Принципы работы приложений Qt

№ урока: 3 **Курс:** Qt Framework

Средства обучения: Qt Creator

Обзор, цель и назначение урока

Описать основные принципы работы приложений Qt. Рассмотреть базовый класс QObject и его возможности. Ознакомится с особенностями классов контейнеров фреймворка Qt.

Изучив материал данного занятия, учащийся сможет:

- Понять принципы работы приложений на языке программирования C++, которые используют фреймворк Qt.
- Эффективно и правильно использовать возможности класса QObject.
- Сделать осознанный выбор между использованием STL и/или Qt контейнеров в своем приложении.

Содержание урока

1. Типа приложений Qt.
2. QObject.
3. Сигналы и слоты.
4. Многопоточность.
5. Контейнеры Qt.

Резюме

- Для обеспечения корректной работы всех возможностей Qt (обработка событий, сигналы и слоты, локализация приложения и др.) перед их использованием необходимо создать специальный объект приложения (QCoreApplication, QGuiApplication или QApplication).
- Класс QObject является базовым для большинства других классов во фреймворке Qt. Данный класс является основой для объектной модели Qt и предоставляет доступ к большинству возможностей, которые Qt привносит в C++.
- Qt позволяет создавать иерархии объектов, в которых удаление родительского объекта приведет к рекурсивному удалению дочерних объектов.
- Сигналы и слоты позволяют настроить взаимодействиями между частями системы. После подключения сигнала одного объекта к слоту другого объекта, каждая генерация сигнала первого объекта приведет к вызову метода-слота второго объекта.
- Объекты класса QObject могут быть “перенесены” в отдельный поток. При этом можно обеспечить потокобезопасное взаимодействие двух объектов, которые находятся в разных потоках, при помощи сигналов и слотов.
- Qt предоставляет набор классов-контейнеров, аналогичный контейнерам из библиотеки STL с почти идентичными интерфейсами, что позволяет совместно использовать STL и Qt контейнеры в одном приложении. Основными преимуществами контейнеров Qt является использование концепцию Implicit Sharing и потоковая безопасность.

Закрепление материала

- Создать проект консольного приложение с использование Qt.

CyberBionic systematics

CyberBionic Systematics ® 2019
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com

- Создать класс отправитель, который каждые t секунд будет генерировать сигнал с текстом txt и выводить отладочную информацию об этом в консоль (например, “Send: Hello”). Время t и сообщение txt должны указываться в конструкторе объекта.
- Создать класс приемник, который будет принимать сообщения от отправителей и выводить отладочную информацию об это в консоль (например, “Received: Hello”).
- Создать один объект приемника и несколько (для начала можно и 1) объектов отправителей и настроить соединения сигналов и слотов (используя подключение по умолчанию) так чтобы приемник получал сообщения от всех отправителей. Сообщения об получении должны приходить сразу же после сообщений об отправке.
- Добавить в приемнике tSleep секунд ожидания (например используя QThread::sleep) после получения каждого сообщения. После получения сообщения должна быть задержка, во время которой новые сообщения не будут отправлены.

Дополнительное задание

Задание
Перенести объект получателя в другой поток. Понять, как и почему изменилось поведение приложения. Попробовать разные способы подключения сигналов и слотов.

Самостоятельная деятельность учащегося

Задание 1
Исследовать возможности Qt для работы с сетью (в частности с UDP).

Задание 2
Создать приложение, которое будет получать дейтаграммы по протоколу UDP и через каждые 10 секунд выводить все полученные дейтаграммы в консоль или сообщении об отсутствии дейтаграмм если их не пришло за последние 10 секунд.

Задание 3
Изучить инструменты Qt для работы со временем и выводить полученные дейтаграммы не каждые 10 секунд, а в начале каждой новой минуты. Желательно сделать это оптимально без постоянного (каждую секунду или чаще) запроса текущего времени.

Рекомендуемые ресурсы

http://doc.qt.io/qt-5/index.html
https://www.google.com/

CyberBionic Systematics ® 2019
19 Eugene Sverstyuk Str., 5 floor
Kyiv, Ukraine

Tel. 0 800 750 312
E-mail: edu@cbsystematics.com
Site: www.edu.cbsystematics.com | itvdn.com