QtCore：

这个模块是非GUI的核心库，用来处理时间、文件、目录、各种类型的数据、流（stream）、URLs，mime类型、线程和进程等。常用的类包括：QObject（所有Qt对象的基类，支持信号和槽机制）、QTimer（用于定时器操作，可以实现定时执行某个函数或操作）等。

QtGui：

这个模块有窗口系统集成、事件处理、2D图形，基本图像、字体、文本的类。它提供了2D图形、图像、字体和高级排版的类。QPainter支持绘制线条、多边形、矢量路径、图像和文本。

QtWidgets：

这个模块提供了创建经典风格的用户界面的类。它提供了一套UI元素来创建经典的桌面风格的用户界面。常用的类包括QApplication、QWidget、QLabel、QPushButton、QLineEdit等。

QtDBus：

这个模块是使用D-Bus处理IPC通讯的类。D-Bus是一个IPC和RPC机制，最初为Linux开发，用于替代现有和竞争的IPC解决方案，以一个统一的协议。

QtNetwork：

这个模块是网络编程类，这些类使网络编程变得更容易，可移植性也更好，方便了TCP/IP和UDP服务端和客户端编程。

QtHelp：

这个模块包含了创建、查看和搜索文档的类。

QtXml：

这个模块包含了处理XML文件的类，实现了SAX和DOM API。

QtSvg：

这个模块提供了显示SVG的类，可缩放矢量图形 (SVG)是一种描述二维图像和图像应用的XML语言。

QtSql：

这个模块提供了数据库的类，例如QSqlRelationalTableModel类提供了查看和更改SQL表中数据的模型。

QtTest：这个模块提供了可以对PyQt6应用进行单元测试的工具。Qt Test是一个用于单元测试基于Qt的应用程序和库的框架。

**QPushButton**:  
用户点击时触发操作的按钮控件。通过clicked.connect()连接函数至点击事件。

button = QPushButton("点击我")

button.clicked.connect(some\_function)

**QLabel**:  
用于展示文本或图像的控件。使用setText()方法设定标签文本。

label = QLabel()

label.setText("这是一个标签")

**QLineEdit**:  
单行文本输入框控件。通过text()方法获取输入框内容。

line\_edit = QLineEdit()

text = line\_edit.text()

**QProgressBar**:

进度条控件，利用setValue()方法设定进度值。

progress\_bar = QProgressBar()

progress\_bar.setValue(50)

**QComboBox**:  
下拉选择框控件。利用addItem()添加选项，currentText()获取当前选中项文本。

combo\_box = QComboBox()

combo\_box.addItem("选项1")

combo\_box.addItem("选项2")

selected\_text = combo\_box.currentText()