



Язык описания графического интерфейса QML

# Qt Framework

## Автор курса



Ларионенко Руслан

Опыт работы: 6 лет.

Языки программирования: C++, Python.

Направления: графический интерфейс пользователя и средне/высокоуровневая логика.

Проекты:

Xfinity: работа с графическим интерфейсом, интеграция веб обозревателя, работа с мультимедиа фреймворком.

Tivo: создание многих частей нового интерфейса Hydra, работа над функцией пропуска рекламы AdSkip.

BMV: бизнес логика поискового движка.

# Qt Framework

После урока обязательно



Повторите этот урок в видео формате на  
[ITVDN.com](http://itvdn.com)



Проверьте как Вы усвоили данный материал на  
[TestProvider.com](http://testprovider.com)

## Язык описания графического интерфейса QML

# Qt Framework

## План

1. Что такое QML?
2. Основные концепции QML:
  1. Компоненты.
  2. Связывания.
  3. Подключения.
3. Использование JavaScript в QML.

# Qt Framework

## Что такое QML?

- Декларативный язык программирования, предназначенный для описания графического интерфейса.
- Дополнительная логика может быть описана с помощью JavaScript.
- Библиотека Qt Quick содержит большой набор логических и визуальных компонентов QML.
- Есть возможность взаимодействия с C++ объектами Qt с помощью сигналов, слотов, свойств и методов, помеченных как Q\_INVOKABLE. Для этого объект или класс C++ должен быть зарегистрирован в QML.

<https://doc.qt.io/qt-5/qtqml-index.html>

# Qt Framework

## Основные концепции QML

- Компоненты.
- Связывания.
- Подключения.

<https://doc.qt.io/qt-5/qtqml-index.html>

# Qt Framework

## Компоненты

- Компоненты - строительные блоки приложений на языке QML.
- Они могут быть как исключительно логическими так и визуальными.
- Компоненты имеют свойства которые могут влиять на поведение компонента.
- В компонент можно добавить новые свойства и сигналы или сделать ссылки на существующие.
- Компоненты могут быть импортированы в другие файлы.
- Компоненты имеют свойство `id`, которое может быть использовано для посылания на конкретный компонент из QML.
- Свойство `objectName`, может быть использовано для нахождения QML объекта из C++ кода.

<https://doc.qt.io/qt-5/qml-qtqml-component.html>



# Qt Framework

## Связывание

- Связывание позволяет "связать" свойство компонента с некоторым другим свойством или выражением. Выражение может использовать условные операторы или быть функцией.
- Связывание осуществляется при помощи оператора : или компонента Binding.
- Свойство по левую сторону оператора : будет обновляться каждый раз при изменении свойства или выражения по правую сторону оператора :

<https://doc.qt.io/qt-5/qml-qtqml-binding.html>

# Qt Framework

## Подключения

- Подключения позволяют подключать сигналы к слотам, выражениям или функциям.
- Подключения осуществляются при помощи оператора : или компонента Connections.
- Список аргументов явно не описывается при создании подключения но все аргументы доступны в пространстве обработчика по их именам.

<https://doc.qt.io/qt-5/qml-qtqml-connections.html>

# Qt Framework

## Использование JavaScript в QML

- JavaScript имеет доступ ко всем компонентам и их свойствам в приложении.
- В QML можно использовать JavaScript код в качестве выражений к которым привязываются свойства или в качестве обработчиков сигналов в подключениях.
- В QML коде можно создавать JavaScript функции для дальнейшего их использования.
- JavaScript код можно перенести в отдельный файл и импортировать в QML.
- Не следует злоупотреблять JavaScript если необходимо обеспечить высокую производительность приложения.

<https://doc.qt.io/qt-5/qml-qtqml-connections.html>

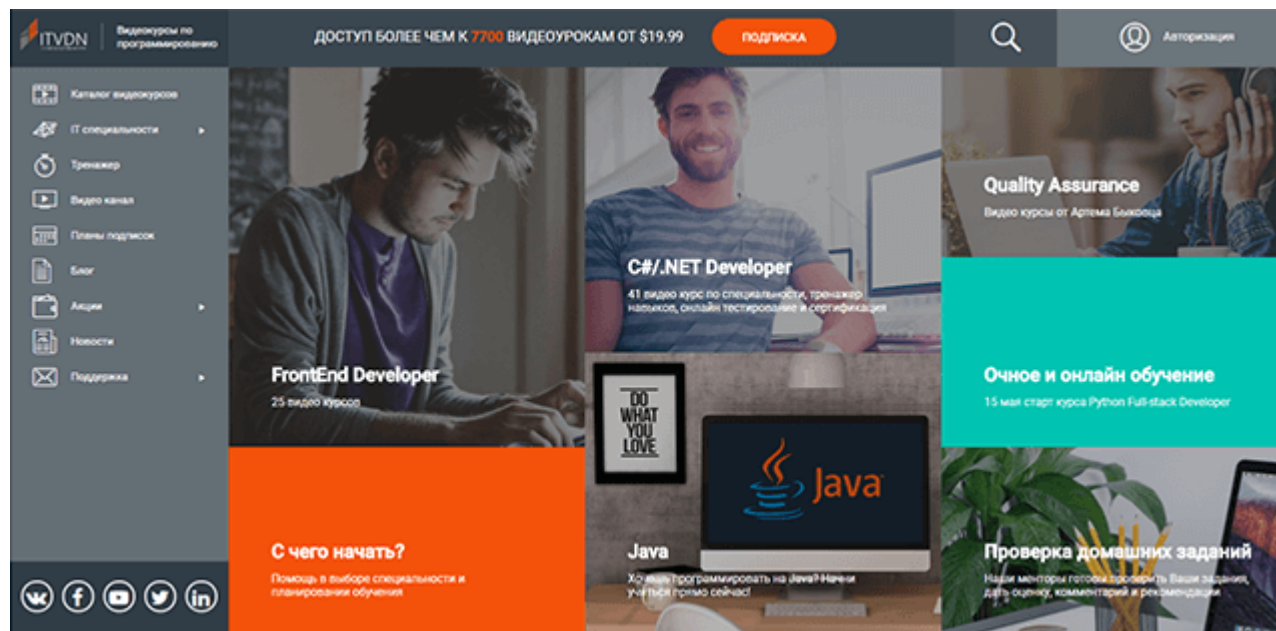
# Qt Framework

## Выводы

- QML предоставляет удобный синтаксис для описания графического интерфейса и логики его работы с использованием компонентов и языка программирования JavaScript.
- QML также работает с концепцией сигналов и слотов, которая используется в C++ Qt Widgets.
- Концепция связывания позволяет удобно описать зависимость свойств объектов друг от друга.

# Смотрите наши уроки в видео формате

ITVDN.com



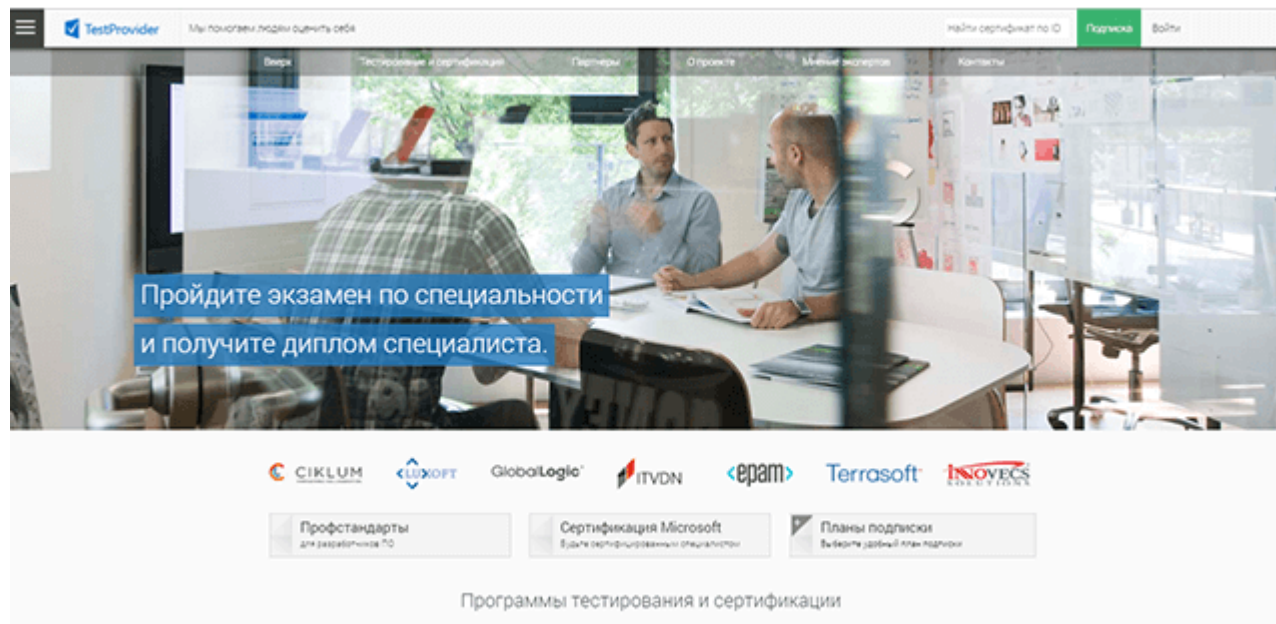
Посмотрите этот урок в видео формате на образовательном портале [ITVDN.com](http://ITVDN.com) для закрепления пройденного материала.

Курсы записаны сертифицированными тренерами, которые работают в учебном центре CyberBionic Systematics и другими высококвалифицированными разработчиками.



# Проверка знаний

TestProvider.com



TestProvider – это online сервис проверки знаний по информационным технологиям. С его помощью Вы можете оценить Ваш уровень и выявить слабые места. Он будет полезен как в процессе изучения технологии, так и для общей оценки знаний IT специалиста.

После каждого урока проходите тестирование для проверки знаний на [TestProvider.com](https://testprovider.com)

Успешное прохождение финального тестирования позволит Вам получить соответствующий Сертификат.



Q&A

# Информационный видеосервис для разработчиков программного обеспечения

