

1.添加模块

在Qt中使用QChart模块，需要在.pro文件当中加入charts

```
1 QT += charts
```

同时在使用类中添加对应的头文件并引入空间

```
1 #include <QtCharts>           // 引入所有QtCharts绘制的头文件，也可以单独引入某一个头文件
2
3 // 两种引入命名空间方法
4 using namespace QtCharts;     // 使用using引入命名空间
5 QT_CHARTS_USE_NAMESPACE      // 使用内置宏引入命名空间
```

2.QtCharts模块的类

参考博客：https://blog.csdn.net/qq_43627907/article/details/124389945

Qt官方文档：<https://doc.qt.io/qt-5.15/qtcharts-module.html>

类名	作用
QAbstractAxis	用于专用轴类的基类
QAbstractBarSeries	所有条形系列类的抽象父类
QAbstractSeries	所有 Qt Chart 系列的基类
QAreaLegendMarker	区域系列的图例标记
QAreaSeries	在面积图中显示数据
QBarCategoryAxis	将类别添加到图表的轴
QBarLegendMarker	条形系列的图例标记
QBarSeries	将一系列数据显示为按类别分组的垂直条
QBarSet	表示条形图中的一组条形
QBoxPlotLegendMarker	箱线图系列的图例标记
QBoxPlotSeries	以盒须图显示数据
QBoxSet	表示盒须图中的一项
QCandlestickLegendMarker	烛台系列的图例标记
QCandlestickModelMapper	烛台系列的抽象模型映射器类

QCandlestickSeries	将数据呈现为烛台
QCandlestickSet	表示烛台图表中的单个烛台项目
QCategoryAxis	在轴上放置命名范围
QChart	管理图表系列、图例和轴的图形表示
QChartView	可以显示图表的独立小部件
QDateTimeAxis	将日期和时间添加到图表的轴
QHBarModelMapper	条形系列卧式模型映射器
QHBoxPlotModelMapper	箱线图系列的水平模型映射器
QHCandlestickModelMapper	烛台系列的水平模型映射器
QHPieModelMapper	饼系列的水平模型映射器
QHXYModelMapper	线、样条和散点系列的水平模型映射器
QHorizontalBarSeries	将一系列数据显示为按类别分组的水平条
QHorizontalPercentBarSeries	将一系列分类数据显示为每个类别的百分比
QHorizontalStackedBarSeries	将一系列数据显示为水平堆叠的条形，每个类别一个条形
QLegend	显示图表的图例
QLegendMarker	可用于访问图例中的标记的抽象对象
QLineSeries	以折线图呈现数据
QLogValueAxis	向图表的轴添加对数刻度
QPercentBarSeries	将一系列分类数据显示为每个类别的百分比
QPieLegendMarker	饼图系列的图例标记
QPieSeries	在饼图中显示数据
QPieSlice	表示饼系列中的单个切片
QPolarChart	在极坐标图中显示数据
QScatterSeries	在散点图中显示数据
QSplineSeries	将数据显示为样条图
QStackedBarSeries	将一系列数据显示为垂直堆叠的条形，每个类别一个条形

QVBarModelMapper	条形系列立式模型映射器
QVBoxPlotModelMapper	箱线图系列的垂直模型映射器
QVCandlestickModelMapper	烛台系列的垂直模型映射器
QVPieModelMapper	饼系列的垂直模型映射器
QVXYModelMapper	线、样条和散点系列的垂直模型映射器
QValueAxis	将值添加到图表的轴
QXYLegendMarker	线、样条线或散点系列的图例标记
QXYSeries	线、样条和散点系列的基类

3.图表当中的常用设置

(1) 设置动画效果

```

1    m_chart->setAnimationOptions(QChart::AllAnimations);//所有动画效果
2    m_chart->setAnimationOptions(QChart::NoAnimation);//无动画效果
3    m_chart->setAnimationOptions(QChart::SeriesAnimations);//系列动画效果
4    m_chart->setAnimationOptions(QChart::GridAxisAnimations);//栅格动画效果

```

(2) 设置主题

```

1    m_chart->setTheme(QChart::ChartThemeLight);//设置为白色主题
2    m_chart->setTheme(QChart::ChartThemeBlueCerulean);
3    m_chart->setTheme(QChart::ChartThemeBlueIcy);
4    m_chart->setTheme(QChart::ChartThemeBlueNcs);
5    m_chart->setTheme(QChart::ChartThemeBrownSand);
6    m_chart->setTheme(QChart::ChartThemeDark);
7    m_chart->setTheme(QChart::ChartThemeHighContrast);
8    m_chart->setTheme(QChart::ChartThemeQt);

```

(3) 图例设置

```

1    m_chart->legend()->setVisible(true);
2    m_chart->legend()->setAlignment(Qt::AlignBottom);//设置为底部对齐
3    m_chart->legend()->setAlignment(0);//设置为无说明
4    m_chart->legend()->setBackgroundVisible(true);//设置背景是否可视
5    m_chart->legend()->setAutoFillBackground(true);//设置背景自动填充
6    m_chart->legend()->setColor(QColor(223,223,223));//设置颜色
7    m_chart->legend()->setLabelColor(QColor(0,128,255));//设置标签颜色
8    m_chart->legend()->setMaximumHeight(50);

```

(4) 标题设置

```
1    m_chart->setTitle("折线图");
2    m_chart->setTitleFont(QFont("微软雅黑"));
3    m_chart->setTitleBrush(QBrush(QColor(0,0,255)));
```

(5) 坐标轴设置

```
1    QValueAxis *axisX=new QValueAxis;
2    axisX->setTitleText("X轴");//设置X轴标题
3    axisX->setRange(0,10);//设置X轴范围
4    axisX->setTickCount(11);//设置主刻度个数
5    axisX->setLineVisible(true);//设置轴线和刻度线可见
6    axisX->setGridLineVisible(true);//设置网格线可见
```

(6) 设置序列属性

```
1    //设置序列属性
2    series0->setName("序列1");//名称
3    series0->setColor(QColor(255,0,0));//颜色
4    series0->setVisible(true);
5    series0->setPointLabelsFormat("(@xPoint,@yPoint)");
6    series0->setPointLabelsVisible(true);
```

(7) 其他

```
1    m_chart->setDropShadowEnabled(true);//背景阴影
2    m_chart->setAutoFillBackground(true);//设置背景自动填充
3    m_view->setRenderHint(QPainter::Antialiasing,true);//设置为防锯齿
```