

Python ファイルを分割する

[Python][PyQt]



```
# -*- encoding:utf-8 -*-
import sys
import re
import os
from datetime import datetime
from PyQt4 import QtGui as gui
from PyQt4 import QtCore as core
import re
import codecs

class MainWindow(gui.QWidget):
    def __init__(self):
        super(MainWindow, self).__init__()
        self.initUi()

    def initUi(self):
        grid = gui.QGridLayout()

        form = gui.QFormLayout()
        self.txtSepFile = gui.QLineEdit()
        btnSepFile = gui.QPushButton(u'...')
        btnSepFile.setMaximumWidth(40)
        btnSepFile.clicked.connect(self.chooseDbFile)
        boxSepFile = gui.QHBoxLayout()
        boxSepFile.addWidget(self.txtSepFile)
        boxSepFile.addWidget(btnSepFile)
        form.addRow(u'対象ファイル', boxSepFile)

        self.txtLimit = gui.QLineEdit()
        self.txtLimit.setText(u'10000')
        self.txtLimit.setAlignment(core.Qt.AlignRight)
        self.txtLimit.setFixedWidth(100)
        form.addRow(u'行数', self.txtLimit)

        encodes = ('shift_jis', 'cp932', 'utf-8')

        self.cmbSrcEncode = gui.QComboBox(self)
        self.cmbSrcEncode.addItem(encodes)
        self.cmbSrcEncode.setFixedWidth(150)
        boxSrcEncode = gui.QHBoxLayout()
        boxSrcEncode.addWidget(self.cmbSrcEncode)
        form.addRow(u'対象エンコード', boxSrcEncode)

        boxCtrl = gui.QHBoxLayout()
        btnExec = gui.QPushButton(u'実行')
        btnExec.clicked.connect(self.doExec)
        boxCtrl.addWidget(btnExec)

        msgCtrl = gui.QVBoxLayout()
        msgCtrl.addWidget(gui.QLabel(u"テキストファイルを指定行数で分割します。"))

        grid.addLayout(msgCtrl, 0, 0)
        grid.addLayout(form, 1, 0)
        grid.addLayout(boxCtrl, 2, 0)
```

```

self.setLayout(grid)
self.resize(400,100)

self.setWindowTitle(u" テキストファイル分割 ")
self.show()

def doExec(self):
    in_file = unicode(self.txtSepFile.text())

    in_codec = unicode(self.cmbSrcEncode.currentText())

    s_limit = unicode(self.txtLimit.text())
    if not unicode.isnumeric(s_limit):
        gui.QMessageBox.warning(self,"Completed",u" 行数の指定が不正です。")
        return

    self.separateFile(in_file, in_codec, int(s_limit))

    gui.QMessageBox.information(self,"Completed",u" 作成完了しました。",gui.QMessageBox.Ok)

def chooseDbFile(self):
    dialog = gui.QFileDialog()
    dialog.setFileMode(gui.QFileDialog.ExistingFile)
    if dialog.exec_():
        fileNames = dialog.selectedFiles()
        for f in fileNames:
            self.txtSepFile.setText(f)
            return
    return self.txtSepFile.setText('')

def separateFile(self, in_file, in_codec, limit):
    extention = ""
    out_file = in_file
    try:
        in_sep_period = in_file.split(".")
        extention = in_sep_period[1]
        out_file = ".".join(in_sep_period[:-1])
    except IndexError:
        pass

    in_fd = None
    out_fd = None
    try:
        suffix = 0
        line = 0
        in_fd = codecs.open(in_file, 'r', in_codec)
        for l in in_fd:
            if line == 0:
                if out_fd:
                    out_fd.close()
                suffix = suffix + 1
                outf = out_file + '_%05d.%s' % (suffix, extention)
                out_fd = codecs.open(outf, 'w', in_codec)
                print "%s..." % (outf)

                out_fd.write(l)

                # print l
                line = line + 1
                if line >= limit:
                    line = 0
            except UnicodeDecodeError:
                gui.QMessageBox.warning(self,"Completed",u" エンコードエラー。エンコードの指定を確認して
ください。")
                if out_fd:
                    out_fd.close()
                if in_fd:
                    in_fd.close()
                pass

    def main():
        app = gui.QApplication(sys.argv)
        mainWin = MainWindow()
        sys.exit(app.exec_())

if __name__ == '__main__':
    main()

```