**代码：**

**main\_window.py:**

**# -\*- coding: utf-8 -\*-**

**# Form implementation generated from reading ui file 'C:\Users\HB\Desktop\python作业\hb.ui'**

**#**

**# Created by: PyQt5 UI code generator 5.9.2**

**#**

**# WARNING! All changes made in this file will be lost!**

**import sys**

**from PyQt5 import QtCore, QtGui, QtWidgets**

**from PyQt5.QtWidgets import QLCDNumber,QPushButton,QWidget,QDesktopWidget,QMessageBox,QFileDialog**

**from PyQt5.Qt import QPoint, QTimer, QTime, QRect**

**from PyQt5.QtCore import Qt,pyqtSignal,QThread**

**from PyQt5.QtGui import QFont**

**import sub\_window**

**class clock(QLCDNumber):**

**#关于其中显示当前时间的类**

**def \_\_init\_\_(self, Parent=None):**

**super().\_\_init\_\_(Parent)**

**self.\_\_InitData() #初始化类的数据**

**self.\_\_InitView() #初始化界面**

**def \_\_InitData(self):**

**self.\_\_ShowColon=True #时间中冒号是否显示**

**self.thread = QTimer(self) #新建一个定时器**

**#关联timeout信号和showTime函数，每当定时器过了指定时间间隔，就会调用showTime函数**

**self.thread.timeout.connect(self.\_\_showTime)**

**self.thread.start(1000) #设置定时间隔为1000ms即1s，并启动定时器**

**def \_\_InitView(self):**

**#初始化界面**

**self.setNumDigits(8) #允许显示8个字符【HH:MM:SS】**

**self.\_\_showTime() #初始化时间的显示**

**def \_\_showTime(self):**

**#更新时间的显示**

**time = QTime.currentTime() #获取当前时间**

**time\_text = time.toString(Qt.DefaultLocaleLongDate) #获取HH:MM:SS格式的时间，在中国获取后是这个格式，其他国家我不知道，如果有土豪愿意送我去外国旅行的话我就可以试一试**

**#冒号闪烁**

**if self.\_\_ShowColon == True:**

**self.\_\_ShowColon = False**

**else:**

**time\_text = time\_text.replace(':',' ')**

**self.\_\_ShowColon = True**

**self.display(time\_text) #显示时间**

**class assist\_clock(QTimer):**

**clock\_sign=pyqtSignal()**

**def \_\_init\_\_(self,parent=None):**

**super().\_\_init\_\_(parent)**

**def run(self):**

**self.clock\_sign.emit()**

**class CountDown(QLCDNumber):**

**#倒计时类**

**def \_\_init\_\_(self,parent=None):**

**super().\_\_init\_\_(parent)**

**self.count\_time=0**

**self.\_\_InitView() #初始化界面**

**self.\_\_timer = QTimer(self) #新建一个定时器**

**self.\_\_timer.timeout.connect(self.\_\_showTime)**

**self.thread1=CountDownSign1()#创建新线程1**

**self.thread2=CountDownSign2()#创建新线程1**

**self.thread1.sign1.connect(self.msg1)**

**self.thread2.sign2.connect(self.msg2)**

**def CountStart(self,time):**

**#开始计时函数**

**self.count\_time=time**

**self.\_\_timer.start(1000) #设置定时间隔为1000ms即1s，并启动定时器**

**def \_\_InitView(self):**

**#初始化界面**

**self.setNumDigits(6) #允许显示6个字符**

**self.\_\_showTime() #初始化时间的显示**

**def \_\_showTime(self):**

**#更新倒计时的显示**

**if self.count\_time>=0:**

**self.display(self.count\_time) #显示时间**

**if self.count\_time==300:**

**self.thread1.start()**

**self.count\_time-=1**

**else:**

**self.thread2.start()**

**self.\_\_timer.stop()**

**def msg1(self):**

**#提示倒数5分钟**

**reply = QMessageBox.information(self, #使用infomation信息框**

**"提示：倒数5分钟",**

**"您只剩下5分钟了，快加快节奏啊！！！",**

**QMessageBox.Yes )**

**self.thread1.terminate()#结束新线程1**

**def msg2(self):**

**#提示时间用尽**

**reply = QMessageBox.information(self, #使用infomation信息框**

**"提示：时间用尽",**

**"时间用完了呢，O(∩\_∩)O",**

**QMessageBox.Yes )**

**self.thread2.terminate()#结束新线程2**

**class CountDownSign1(QThread):**

**#多线程运行辅助倒计时类1**

**sign1=pyqtSignal()#提示倒数5分钟信号**

**def \_\_init\_\_(self,parent=None):**

**super().\_\_init\_\_(parent)**

**#thread.start()后自动调用**

**def run(self):**

**self.sign1.emit()**

**class CountDownSign2(QThread):**

**#多线程运行辅助倒计时类2**

**sign2=pyqtSignal()#提示倒计时时间用完**

**def \_\_init\_\_(self,parent=None):**

**super().\_\_init\_\_(parent)**

**#thread.start()后自动调用**

**def run(self):**

**self.sign2.emit()**

**class SetCountTime(QPushButton):**

**#设置定时长度的**

**def \_\_init\_\_(self,parent=None):**

**super().\_\_init\_\_(parent)**

**class SetBackground(QPushButton):**

**#设置背景按钮类**

**def \_\_init\_\_(self,parent=None):**

**super().\_\_init\_\_(parent)**

**self.clicked.connect(self.BackgroundFileName)**

**self.fileName=''**

**def BackgroundFileName(self):**

**#打开弹窗来选取路径和图片文件的总名字**

**fname = QFileDialog.getOpenFileName(self,'请选择背景文件', '/')**

**if fname[0]:**

**self.fileName=fname[0]**

**class new\_progressbar(QtWidgets.QProgressBar):**

**#进度条类**

**def \_\_init\_\_(self,parent=None):**

**super().\_\_init\_\_(parent)**

**self.thread = QTimer(self) #新建一个定时器，来更新进度条**

**def start(self):**

**self.thread.start(1000) #设置定时间隔为1000ms即1s，并启动定时器**

**def show(self,time1,time2):**

**self.setValue(time1\*100//time2)**

**#如果倒计时结束，停止更新**

**if time1==-1:**

**self.thread.stop()**

**class Ui\_Dialog(object):**

**def setupUi(self, Dialog):**

**#设定窗口**

**Dialog.setObjectName("任务倒计时")**

**Dialog.resize(630, 377)**

**Dialog.setSizeGripEnabled(False)**

**Dialog.setModal(False)**

**#设定解释label**

**self.label\_explanation = QtWidgets.QLabel(Dialog)**

**self.label\_explanation.setGeometry(QtCore.QRect(50, 240, 681, 141))**

**self.label\_explanation.setObjectName("label\_explanation")**

**#设定当前时间显示模块**

**self.lcdNumber\_time = clock(Dialog)#建立clock类对象**

**self.lcdNumber\_time.setGeometry(QtCore.QRect(180, 30, 151, 71))**

**self.lcdNumber\_time.setObjectName("lcdNumber\_time")**

**#设定倒计时显示模块**

**self.lcdNumber = CountDown(Dialog)**

**self.lcdNumber.setGeometry(QtCore.QRect(180, 110, 151, 61))**

**self.lcdNumber.setObjectName("lcdNumber")**

**#设定“设定时间"按钮**

**self.pushButton = SetCountTime(Dialog)**

**self.pushButton.setGeometry(QtCore.QRect(430, 40, 121, 51))**

**self.pushButton.setObjectName("pushButton")**

**self.pushButton.setToolTip('输入<b>1-999999</b>之间一个数来倒计时，将在<b>倒数5分钟和倒数完</b>后分别进行提醒。')**

**#更换背景按钮**

**self.pushButton\_2 = SetBackground(Dialog)**

**self.pushButton\_2.setGeometry(QtCore.QRect(430, 110, 121, 51))**

**self.pushButton\_2.setObjectName("pushButton\_2")**

**self.pushButton\_2.setToolTip('请选择<b>名字较短的jpg格式</b>图片！')**

**#进度条模块**

**self.progressBar = new\_progressbar(Dialog)**

**self.progressBar.setGeometry(QtCore.QRect(180, 200, 191, 21))**

**self.progressBar.setValue(0)**

**self.progressBar.setObjectName("progressBar")**

**#不变的label**

**self.label = QtWidgets.QLabel(Dialog)**

**self.label.setGeometry(QtCore.QRect(50, 30, 111, 61))**

**self.label.setObjectName("label")**

**self.label\_2 = QtWidgets.QLabel(Dialog)**

**self.label\_2.setGeometry(QtCore.QRect(70, 110, 111, 61))**

**self.label\_2.setObjectName("label\_2")**

**self.label\_3 = QtWidgets.QLabel(Dialog)**

**self.label\_3.setGeometry(QtCore.QRect(0, 180, 161, 61))**

**self.label\_3.setObjectName("label\_3")**

**#显示设定的时间的窗口**

**self.label\_designed\_time = QtWidgets.QLabel(Dialog)**

**self.label\_designed\_time.setGeometry(QtCore.QRect(430, 200, 230, 31))**

**self.label\_designed\_time.setObjectName("label\_designed\_time")**

**self.retranslateUi(Dialog)**

**QtCore.QMetaObject.connectSlotsByName(Dialog)**

**def retranslateUi(self, Dialog):**

**\_translate = QtCore.QCoreApplication.translate**

**Dialog.setWindowTitle(\_translate("Dialog", "任务倒计时"))**

**self.label\_explanation.setText(\_translate("Dialog", "<html><head/><body><p><span style=\" font-size:12pt; font-weight:600;color:#ff0000;\">设计者：机械1503班 胡斌 U201510603 </span></p><p><span style=\" font-size:12pt; font-weight:600;color:#ff0000;\">用于计时，来提醒你时间到了，该做下一件事了。</span></p>"))**

**self.pushButton.setText(\_translate("Dialog", "设置时长"))**

**self.pushButton\_2.setText(\_translate("Dialog", "设置背景"))**

**self.label.setText(\_translate("Dialog", "<html><head/><body><p><span style=\" font-size:14pt; font-weight:600; color:#ff0000;\">当前时间：</span></p></body></html>"))**

**self.label\_2.setText(\_translate("Dialog", "<html><head/><body><p><span style=\" font-size:14pt; font-weight:600; color:#ff0000;\"> 倒计时：</span></p></body></html>"))**

**self.label\_3.setText(\_translate("Dialog", "<html><head/><body><p><span style=\" font-size:14pt; font-weight:600; color:#ff0000;\">倒计时进度条：</span></p></body></html>"))**

**self.label\_designed\_time.setText(\_translate("Dialog", "<html><head/><body><p><span style=\" font-size:12pt; font-weight:600;\">设定时间为0秒 </span></p></body></html>"))**

**if \_\_name\_\_=='\_\_main\_\_':**

**app = QtWidgets.QApplication(sys.argv)**

**Dialog = QtWidgets.QDialog()**

**ui = Ui\_Dialog()**

**ui.setupUi(Dialog)**

**#设定初始背景**

**Dialog.setStyleSheet('QDialog{border-image:url(./image/3.jpg)}')**

**Dialog.show()**

**#设定“设定时间"按钮触发的第二个窗口的设定**

**Dialog2 = QtWidgets.QDialog()**

**ui2 = sub\_window.Ui\_Dialog\_second()**

**ui2.setupUi\_second(Dialog2)**

**ui.pushButton.clicked.connect(Dialog2.show)#按下后弹出子窗口来设置时间**

**#按下设置背景按钮后，选取图片路径并进行背景的替换**

**ui.pushButton\_2.clicked.connect(lambda:ui.pushButton\_2.fileName==''**

**and [Dialog.setStyleSheet('QDialog{border-image:url(./image/3.jpg)}')] or [Dialog.setStyleSheet('QDialog{border-image:url('+ui.pushButton\_2.fileName+')}')][0])**

**#子窗口的“ok”按钮按下事件，连接到开始倒计时函数(三目选择，使得输入为空时，默认为输入0)**

**ui2.pushButton.clicked.connect(lambda:ui2.lineEdit.text()=='' \**

**and [ui.lcdNumber.CountStart(0)] \**

**or [ui.lcdNumber.CountStart(int(ui2.lineEdit.text()))][0])**

**#子窗口的“ok”按钮按下事件，连接到改变倒计时总时长的显示(三目选择，使得输入为空时，默认为输入0)**

**ui2.pushButton.clicked.connect(lambda:ui2.lineEdit.text()==''\**

**and [ui.label\_designed\_time.setText("<html><head/><body><p><span style=\" font-size:12pt; font-weight:600;\">设定时间为0秒 </span></p></body></html>")] \**

**or [ui.label\_designed\_time.setText("<html><head/><body><p><span style=\" font-size:12pt; font-weight:600;\">设定时间为"+ui2.lineEdit.text()+"秒 </span></p></body></html>")][0])**

**#子窗口的“ok”按钮按下事件，进度条开始更新**

**ui2.pushButton.clicked.connect(ui.progressBar.start)**

**ui.progressBar.thread.timeout.connect(lambda:ui2.lineEdit.text()=='' \**

**and [ui.progressBar.show()] \**

**or ui.progressBar.show(int(ui2.lineEdit.text())-ui.lcdNumber.count\_time,int(ui2.lineEdit.text())))**

**sys.exit(app.exec\_())**

**sub\_window.py:**

**# -\*- coding: utf-8 -\*-**

**# Form implementation generated from reading ui file 'C:\Users\HB\Desktop\python作业\hb2.ui'**

**#**

**# Created by: PyQt5 UI code generator 5.9.2**

**#**

**# WARNING! All changes made in this file will be lost!**

**import sys**

**from PyQt5 import QtCore, QtGui, QtWidgets**

**from PyQt5.QtWidgets import QLCDNumber,QPushButton,QWidget,QDesktopWidget**

**from PyQt5.Qt import QPoint, QTimer, QTime, QRect**

**from PyQt5.QtCore import Qt**

**from PyQt5.QtGui import QFont,QIntValidator**

**class Ui\_Dialog\_second(object):**

**def setupUi\_second(self, Dialog):**

**Dialog.setObjectName("Dialog")**

**Dialog.resize(421, 246)**

**#输入区**

**self.lineEdit = QtWidgets.QLineEdit(Dialog)**

**self.lineEdit.setMaxLength(6)#限制最大字符串长度**

**pIntvalidator=QIntValidator(self.lineEdit)#限制输入为整数**

**pIntvalidator.setRange(1,999999)#限制的范围**

**self.lineEdit.setValidator(pIntvalidator)#限制输入6位以内整数**

**self.lineEdit.setGeometry(QtCore.QRect(60, 20, 291, 41))**

**self.lineEdit.setObjectName("lineEdit")**

**#输入提示label**

**self.label = QtWidgets.QLabel(Dialog)**

**self.label.setGeometry(QtCore.QRect(60, 90, 331, 51))**

**self.label.setObjectName("label")**

**#确认按钮**

**self.pushButton = QtWidgets.QPushButton(Dialog)**

**self.pushButton.setGeometry(QtCore.QRect(70, 160, 121, 51))**

**self.pushButton.setObjectName("pushButton")**

**self.pushButton.clicked.connect(Dialog.close)**

**#取消按钮**

**self.pushButton\_Cancel = QtWidgets.QPushButton(Dialog)**

**self.pushButton\_Cancel.setGeometry(QtCore.QRect(230, 160, 121, 51))**

**self.pushButton\_Cancel.setObjectName("pushButton\_Cancel")**

**self.pushButton\_Cancel.clicked.connect(Dialog.close)**

**self.retranslateUi(Dialog)**

**QtCore.QMetaObject.connectSlotsByName(Dialog)**

**def retranslateUi(self, Dialog):**

**\_translate = QtCore.QCoreApplication.translate**

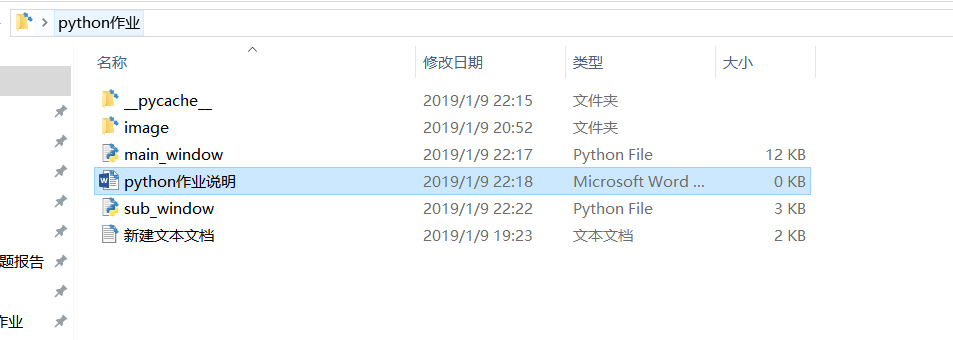
**Dialog.setWindowTitle(\_translate("Dialog", "Dialog"))**

**self.label.setText(\_translate("Dialog", "<html><head/><body><p>请输入一个6位以内的正整数，作为计数的秒数</p></body></html>"))**

**self.pushButton.setText(\_translate("Dialog", "OK"))**

**self.pushButton\_Cancel.setText(\_translate("Dialog", "Cancel"))**

**整个文件夹布局如下图，image文件夹用来存放初始化的背景3.jpg**



**运行main\_window.py即可运行。总效果如图1：**



**图1**

**将鼠标移到“设置时长”和“设置背景”上，会显示提示框，如下图2、3：**

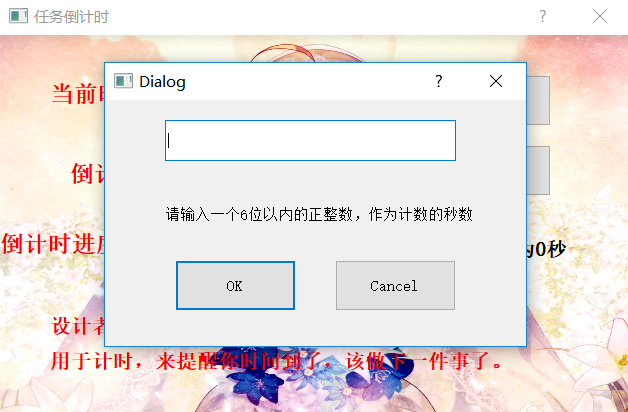


**图2**



**图3（也不时一定要jpg格式的图片，但仅测试了jpg格式的图片）**

**点击“设置时长”按钮后，弹出sub\_window，如图4：**



**图4**

**设定时间为空时，不变。设定时间大于300s时，在倒数到300s（即倒数5分钟）时，弹窗提示，如图5：**



**图5**

**倒数到0s时，也会弹窗提醒，如图6：**



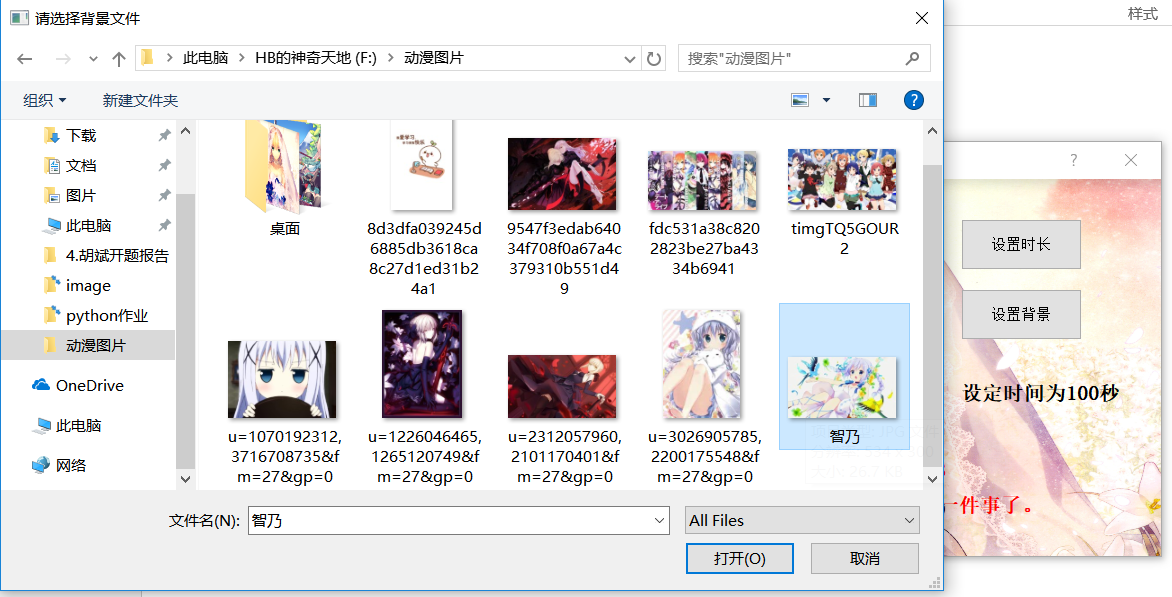
**图6**

**倒计时的同时，会在窗口上显示你设置的时间，以及进度条会跟着变动，如图7：**



**图7**

**点击设置背景，弹窗选取背景文件，如图8、9：**



**图8**



**图9**