_line.cpp.

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

- [include/dxf_2D.h](#)
- [src/dxf_line.cpp](#)

5.5 point Class Reference

Point entity class.

```
#include <dxf_2D.h>
```

Public Member Functions

- [point](#) ()
Default constructor.
- [point](#) (double, double y, double z=0)
Parameterised constructor. Default value of z is set to 0.

Public Attributes

- double [xCo](#)
- double [yCo](#)
- double [zCo](#)

5.5.1 Detailed Description

Point entity class.

Definition at line 90 of file dxf_2D.h.

5.5.2 Constructor & Destructor Documentation

5.5.2.1 point::point ()

Default constructor.

Initializes xCo, yCo, zCo coordintes of point.

Definition at line 29 of file dxf_point.cpp.

5.5.3 Member Data Documentation

5.5.3.1 double point::xCo

x-coordinate of point(x, y, z)

Definition at line 94 of file dxf_2D.h.

5.5.3.2 double point::yCo

y-coordinate of point(x, y, z)

Definition at line 94 of file dxf_2D.h.

5.5.3.3 double point::zCo

z-coordinate of point(x, y, z)

Definition at line 94 of file dxf_2D.h.

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

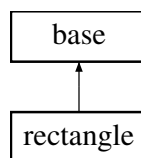
- [include/dxf_2D.h](#)
- [src/dxf_point.cpp](#)

5.6 rectangle Class Reference

Rectangle entity class.

```
#include <dxf_2D.h>
```

Inheritance diagram for rectangle:



Public Member Functions

- [rectangle](#) ()
Default constructor.
- [rectangle](#) (point &, point &, string, dxf &)
Parameterized constructor.
- [rectangle](#) (point &, point &, string, dxf &, int hflag, int hcolor=256)
Parameterized constructor for solid fill.
- [rectangle](#) (point &, point &, string, dxf &, int hflag, string p_name, int hcolor=256)
Parameterized constructor.

Additional Inherited Members

5.6.1 Detailed Description

Rectangle entity class.

Definition at line 219 of file dxf_2D.h.

5.6.2 Constructor & Destructor Documentation

5.6.2.1 rectangle::rectangle (point &pt1, point &pt2, string dlayer, dxf & d)

Parameterized constructor.

Initializes starting and ending points of rectangle, specify the layer and calls write_Rect(xStart, yStart, zStart, xEnd, yEnd, zEnd, layer) function of dxf class for writing rectangle entity to DXF file.

Definition at line 44 of file dxf_rect.cpp.

5.6.2.2 `rectangle::rectangle (point & pt1, point & pt2, string dlayer, dxf & d, int hflag, int hcolor = 256)`

Parameterized constructor for solid fill.

Initialize starting and ending points of rectangle, specify the layer, flag = 1 for solid fill and calls `write_Rect(xStart, yStart, zStart, xEnd, yEnd, zEnd, layer, flag, color)` function of `dxf` class for writing rectangle (solid fill) entity to DXF file.

Definition at line 76 of file `dxf_rect.cpp`.

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

- [include/dxf_2D.h](#)
- [src/dxf_rect.cpp](#)

Chapter 6

File Documentation

6.1 include/dxf_2D.h File Reference

```
#include <cmath>
#include <cstring>
#include <fstream>
#include <iostream>
```

Classes

- class [base](#)
Base class defines functions for DXF header, footer and hatching section.
- class [point](#)
Point entity class.
- class [dxf](#)
Draw entities in DXF file.
- class [line](#)
Line entity class.
- class [circle](#)
Circle entity class.
- class [rectangle](#)
Rectangle entity class.

6.1.1 Detailed Description

This is a header file that declare classes for creating 2D drawings.

Version

0.1

Date

03/19/2013 01:34:03 PM Compiler gcc

Author

Avneet Kaur, kauravneet958@gmail.com License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [dx_f_2D.h](#).

6.2 src/dxf_base.cpp File Reference

This file defines base class. base class is used to write header, footer and common hatching part to DXF file.

```
#include "../include/dxf_2D.h"
```

6.2.1 Detailed Description

This file defines base class. base class is used to write header, footer and common hatching part to DXF file.

Version

0.1

Date

03/19/2013 09:30:23 PM Compiler gcc

Author

Avneet Kaur, kauravneet958@gmail.com License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [dxf_base.cpp](#).

6.3 src/dxf_circle.cpp File Reference

This file defines circle class. circle class is used to create circle entity with or without hatching.

```
#include "../include/dxf_2D.h"
```

6.3.1 Detailed Description

This file defines circle class. circle class is used to create circle entity with or without hatching.

Version

0.1

Date

03/19/2013 09:30:23 PM Compiler gcc

Author

Avneet Kaur, kauravneet958@gmail.com License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [dxf_circle.cpp](#).

6.4 src/dxf_dxf.cpp File Reference

This file defines dxf class. dxf class is used to draw multiple entites in a DXF file.

```
#include "../include/dxf_2D.h"
```

6.4.1 Detailed Description

This file defines dxf class. dxf class is used to draw multiple entites in a DXF file.

Version

0.1

Date

03/19/2013 09:10:35 PM Compiler gcc

Author

Avneet Kaur, kauravneet958@gmail.com License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [dxf_dxf.cpp](#).

6.5 src/dxf_line.cpp File Reference

This file is defining line class. line class is used to create line entity.

```
#include "../include/dxf_2D.h"
```

6.5.1 Detailed Description

This file is defining line class. line class is used to create line entity.

Version

0.1

Date

03/19/2013 09:10:35 PM Compiler gcc

Author

Avneet Kaur, kauravneet958@gmail.com License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [dxf_line.cpp](#).

6.6 src/dxf_point.cpp File Reference

This file defines point class. point class is used to initialize xCo, yCo and zCo coordinates of point.

```
#include "../include/dxf_2D.h"
```

6.6.1 Detailed Description

This file defines point class. point class is used to initialize xCo, yCo and zCo coordinates of point.

Version

0.1

Date

03/19/2013 09:30:23 PM Compiler gcc

Author

Avneet Kaur, kauravneet958@gmail.com License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [dxf_point.cpp](#).

6.7 src/dxf_rect.cpp File Reference

This file defines rectangle class. rectangle class is used to create rectangle with or without hatching.

```
#include "../include/dxf_2D.h"
```

6.7.1 Detailed Description

This file defines rectangle class. rectangle class is used to create rectangle with or without hatching.

Version

0.1

Date

03/19/2013 09:30:23 PM Compiler gcc

Author

Avneet Kaur, kauravneet958@gmail.com License GNU General Public License

Copyright

Copyright (c) 2013, GreatDevelopers <https://github.com/GreatDevelopers>

Definition in file [dxf_rect.cpp](#).

Index

base, [9](#)

- color, [11](#)
- dxf_filename, [11](#)
- edge_type, [11](#)
- edges, [11](#)
- flag, [11](#)
- layer, [11](#)
- name, [11](#)
- pattern_name, [11](#)
- radius, [11](#)
- read_Footer, [10](#)
- read_Header, [10](#)
- readFile, [11](#)
- readwrite, [12](#)
- write_Hatch_End, [10](#)
- write_Hatch_Start, [10](#)
- writeFile, [12](#)
- xEnd, [12](#)
- xMid, [12](#)
- xStart, [12](#)
- yEnd, [12](#)
- yMid, [12](#)
- yStart, [12](#)
- zEnd, [12](#)
- zStart, [13](#)

circle, [13](#)

- circle, [13](#), [14](#)

color

- base, [11](#)

dxf, [14](#)

- dxf, [15](#)
- save, [15](#)
- write_Circle, [15](#), [16](#)
- write_Hatch_Circle, [16](#)
- write_Hatch_Rect, [16](#)
- write_Line, [16](#)
- write_Rect, [16](#), [17](#)

dxf_filename

- base, [11](#)

edge_type

- base, [11](#)

edges

- base, [11](#)

flag

- base, [11](#)

include/dxf_2D.h, [21](#)

layer

- base, [11](#)

line, [17](#)

- line, [17](#)

name

- base, [11](#)

pattern_name

- base, [11](#)

point, [18](#)

- point, [18](#)

- xCo, [18](#)

- yCo, [18](#)

- zCo, [18](#)

radius

- base, [11](#)

read_Footer

- base, [10](#)

read_Header

- base, [10](#)

readFile

- base, [11](#)

readwrite

- base, [12](#)

rectangle, [19](#)

- rectangle, [19](#)

save

- dxf, [15](#)

src/dxf_base.cpp, [22](#)

src/dxf_circle.cpp, [22](#)

src/dxf_dxf.cpp, [23](#)

src/dxf_line.cpp, [23](#)

src/dxf_point.cpp, [24](#)

src/dxf_rect.cpp, [24](#)

write_Circle

- dxf, [15](#), [16](#)

write_Hatch_Circle

- dxf, [16](#)

write_Hatch_End

- base, [10](#)

write_Hatch_Rect

- dxf, [16](#)

write_Hatch_Start

- base, [10](#)

write_Line

- dxf, [16](#)

write_Rect

- dx, [16](#), [17](#)
- writeFile
 - base, [12](#)
- xCo
 - point, [18](#)
- xEnd
 - base, [12](#)
- xMid
 - base, [12](#)
- xStart
 - base, [12](#)
- yCo
 - point, [18](#)
- yEnd
 - base, [12](#)
- yMid
 - base, [12](#)
- yStart
 - base, [12](#)
- zCo
 - point, [18](#)
- zEnd
 - base, [12](#)
- zStart
 - base, [13](#)