1. 新建qt工程文件

图形用户界面, 文本, 应用程序

描述已自动生成

2. \*.ui文件中的QWidget对象进行设置（添加label和push button两个控件）

日历

描述已自动生成

3.在widget.cpp中添加相应代码

QFont font;

font.setFamily("Noto Sans CJK SC");

font.setItalic(true);

font.setBold(false);

font.setPixelSize(32);

ui->titleLabel->setFont(font);

ui->titleLabel->setWordWrap(false);//关闭自动换行

QString string = tr("物联网灯光控制系统");

//截断、省略文本的作用

//QString str = ui->titleLabel->fontMetrics().elidedText(string, Qt::ElideRight, 120);

ui->titleLabel->setText(string);

//label\_2

ui->picLabel->setPixmap(QPixmap(":/resources/img/lightoff.png"));

qDebug("显示off已经执行");

QMovie \*movie = new QMovie(":/resources/img/simulation.gif");

ui->gifLabel->setMovie(movie);

movie->start();

}

Widget::~***Widget***()

{

delete ui;

}

void Widget::**on\_onButton\_clicked**()

{

ui->picLabel->setPixmap(QPixmap(":/resources/img/lighton.png"));

qDebug("显示on已经执行");

}

void Widget::**on\_offButton\_clicked**()

{

ui->picLabel->setPixmap(QPixmap(":/resources/img/lightoff.png"));

qDebug("显示off已经执行");

}

4.最终效果图

图形用户界面

中度可信度描述已自动生成 图形用户界面

描述已自动生成