案例：图片下载与显示

目标： 创建一个程序，能够下载多张图片并在界面上显示它们。

**功能：**

界面设计： 创建一个简单的 Qt 窗口界面，包括一个按钮 "Download" 和一个用于显示图片的标签。

**后台下载**： 创建一个继承自 QThread 的子类，用于后台下载图片。在这个子类中，你可以使用 Qt 的网络模块 QNetworkAccessManager 来进行图片下载。

**显示图片**： 在主线程中，将下载的图片数据显示到界面上的标签中。使用 Qt 的信号和槽机制来实现线程间的通信，确保在主线程中更新 UI。

多图片下载： 可以创建多个后台下载线程，每个线程下载不同的图片，这样可以模拟并行下载。

**错误处理：** 考虑异常情况，例如网络连接错误，确保你的程序能够处理这些情况并进行适当的错误提示。

**取消下载：** 添加一个 "Cancel" 按钮，可以中断下载过程。

**高级任务：** 可以考虑使用 Qt 的高级类，例如 QThreadPool，来管理线程池，以及优化多线程的性能。

通过这个案例，你可以学习以下内容：

1.如何在 Qt 中创建多线程并管理线程生命周期。

2.如何使用 Qt 的网络模块进行网络请求。

3.如何在多线程中使用信号和槽来实现线程间的通信。

4.如何在主线程中更新界面以显示下载的图片。

5.如何处理多线程情况下的错误和异常。

6.如何使用 Qt 的线程池来优化多线程应用。

7.在编写代码时，记得查阅 Qt 的官方文档（qthread 是重点）以获取更详细的信息和示例代码。