

# Qt 编程基础

2019 年 10 月 26 日星期六下午第四次验收

# 目录

① Qt 简介

② 使用 Qt

③ 示例

④ 后记

## 学习目标

- ① 掌握 Qt 开发环境的基本使用
- ② 了解基本的 window 界面框架
- ③ 理解信号与槽机制
- ④ 理解事件机制
- ⑤ 理解模型/视图结构
- ⑥ 学会简单动画绘图

# 1 Qt 简介

## 跨平台 C++ 图形用户界面应用程序开发框架

- 它既可以开发 GUI 程序，也可用于开发非 GUI 程序，比如控制台工具和服务器。
- Qt 是面向对象的框架，使用元对象编译器生成扩展代码，易于扩展，允许组件编程。

# 1 Qt 简介

## 跨平台 C++ 图形用户界面应用程序开发框架

- 它既可以开发 GUI 程序，也可用于开发非 GUI 程序，比如控制台工具和服务器。
- Qt 是面向对象的框架，使用元对象编译器生成扩展代码，易于扩展，允许组件编程。



## 跨平台

桌面 OS: Windows, Linux, Mac; 移动 OS: iOS, Android, WP; 嵌入式 OS: QNX, VxWorks

# 1 Qt 简介

## 跨平台 C++ 图形用户界面应用程序开发框架

- 它既可以开发 GUI 程序，也可用于开发非 GUI 程序，比如控制台工具和服务器。
- Qt 是面向对象的框架，使用元对象编译器生成扩展代码，易于扩展，允许组件编程。



### 跨平台

桌面 OS: Windows, Linux, Mac; 移动 OS: iOS, Android, WP; 嵌入式 OS: QNX, VxWorks

### 集成开发平台

Design, Code, Debug & Deploy Quickly

# 1 Qt 简介

## 跨平台 C++ 图形用户界面应用程序开发框架

- 它既可以开发 GUI 程序，也可用于开发非 GUI 程序，比如控制台工具和服务器。
- Qt 是面向对象的框架，使用元对象编译器生成扩展代码，易于扩展，允许组件编程。



### 跨平台

桌面 OS: Windows, Linux, Mac; 移动 OS: iOS, Android, WP; 嵌入式 OS: QNX, VxWorks

### 集成开发平台

Design, Code, Debug & Deploy Quickly

应用: WPS、Skype、豆瓣电台、虾米音乐、暴雪的战网客户端、VirtualBox、Opera、咪咕音乐、Google 地图、Adobe Photoshop Album

# 1 Qt 简介

## 软件下载

Qt 版本 Qt 5



# 1 Qt 简介

## 软件下载

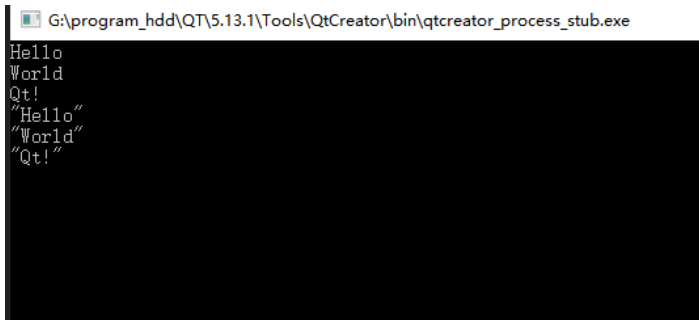
Qt 版本 Qt 5

## 在线教程

- <http://www.devbean.net/2012/08/qt-study-road-2-catalog/>
- <http://doc.qt.io/qt-5/gettingstarted.html>
- <http://doc.qt.io/qt-5/examples-widgets.html>

## 2 使用 Qt

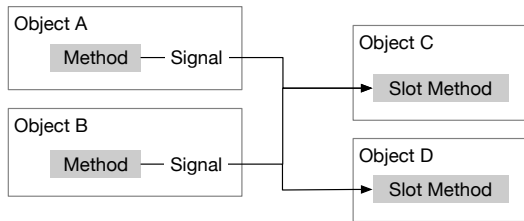
### (1) 创建 Hello World Qt!



```
G:\program_hdd\QT\5.13.1\Tools\QtCreator\bin\qtcreator_process_stub.exe
Hello
World
Qt!
"Hello"
"World"
"Qt!"
```

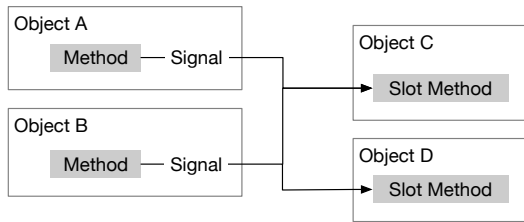
## 2 使用 Qt

(2) 使用信号与槽机制实现简单的对象间通讯



## 2 使用 Qt

### (2) 使用信号与槽机制实现简单的对象间通讯



#### 信号 signals

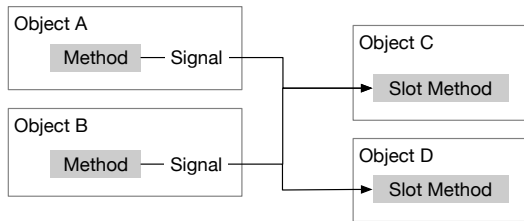
- 信号由对象发射
- 基类的信号被派生类继承
- 无实现，类似 C++ 纯虚函数

#### 槽 slots

- 槽是普通的 C++ 成员函数
- 多个信号可以与其相关联
- 槽可以有参数，但参数不能有缺省值。

## 2 使用 Qt

### (2) 使用信号与槽机制实现简单的对象间通讯



#### 信号 signals

- 信号由对象发射
- 基类的信号被派生类继承
- 无实现，类似 C++ 纯虚函数

#### 槽 slots

- 槽是普通的 C++ 成员函数
- 多个信号可以与其相关联
- 槽可以有参数，但参数不能有缺省值。

信号-槽链接：connect(sender, signal, receiver, slot)

signal 接口函数声明; slot 响应 signal 的实现; sender 发送消息的对象; receiver 接收消息的对象

## 2 使用 Qt

(2) 使用信号与槽机制实现简单的对象间通讯

### 示例 1-NewsPaper

G:\program\_hdd\QT\5.13.1\Tools\QtCreator\bin\qtcreator\_process\_stub.exe

Receives Newspaper: "Newspaper A" QDate("2019-10-15")

## 2 使用 Qt

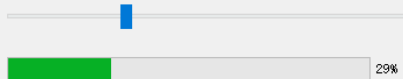
(2) 使用信号与槽机制实现简单的对象间通讯

### 示例 1-NewsPaper

```
G:\program_hdd\QT\5.13.1\Tools\QtCreator\bin\qtcreator_process_stub.exe  
Receives Newspaper: "Newspaper A"   QDate("2019-10-15")
```

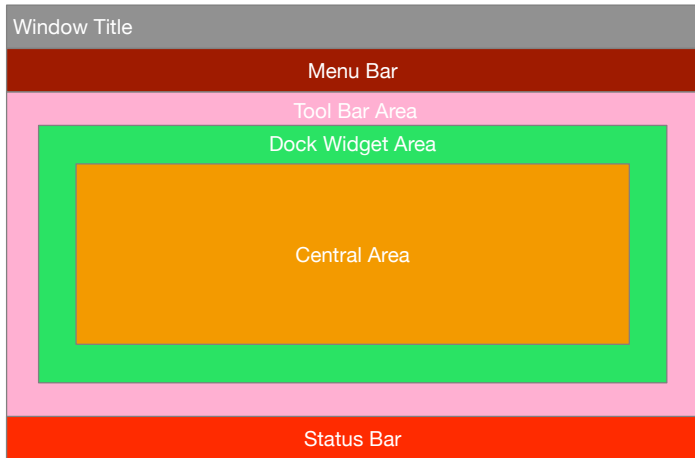
### 示例 2-SliderSignals

MainWindow



## 2 使用 Qt

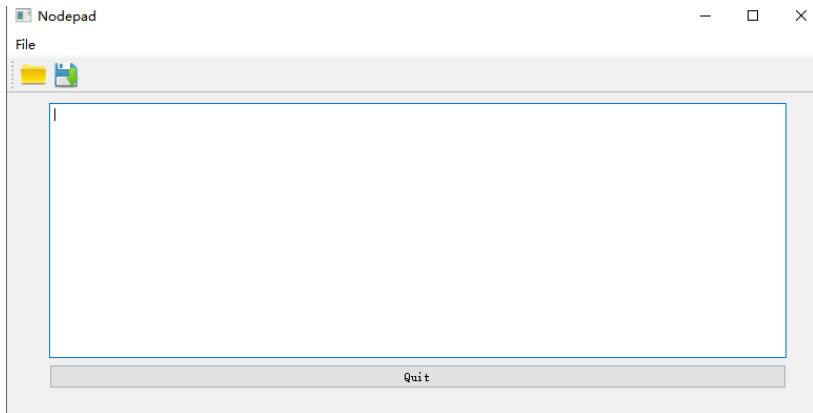
### (3) 创建 GUI 界面





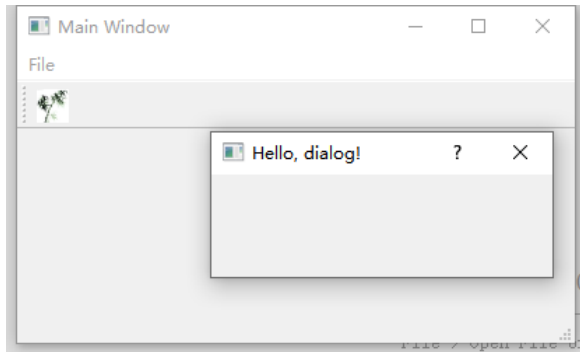
## 2 使用 Qt

### (3) 创建 GUI 界面示例-Notepad



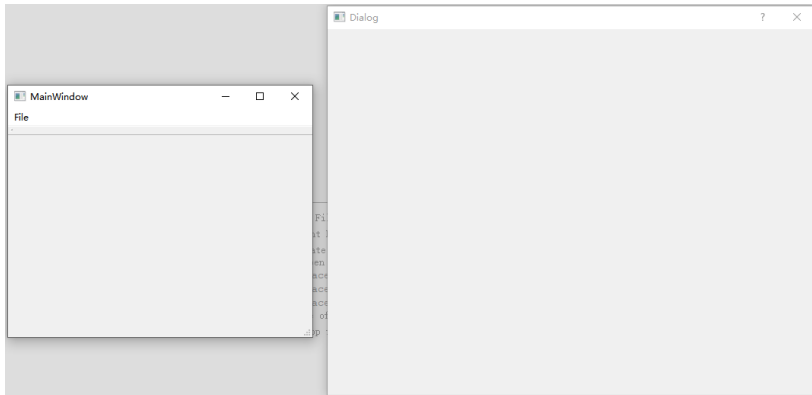
## 2 使用 Qt

### (4) 动作 (action)-示例



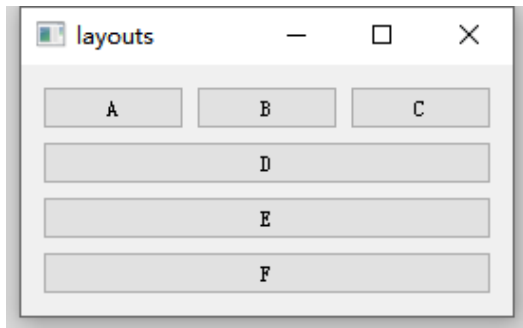
## 2 使用 Qt

### (5) 使用设计模式添加控件示例-dialog



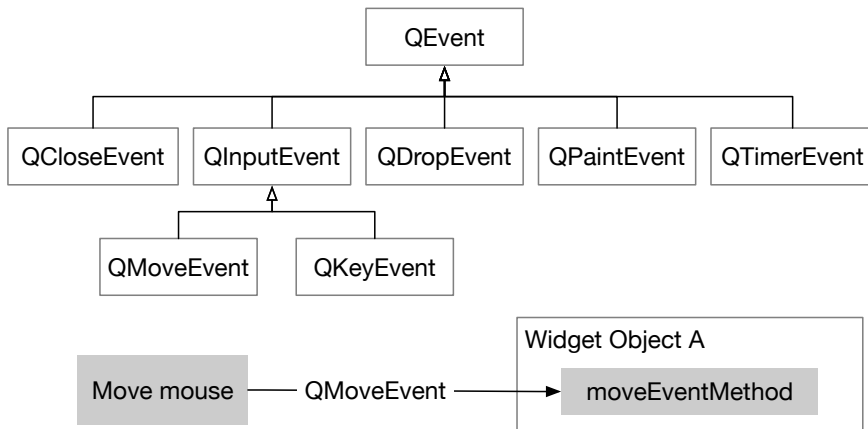
## 2 使用 Qt

### (6) 使用代码编辑 UI 控件布局



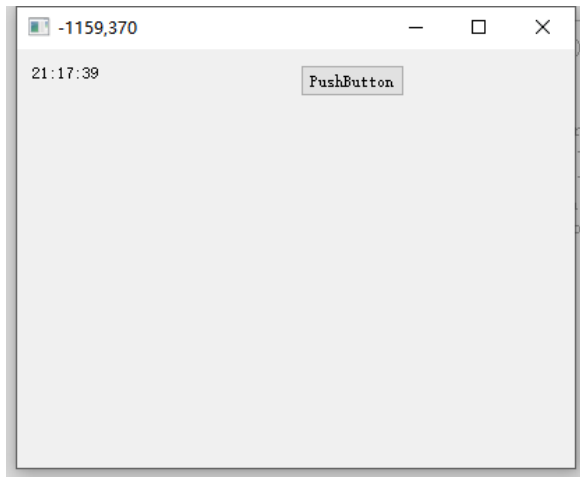
## 2 使用 Qt

### (7) 事件 (event)



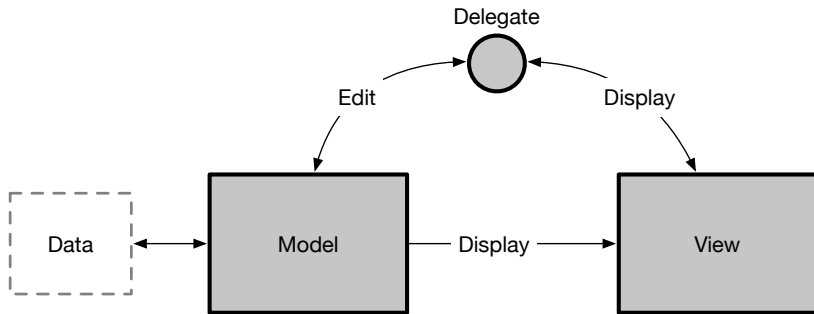
## 2 使用 Qt

### (7) 事件 (event)-示例



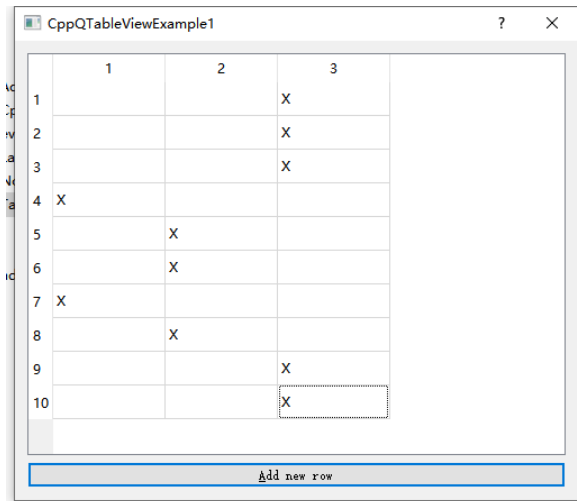
## 2 使用 Qt

### (8) 模型/视图结构 (Model/View)



## 2 使用 Qt

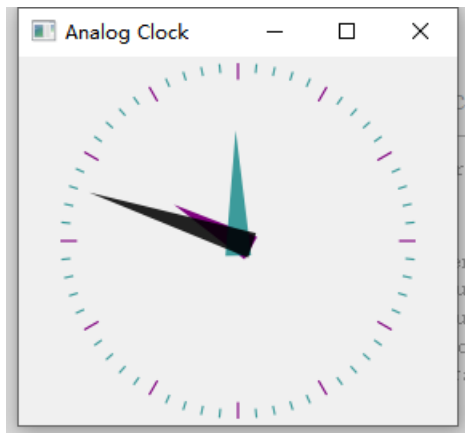
### (8) 模型/视图结构 (Model/View)-示例





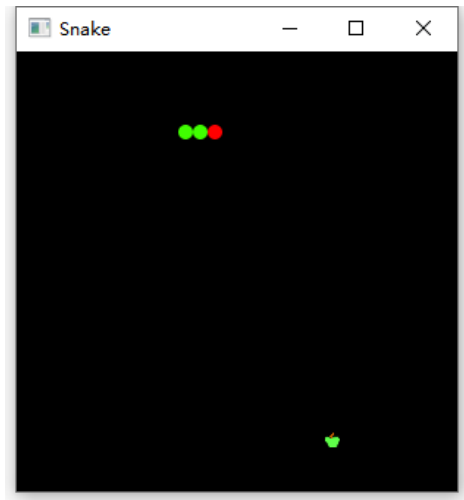
### 3 示例

#### 示例一：时钟



### 3 示例

示例二：贪吃蛇



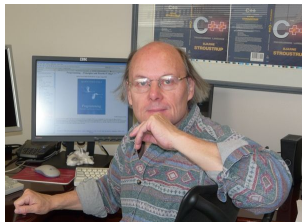
## 4 后记

### C++ 编程风格

- 数据抽象
- 过程化
- 面向对象
- 泛型编程

### C++ 知识层级

- 层级一：语法/语意
- 层级二：专家经验
- 层级三：底层机制
- 层级四：设计观念的复用



### 网友的评价

- **精通 C++ 是一个艰巨的任务。**为什么 C++ 比别的语言难学这么多？其实这基本上是因为 C++ 他爹 Bjarne Stroustrup 说过的一句话“我特别讨厌语言的设计者把自己的喜好强加给用户” C++ 能够自由的让你放弃某些部分，而别的语言会阻止你放弃某些部分。
- 谷歌工程师对 C++ 的掌握有两个级别：拥有 C++ 的 readability（可读性）认证；顾问级 C++ 程序猿
- Never trust a programmer who says he knows C++

## 4 后记

### 推荐 C++ 书籍

- 层级一：语法/语意 (C++)
  - ① C++ Primer ( 中文版, 侯俊杰译) by Stanley B. Lippman
- 层级二：专家经验 (C++/OOP)
  - ① (More )Effective C++(中文版, 侯俊杰译), by Scott Meyers.
  - ② (More )Exceptional C++ (中文版, 侯俊杰译), by Herb Sutter
  - ③ Effective Modern C++, by Scott Meyers
- 层级三：底层机制 (C++ Object Model)
  - ① Inside the C++ Object Model (深度探索 C++ 物件模型, 侯俊杰译),by Stanley Lippman.
- 层级四：设计观念的复用 (C++/Patterns)
  - ① Design Patterns: Elements of Reusable Object Oriented Software, by Erich Gamma,Richard Helm,Ralph Johnson,and John Vlissides
  - ② Modern C++ Design: Generic Programming and Design Patterns Applied by Andrei Alexandrescu.

### 编程能力

数据结构与算法、并行编程、数据库、网络编程、图形图像与游戏

AI 之路漫漫其修远兮，吾将上下而求索