



FORMATION "INTRODUCTION À QT WIDGETS"

PROGRAMME



Formateurs

La société KDAB possède une expérience de plus de 10 années dans le développement d'applications utilisant la bibliothèque Qt, ainsi que dans les formations à ce type de développement.

La filiale française *KDAB (France)* propose des services de développement, de conseil, d'assistance et de formation, autour de Qt.

La déclaration d'activité de formation de KDAB (France) a été enregistrée sous le **numéro 93 84 03221 84** auprès du préfet de région de Provence-Alpes-Côte d'Azur.

Le Directeur Général de KDAB France, David Faure, ingénieur en informatique de l'INSA de Lyon, est développeur KDE (bureau de logiciel libre basé sur Qt) depuis 1998, et a participé au sein de KDAB à de nombreux projets de développements basés sur Qt, depuis 2003. Il a aussi effectué de nombreuses formations Qt dans toute l'Europe. Il travaille depuis les bureaux de KDAB (France) dans la banlieue d'Avignon.

Nicolas Arnaud-Cormos, ingénieur d'études et développement de l'Université de Rennes, a rejoint KDAB en 2007, il est formateur Qt, ainsi que développeur et chef de projet, en particulier pour un projet de logiciel de visualisation scientifique 2D/3D. Il est basé sur Limoges.

Franck Arrecot, titulaire d'un Master Informatique et Développement Logiciel de l'Université de Toulouse utilise le framework Qt depuis 2011. Il a rejoint KDAB en 2014 en tant que développeur et formateur spécialisé sur les technologies QtWidgets et QtQuick. Il intervient régulièrement dans des conférences pour y parler techniques de développement ou faire découvrir le travail de la communauté KDE. Il est basé sur Bayonne.

Mike Krus, titulaire d'un doctorat en réalité virtuelle de l'Université Paris Sud, est un développeur C++ depuis de nombreuses années, et utilise Qt sur plusieurs plateformes, y compris mobiles, depuis plus de 12 ans. Il a rejoint KDAB en 2015 où il est également formateur Qt et OpenGL. Il participe au développement de Qt3D et est responsable du port de Qt pour tvOS. Il est basé en Ecosse.

Durée

En cas de formation sur site, la formation "Introduction à Qt Widgets" dure de 3 jours (21 heures) à 5 jours (35 heures), au choix du client.

En cas de formation inter-entreprises, la formation dure 3 jours (21 heures).

Objectif

Cette formation enseigne les techniques de développement d'applications graphiques, réseau et/ou multitâches, en utilisant la technologie objet Qt/C++.

A l'issue de la formation, les participants seront capables d'appliquer leur nouvelle compréhension des principes de Qt et leur nouveau savoir-faire pour le développement d'applications basées sur Qt.

Audience

Cette formation est destinée aux programmeurs d'applications et systèmes ayant à connaître la



programmation Qt.

Prérequis

Une expérience professionnelle de la programmation objet est indispensable, et une expérience en C++ est fortement recommandée. Cependant si le client en exprime le besoin, la première demi-journée de formation pourra être consacrée à des rappels de C++, afin de s'assurer que les notions nécessaires à la suite de la formation sont bien connues par les participants.

Compétences professionnelles visées

- Compiler et exécuter des applications Qt avec qmake et QtCreator
- Utiliser les signaux et les slots pour faire communiquer des objets entre eux
- Développer une interface graphique à l'aide des widgets Qt, en C++
- Développer une interface graphique à l'aide des widgets Qt, en utilisant Qt Designer
- Développer un widget avec un rendu personnalisé
- Ecrire un modèle pour fournir des données à une vue de type liste, table ou arborescente
- Personnaliser le rendu et l'édition des items dans une telle vue, au moyen du delegate
- Utiliser les classes fournies par Qt, en particulier QtCore, en suivant les bonnes pratiques établies par la communauté de développeurs Qt au fil des années

Thèmes abordés

1. Rappels de C++ (si demandé par une majorité de participants)
 - Qu'est-ce que C++
 - Une pile en C
 - Une pile en C++
 - Constructions de bases
 - Héritage
 - Pointeurs et références
 - Surcharge d'opérateurs
 - Templates
 - Exceptions
 - Différences entre Java et C++
 - Références de lecture pour approfondir
2. Les bases de Qt
 - L'historique de Qt
 - Développer une application Hello World
 - Les classes d'application



- Introduction à Qt Creator
- Conseils pratiques pour développeurs
- 3. L'intégration dans Visual Studio
- 4. Les objets Qt
 - Fonctionnalités communes aux objets Qt
 - Le système de méta-objet Qt
 - Communication entre objets à base de signaux et slots
 - Définir des méthodes invocables
 - Gérer des évènements avec Qt
- 5. Widgets
 - Widgets courants
 - Gestion des layouts
 - Conseils pour les widgets personnalisés
- 6. Qt Designer
- 7. Fenêtres et boîtes de dialogue
 - Boîtes de dialogue, principes généraux
 - Boîtes de dialogue fournies par Qt
 - Fenêtre principale
- 8. Classes QtCore
 - Gestion des chaînes de caractères
 - Classes conteneur
 - Gestion de fichiers
 - QVariant et système de propriétés
- 9. Dessiner un widget
 - Les bases du dessin
 - Gestion des couleurs
 - Opérations de dessin
 - Considérations de performances
- 10. Programmation Modèle/Vue
 - Concepts de la programmation Modèle/Vue
 - Modèles personnalisés
 - Vues
 - Delegates
 - Sélection

Les formations de plus de 3 jours comportent aussi un nombre variable de sujets supplémentaires, que le client (ou les participants, à la majorité, pour les formations inter-entreprises) devra choisir parmi les sujets suivants:

- 11. Infrastructure d'applications
 - Système d'évènements
 - Impression
 - Traduction (internationalisation)
 - Ressources
 - Configuration (Settings)
 - Fichiers spécifiques à la plate-forme, au périphérique, à la locale ou à la résolution
- 12. Développement et tests
 - Déploiement
 - Outils de développement sous Linux



- Aides au débogage
 - Tests unitaires avec QTestLib
 - Tests automatisés avec Squish
 - Introduction à Git
 - Introduction à CMake
13. XML, JSON et SQL
- Utiliser XML dans Qt
 - JSON
 - Bases de données SQL
14. Communication inter-processus
- Sockets réseau (QtNetwork)
 - QtDBus
 - QProcess

Pour une formation sur site de 3 jours il est possible de personnaliser le programme en échangeant certains sujets principaux par des sujets supplémentaires.

Moyens pédagogiques

Le support de formation comporte plus de 1200 pages en anglais, et est régulièrement mis à jour pour suivre les évolutions de Qt. Le support de formation est projeté sur écran au moyen d'un vidéo-projecteur. Une copie imprimée du support de formation est remise à chaque participant.

Un grand nombre de ces chapitres est assorti d'un exercice de travaux pratiques qui permettra aux participants de rendre les notions abordées plus concrètes. Ces exercices seront effectués sur ordinateur (au minimum un ordinateur pour deux participants).

Une clé USB sera remise à chaque participant, avec les points de départ pour les travaux pratiques, leurs corrigés, et le code des exemples présentés pendant la formation.

Résultats

A l'issue de la formation, les participants seront capables d'appliquer leur nouvelle compréhension des principes de Qt et leur nouveau savoir-faire pour le développement d'applications basées sur Qt.

Évaluation

Une évaluation écrite faite dans la dernière demi-journée permettra d'établir le niveau en développement Qt des participants à la formation.

Après l'examen individuel, les questions et les bonnes réponses sont présentées en séance pour débattre avec les participants. Le résultat individuel de chaque participant lui est remis à l'issue de la formation.

Suivi après la formation

Il est possible de compléter la formation par une formation avancée sur QGraphicsView, sur le multitâche, ou sur Model/View (3 jours chacune). Il est aussi possible de commander une assistance au développement (conseils, débogage, ...) auprès de KDAB (France).