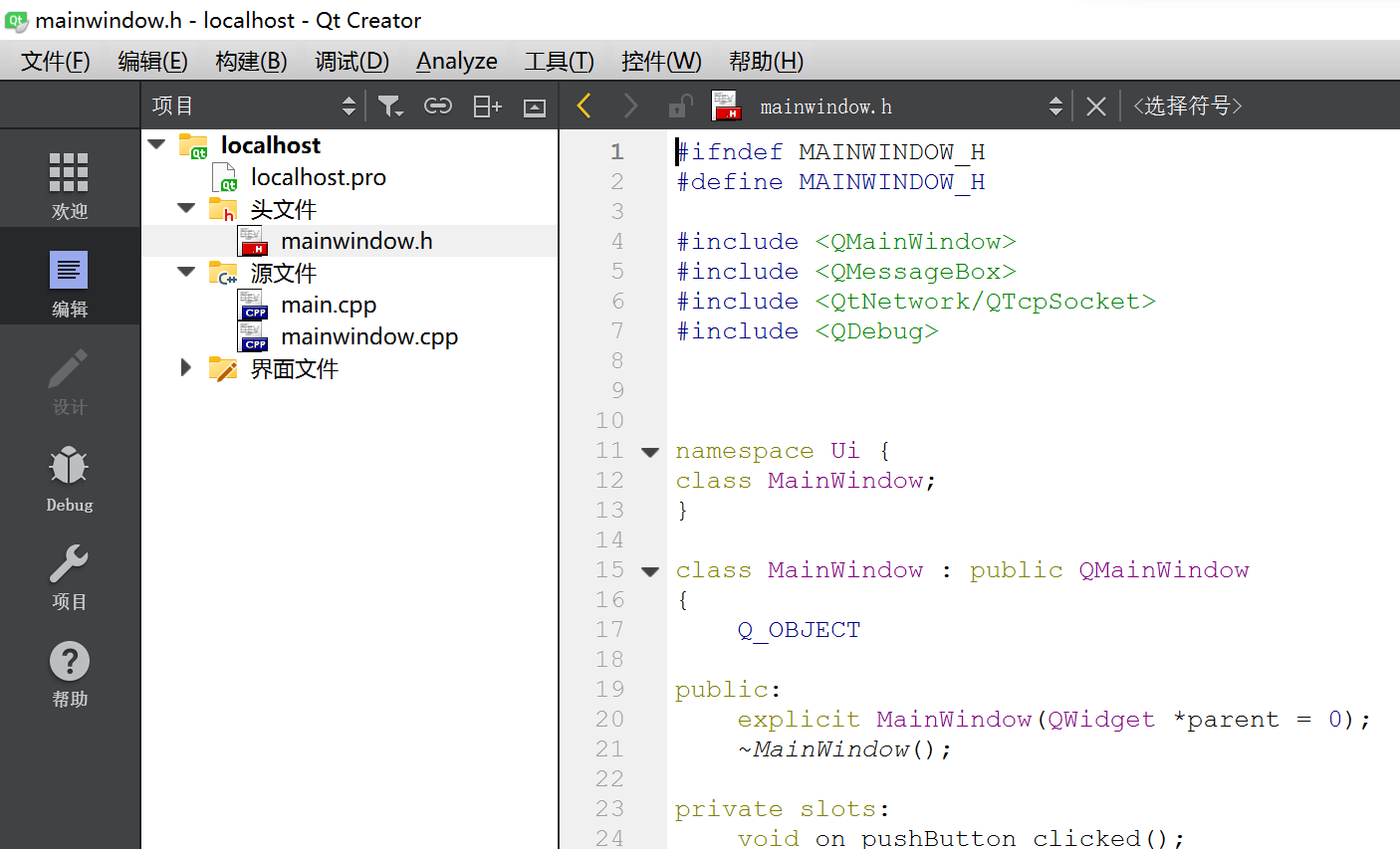
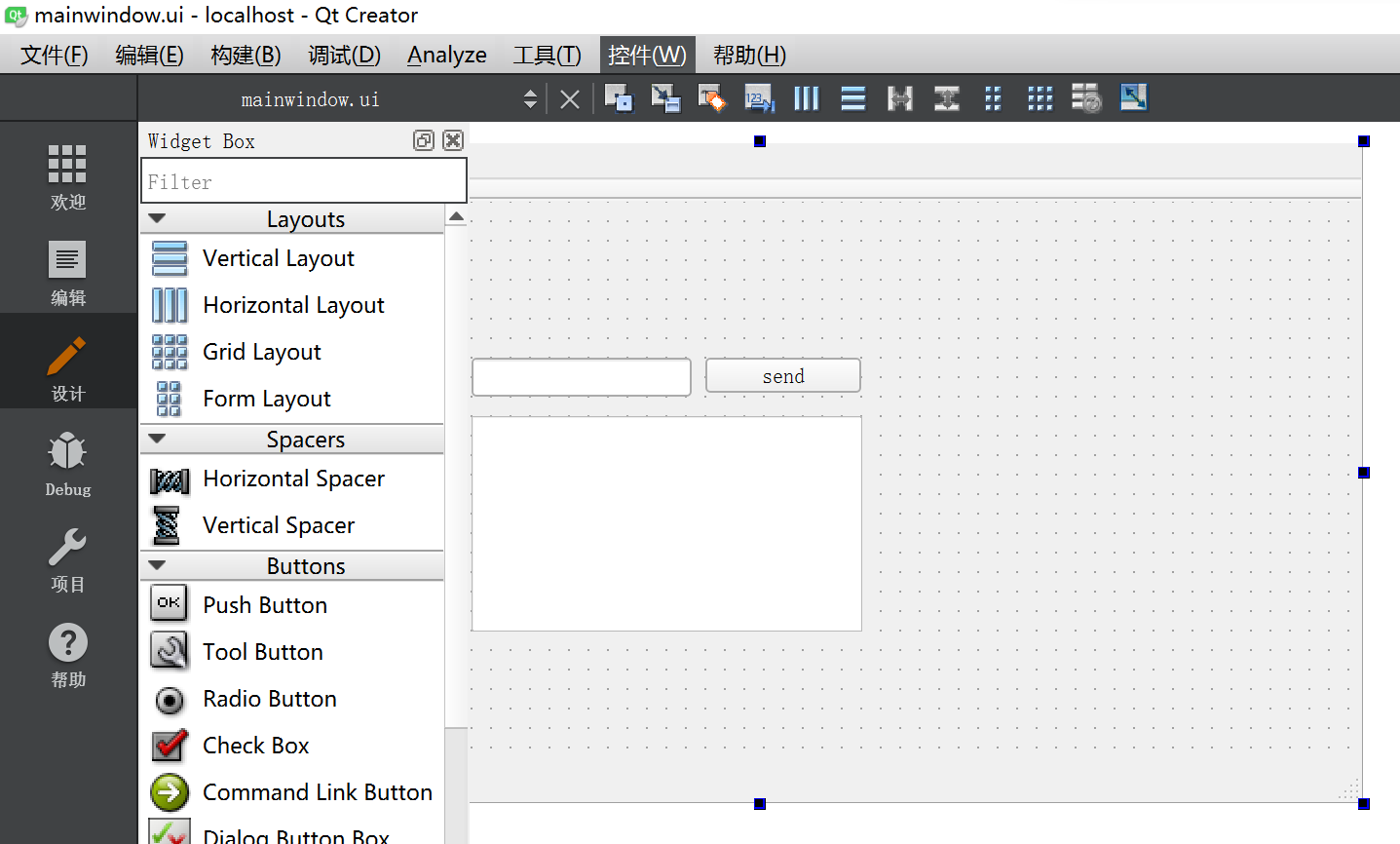
TCPSocket的接收与发送

1.创建项目



2.设计.ui

3.在.pro里面增加

QT += core gui network

4.在.h 里面编写

private slots:

void on\_pushButton\_clicked();

//发送

void read\_data();

//接收

private:

Ui::MainWindow \*ui;

QTcpSocket \*TCPSocket;

};

5.在.cpp 文件中编写代码

MainWindow::MainWindow(QWidget \*parent) :

QMainWindow(parent),

ui(new Ui::MainWindow)

{

ui->setupUi(this);

TCPSocket =new QTcpSocket();

//TCPSocket->connectToHost(QHostAddress(HOST\_Address),Port\_Number);

TCPSocket->*connectToHost*(QHostAddress ::LocalHost,8080);

TCPSocket->*open*(QIODevice::ReadWrite);

if(TCPSocket->isOpen())

{

QMessageBox::information(this,"Qt With wk","Connected To The Server.");

}

else

{

QMessageBox::information(this,"Qt With wk","Not Connected To The Server.");

}

//你需要使用connect函数来完成连接。在调用connect函数之前，你需要先创建一个socket，然后使用bind函数将socket绑定到本地地址和端口，最后调用connect函数连接远程服务器。连接成功后，你就可以使用该socket与远程服务器通信了。需要注意的是，在使用connect函数之前，你应该先确保远程服务器已经处于监听状态

connect(TCPSocket,SIGNAL(readyRead()),this,SLOT(read\_data()));

}

MainWindow::~*MainWindow*()

{

delete ui;

}

void MainWindow::on\_pushButton\_clicked()

{

if(TCPSocket )

{

if(TCPSocket->isOpen())

{ QString writeDate=ui->lineEdit->text().append(char(10));

TCPSocket->write(writeDate.toStdString().c\_str());

//toLocal8Bit

//toStdString

}

else{

QMessageBox ::information(this,"QT with me","error:"+TCPSocket->error());

}

}

else

{

QMessageBox ::information(this,"QT with me","error:"+TCPSocket->error());

}

}

void MainWindow::read\_data(){

if(TCPSocket)

{

if(TCPSocket->isOpen())

{

QByteArray a=TCPSocket->readAll();

QString Message=QString::fromLocal8Bit(a);

ui->textEdit->append(Message);

}

}

}

6.成果展示

