

Programmation Rust - Introduction

20 novembre 2022

Présentation

Cette formation a pour but de donner les bases du développement avec le langage Rust. Rust est un langage de programmation système ultra-rapide, qui prévient les erreurs de segmentation et garantit la sûreté entre threads. Il a été conçu pour être un langage sécurisé, concurrent, pratique, supportant les styles de programmation purement fonctionnel, modèle d'acteur, procédural, ainsi qu'orienté objet sous certains aspects.

Objectifs

A la fin de la formation, les stagiaires seront capables de :

- Ecrire du code idiomatique Rust.
- Compiler et tester leur code.
- Choisir les structures de données adaptées à chaque usage.
- Utiliser différentes bibliothèques du langage et la documentation.
- Identifier les points forts et les points faibles du langage Rust.
- Concevoir et implémenter leur première application Rust.

Prérequis

Connaissances de base en développement.

Environnement Unix / Linux.

Sanction

Une attestation est remise à chaque stagiaire ayant suivi la formation complète.

Public visé

Cette formation s'adresse à toute personne ou structure souhaitant se former/former leurs équipes techniques (développeurs, ingénieurs, techniciens, team leader etc.) à un langage de programmation système, performant et sûr.

Formats

Choisissez la formule adaptée à vos besoins. La formule de 4 jours propose un travail plus approfondi du projet final.

Formule 1 :

- Durée : 3 jours (21 heures)
- Tarif : 1700 € (HT) / personne

Formule 2 :

- Durée : 4 jours (28 heures)
- Tarif : 2200 € (HT) / personne

Modalités et délai d'accès, déroulement

Cette formation peut être réalisée dans les locaux de votre entreprise ou à distance. Le délai d'accès estimé pour cette formation est de maximum un mois.

Avant la formation

Entretien ou questionnaire pour identifier les attentes et besoins.

Envoi des éléments administratifs : convention, livret d'accueil et règlement intérieur.

Envoi des ressources et des instructions pour le bon déroulement de la séance.

Pendant la formation

Le stagiaire se présente à la formation avec son ordinateur personnel.

Évaluation tout au long de la formation (mise en situation, quiz, interrogation).

A la fin de la formation

Remise d'une attestation de formation.

Questionnaire d'auto-évaluation.

Questionnaire de satisfaction permettant d'évaluer la formation.

Après la formation

Questionnaire de satisfaction à froid (à 3 mois de la formation).

Accessibilité, inclusion

Pour toutes nos formations, nous réalisons des études préalables à la formation pour adapter les locaux, les modalités pédagogiques et l'animation de la formation en fonction de la situation de handicap annoncée.

Nous sommes à votre écoute pour toute question ou besoin spécifique, contactez-nous : formation@cenotelie.fr

Méthodes mobilisées, évaluation

Cette formation s'effectue, pour chaque module, à 60% par des exercices pratiques et à 40% par des présentations. Plusieurs modalités pédagogiques sont mises en place en fonction du niveau des participants et de la taille du groupe : exercices pratiques, discussions guidées, codéveloppement, coaching individuel etc. La validation des acquis et l'évaluation est faite par le formateur tout au long de l'apprentissage. Un ou plusieurs projets à réaliser en autonomie sont proposés dans l'objectif de consolider les connaissances apprises.

Indicateurs de résultat

Qualité du formateur : 4.9/5

Taux de satisfaction : 4.8/5

Programme

Module 1 – Introduction

Positionnement du langage

- Vitesse d'exécution
- Gestion des erreurs
- Sécurité
- Multi-threading

Paradigmes de programmation

Module 2 – Programmation de base

Types primitifs

Fonctions

Variables et Mutabilité

Control flow

Types structurés

Type énumérés (enum, match, Option)

Module 3 – Ownership et références

Ownership

Référence et Emprunt

Slices

Durée de vie (lifetime)

Types contenant des références

Module 4 – Types avancés

Collections (String, Vec, HashMap)

Iterateurs

Gestion d'erreur (Result, propagation, panic)

Fermetures (Closures)

Module 5 – Généricité

Types génériques

Système de Trait

Traits standards

- Surcharge d'opérateurs

Smart pointers (Box, Rc, Cell, RefCell)

Module 6 – Environnement de développement

Installation (rustup, paquets système)

Compilation et Système de build (rustc, cargo)

Gestionnaire de paquets (cargo et crates.io)

Tests unitaires

Documentation : rustdoc

Environnements / Editeurs

Module 7 – Multi-threading

Communication entre threads

Partage de données

Trait Send et Sync

Module 8 – Programmation Rust avancée

Macros

Utiliser C/C++ depuis Rust

Utiliser Rust depuis C/C++

Unsafe Rust

Module 9 – Projet final

Formule 3 jours : Projet récapitulatif sur 1/2 journée

Formule 4 jours : Projet récapitulatif sur 1 journée.

Le formateur

Laurent Wouters est co-fondateur de Cénotélie et un chercheur et formateur dédié. Laurent pratique ce langage au quotidien depuis plusieurs années, notamment dans le cadre du développement d'une plateforme en ligne pour le suivi de la croissance des cultures agricoles sur la base de l'analyse d'images satellite. Il s'agit d'un produit industriel complexe, dont tous les composants côté serveur sont écrits en Rust. Laurent détient un titre de doctorat en Théorie des langages. Par ailleurs, il a participé à l'encadrement de plusieurs travaux de stage et de thèse. Avant de devenir un formateur professionnel, il s'est forgé une expérience en tant qu'enseignant dans plusieurs grandes universités parisiennes, dont l'Ecole Centrale de Paris.

Contactez le formateur : formation@cenotelie.fr