

Mis à jour le 12/01/2026

S'inscrire

## Formation JetBrains RubyMine

3 jours (21 heures)

### Présentation

JetBrains RubyMine est un IDE complet pour accélérer le développement Ruby et Ruby on Rails grâce à l'autocomplétion, la navigation intelligente et les outils de refactoring. La formation cible des cas d'usage concrets : prise en main d'un projet existant, amélioration de la qualité, et productivité au quotidien.

Cette formation vous apprend à configurer RubyMine pour votre stack (Ruby, Rails, Bundler) et à exploiter ses fonctionnalités clés : inspection de code, recherche avancée, exécution et débogage, tests, et intégration Git. L'objectif est de réduire le temps passé à diagnostiquer et à maintenir, tout en fiabilisant vos livraisons.

L'approche est résolument pratique : ateliers guidés sur une application Rails, démos de refactorings, et exercices de debugging. Vous repartez avec une configuration IDE reproductible, des raccourcis et workflows efficaces, ainsi qu'une check-list d'outillage (tests, linters, run configurations) applicable à vos projets.

### Objectifs

- Configurer RubyMine pour Ruby/Rails, Bundler et les SDK.
- Naviguer efficacement dans une base de code (symboles, usages, recherches).
- Déboguer une application et analyser les erreurs d'exécution.
- Exécuter et industrialiser les tests (RSpec/Minitest) depuis l'IDE.
- Appliquer des refactorings sûrs et exploiter les inspections de code.

### Public visé

- Développeurs Ruby / Ruby on Rails.
- Développeurs full-stack intervenant sur une codebase Rails.
- Tech leads souhaitant standardiser les pratiques IDE.

### Pré-requis

- Connaissances de base en Ruby.
- Notions de Ruby on Rails (MVC, routes, console).
- Familiarité avec Git et le terminal.
- Compréhension des tests automatisés (principes).

## Pré-requis techniques

- RubyMine installé (version récente) et licence/accès activé.
- OS : macOS, Linux ou Windows (WSL2 recommandé).
- RAM : 8 Go minimum, 16 Go recommandés.
- Ruby + Bundler installés, et un projet Rails de travail (fourni ou interne).
- Accès Internet pour récupérer dépendances (gems) et plugins.

## Programme de formation JetBrains RubyMine

[Jour 1 - Matin]

### Prise en main de RubyMine et configuration de l'environnement Ruby

- Installer RubyMine et configurer le Ruby SDK (rbenv, RVM, asdf) et Bundler
- Créer/importer un projet : structure, Gemfile, gestion des dépendances
- Paramétriser l'éditeur : inspections, formatage, templates de fichiers, raccourcis essentiels
- Configurer l'exécution : Run/Debug configurations, variables d'environnement, arguments
- Atelier pratique : Initialiser un projet Ruby, configurer le SDK et exécuter un script avec Bundler.

[Jour 1 - Après-midi]

### Productivité au quotidien : navigation, édition intelligente et refactoring

- Navigation rapide : classes, méthodes, fichiers, symboles, recherche structurée
- Édition assistée : complétion, intentions, génération de code, live templates
- Refactorings Ruby : rename, extract method/variable, inline, safe delete
- Qualité : inspections, quick-fixes, conventions Ruby, gestion des warnings
- Atelier pratique : Refactorer un module Ruby existant en améliorant lisibilité et conventions sans casser les tests.

[Jour 2 - Matin]

### Débogage Ruby et analyse d'exécution

- Débogueur RubyMine : breakpoints, step into/over/out, evaluate expression
- Inspection de l'état : variables, watches, call stack, frames et scopes
- Débogage de scripts Bundler et tâches Rake : configuration et pièges courants
- Outils d'analyse : logs, console Ruby, gestion des exceptions et backtraces
- Atelier pratique : Diagnostiquer et corriger un bug (nil/NoMethodError) via breakpoints et évaluation d'expressions.

[Jour 2 - Après-midi]

## Tests automatisés et outillage qualité (RSpec/Minitest)

- Configurer et exécuter RSpec ou Minitest depuis RubyMine (run, rerun, focus)
- Debug des tests : breakpoints dans specs/tests, isolation d'un scénario
- Couverture et feedback : lecture des résultats, tests lents, organisation des suites
- Linting/formatage : RuboCop, auto-corrections, intégration aux inspections
- Atelier pratique : Mettre en place une suite RSpec, ajouter RuboCop et corriger les violations via quick-fixes.

[Jour 3 - Matin]

## Ruby on Rails dans RubyMine : exécution, console et productivité

- Configurer un projet Rails : SDK, Bundler, bases de données, variables d'environnement
- Lancer serveur, console et tâches : rails server/console, rake/rails tasks, runners
- Navigation Rails : routes, controllers, views, models, associations, helpers
- Générateurs et templates : scaffolding, snippets, conventions et structure
- Atelier pratique : Créer une fonctionnalité Rails (resource + validations) et naviguer efficacement entre routes, contrôleur et vues.

[Jour 3 - Après-midi]

## Intégration Git, revue de code et optimisation du flux de travail

- Git dans RubyMine : commit, amend, stash, rebase interactif, résolution de conflits
- Outils de comparaison : diff, historique, annotate/blame, recherche dans l'historique
- Revue de code locale : inspections, TODO, bookmarks, checklist de qualité avant PR
- Personnalisation avancée : keymap, plugins utiles, réglages projet vs IDE
- Atelier pratique : Réaliser une mini-revue de code, corriger un conflit Git et préparer un commit propre (message, scope, lint/tests).

## Sociétés concernées

Cette formation s'adresse à la fois aux particuliers ainsi qu'aux entreprises, petites ou grandes,

souhaitant former ses équipes à une nouvelle technologie informatique avancée ou bien à acquérir des connaissances métiers spécifiques ou des méthodes modernes.

## Positionnement à l'entrée en formation

Le positionnement à l'entrée en formation respecte les critères qualité Qualiopi. Dès son inscription définitive, l'apprenant reçoit un questionnaire d'auto-évaluation nous permettant d'apprécier son niveau estimé sur différents types de technologies, ses attentes et objectifs personnels quant à la formation à venir, dans les limites imposées par le format sélectionné. Ce questionnaire nous permet également d'anticiper certaines difficultés de connexion ou de sécurité interne en entreprise (intraentreprise ou classe virtuelle) qui pourraient être problématiques pour le suivi et le bon déroulement de la session de formation.

## Méthodes pédagogiques

Stage Pratique : 60% Pratique, 40% Théorie. Support de la formation distribué au format numérique à tous les participants.

## Organisation

Le cours alterne les apports théoriques du formateur soutenus par des exemples et des séances de réflexions, et de travail en groupe.

## Validation

À la fin de la session, un questionnaire à choix multiples permet de vérifier l'acquisition correcte des compétences.

## Sanction

Une attestation sera remise à chaque stagiaire qui aura suivi la totalité de la formation.