**本文博客链接:**[**http://blog.csdn.net/jdh99**](http://blog.csdn.net/jdh99)**,作者:jdh,转载请注明.**

**环境：**

主机:WIN8

开发环境:Qt5.4.0

**说明:**

调用热敏打印机，打印测试字符

**源码:**

**pro文件加载库文件**

**[html]** [view plaincopy](http://blog.csdn.net/jdh99/article/details/42585987)

1. #-------------------------------------------------
2. #
3. # Project created by QtCreator 2015-01-10T09:52:53
4. #
5. #-------------------------------------------------
7. QT       += core gui
8. qtHaveModule(printsupport): QT += printsupport
10. greaterThan(QT\_MAJOR\_VERSION, 4): QT += widgets
12. TARGET = test\_printer
13. TEMPLATE = app

16. SOURCES += main.cpp\
17. mainwindow.cpp
19. HEADERS  += mainwindow.h
21. FORMS    += mainwindow.ui

**mainwindows.h**

**[cpp]** [view plaincopy](http://blog.csdn.net/jdh99/article/details/42585987)[在CODE上查看代码片](https://code.csdn.net/snippets/576771)

1. #ifndef MAINWINDOW\_H
2. #define MAINWINDOW\_H
4. #include <QMainWindow>
6. #include <QPrintDialog>
7. #include <QPrinter>
8. #include <QPrintPreviewDialog>
9. #include <QFileDialog>
10. #include <QPageSetupDialog>
12. **namespace** Ui {
13. **class** MainWindow;
14. }
16. **class** MainWindow : **public** QMainWindow
17. {
18. Q\_OBJECT
20. **public**:
21. **explicit** MainWindow(QWidget \*parent = 0);
22. ~MainWindow();
24. **private** slots:
25. **void** on\_pushButton\_clicked();
27. **void** doPrint();
28. **void** doPrintPreview();
29. **void** printPreview(QPrinter \*printer);
30. **void** createPdf();
31. **void** setUpPage();

34. **private**:
35. Ui::MainWindow \*ui;
36. };
38. #endif // MAINWINDOW\_H

**mainwindow.cpp**

**[cpp]** [view plaincopy](http://blog.csdn.net/jdh99/article/details/42585987)[在CODE上查看代码片](https://code.csdn.net/snippets/576771)

1. #include "mainwindow.h"
2. #include "ui\_mainwindow.h"
4. #include <QDebug>
5. #include <QTextDocument>
7. MainWindow::MainWindow(QWidget \*parent) :
8. QMainWindow(parent),
9. ui(**new** Ui::MainWindow)
10. {
11. ui->setupUi(**this**);
12. }
14. MainWindow::~MainWindow()
15. {
16. **delete** ui;
17. }
19. **void** MainWindow::on\_pushButton\_clicked()
20. {
21. QPrinter printer;
22. QTextDocument doc;
24. QFont font("宋体",2);
25. doc.setDefaultFont(font);
27. //    QFont font = doc.defaultFont();
28. //    font.setBold(true);
29. //    font.setPointSize(10);
30. //    doc.setDefaultFont(font);
32. QSizeF s = QSizeF(printer.logicalDpiX() \* (58 / 25.4),
33. printer.logicalDpiY() \* (297 / 25.4));
34. doc.setPageSize(s);
35. printer.setPageSizeMM(s);
36. printer.setOutputFormat(QPrinter::NativeFormat);
37. doc.setPlainText("I am jdh!\n打印测试");
38. doc.print(&printer);
39. }
41. **void** MainWindow::doPrint()
42. {
43. // 创建打印机对象
44. QPrinter printer;
45. // 创建打印对话框
46. QString printerName = printer.printerName();
47. **if**( printerName.size() == 0)
48. **return**;
49. QPrintDialog dlg(&printer, **this**);
50. //如果编辑器中有选中区域，则打印选中区域
51. **if** (ui->textEdit->textCursor().hasSelection())
52. dlg.addEnabledOption(QAbstractPrintDialog::PrintSelection);
53. // 如果在对话框中按下了打印按钮，则执行打印操作
54. **if** (dlg.exec() == QDialog::Accepted)
55. {
56. //ui->textEdit->print(&printer);
57. // print the existing document by absoult path
58. //printFile("D:/1.doc");
60. QTextDocument doc;
61. //doc.setHtml(htmlString);
62. QFont font = doc.defaultFont();
63. font.setBold(**true**);
64. font.setPointSize(10);
65. doc.setDefaultFont(font);
66. QSizeF s = QSizeF(printer.logicalDpiX() \* (58 / 25.4), printer.logicalDpiY() \* (297 / 25.4));
67. //doc.setPageSize(s);
68. printer.setPageSizeMM(s);
69. printer.setOutputFormat(QPrinter::NativeFormat);
71. doc.setHtml("店联");
72. doc.print(&printer);
73. }
74. }
75. // 打印预览
76. **void** MainWindow::doPrintPreview()
77. {
78. QPrinter printer;
80. // 创建打印预览对话框
81. QPrintPreviewDialog preview(&printer, **this**);
82. // 当要生成预览页面时，发射paintRequested()信号
83. connect(&preview, SIGNAL(paintRequested(QPrinter\*)),
84. **this**,SLOT(printPreview(QPrinter\*)));
85. preview.exec();
86. }
87. **void** MainWindow::printPreview(QPrinter \*printer)
88. {
89. //ui->textEdit->print(printer);
90. QTextDocument doc;
91. //doc.setHtml(htmlString);
92. //    QFont font = doc.defaultFont();
93. //    font.setBold(true);
94. //    font.setPointSize(font.pointSize() + 1);
95. //    doc.setDefaultFont(font);
96. QSizeF s = QSizeF(printer->logicalDpiX() \* (58 / 25.4), printer->logicalDpiY() \* (297 / 25.4));
97. //    doc.setPageSize(s);
98. printer->setPageSizeMM(s);
99. printer->setOutputFormat(QPrinter::NativeFormat);
101. doc.setHtml("店联");
102. doc.print(printer);
103. }
104. // 生成PDF文件
105. **void** MainWindow::createPdf()
106. {
107. QString fileName = QFileDialog::getSaveFileName(**this**, tr("导出PDF文件"), QString(), "\*.pdf");
108. **if** (!fileName.isEmpty()) {
109. // 如果文件后缀为空，则默认使用.pdf
110. **if** (QFileInfo(fileName).suffix().isEmpty())
111. fileName.append(".pdf");
112. QPrinter printer;
113. // 指定输出格式为pdf
114. printer.setOutputFormat(QPrinter::PdfFormat);
115. printer.setOutputFileName(fileName);
116. ui->textEdit->print(&printer);
117. }
118. }
119. // 页面设置
120. **void** MainWindow::setUpPage()
121. {
122. QPrinter printer;
123. QPageSetupDialog pageSetUpdlg(&printer, **this**);
124. **if** (pageSetUpdlg.exec() == QDialog::Accepted)
125. {
126. printer.setOrientation(QPrinter::Landscape);
127. }
128. **else**
129. {
130. printer.setOrientation(QPrinter::Portrait);
131. }
132. }