

Mis à jour le 12/05/2026

S'inscrire

Formation Lean : Assistant de preuve

3 jours (21 heures)

Présentation

Lean est un assistant de preuve et un langage de programmation permettant de formaliser des mathématiques et de vérifier des propriétés logiques avec une grande fiabilité. Il s'utilise pour sécuriser des démonstrations, documenter des invariants et prototyper des modèles formels.

Cette formation vise à rendre Lean opérationnel pour écrire des preuves lisibles, réutilisables et maintenables. Vous apprendrez à structurer un développement, à exploiter la bibliothèque standard et à transformer des objectifs en scripts de preuve robustes.

L'approche est résolument pratique : ateliers guidés, démos pas à pas, exercices de formalisation (logique propositionnelle, quantificateurs, structures algébriques simples). Les livrables incluent un projet Lean versionné, une collection de lemmes réutilisables et une checklist de bonnes pratiques (tactiques, simplification, réécriture, organisation des fichiers).

Comme toutes nos formations, celle-ci vous présentera **la dernière version stable** de la technologie et ses nouveautés.

Objectifs

- Installer et configurer un environnement Lean reproductible.
- Écrire des preuves avec tactiques et style déclaratif.
- Manipuler égalités, réécritures et simplifications efficacement.
- Structurer des fichiers, namespaces et dépendances de projet.
- Déboguer des erreurs de typage et d'inférence de preuves.

Public visé

- Développeurs souhaitant introduire des garanties formelles.

- Ingénieurs R&D et data scientists manipulant des modèles critiques.
- Étudiants/chercheurs en mathématiques ou informatique théorique.

Pré-requis

- Logique de base (implication, quantificateurs, raisonnement par cas).
- Notions de programmation fonctionnelle (types, fonctions, immutabilité).
- Confort avec la ligne de commande.
- Pratique minimale de Git.

Pré-requis techniques

- 8 Go de RAM minimum (16 Go recommandé).
- Linux, macOS ou Windows avec WSL2.
- Lean 4 et l'outil de build associé (via gestionnaire de versions Lean).
- Éditeur de code avec support Lean (extension dédiée) et terminal.

Programme de notre formation Lean

[Jour 1 - Matin]

Prise en main de Lean et du workflow de preuve

- Installer Lean 4 et l'environnement (VS Code, extension Lean, lake)
- Comprendre la boucle : objectifs, tactiques, messages d'erreur, recherche
- Manipuler les bases : Prop, Type, variables, implicites, notations
- Écrire des preuves simples : by, intro, exact, apply
- Atelier pratique : Configurer un projet Lean et prouver des implications élémentaires.

[Jour 1 - après-midi]

Logique propositionnelle : connecteurs et tactiques essentielles

- Conjonction/disjonction : constructor, cases, left/right
- Négation et contradiction : by_contra, contradiction, exfalso
- Égalité : rfl, simp, réécriture avec rw
- Structurer une preuve : have, let, sous-preuves et lisibilité
- Atelier pratique : Prouver une série de lemmes de logique (de Morgan, contraposée) avec simp et rw.

[Jour 2 - Matin]

Quantificateurs et preuves sur les types de base

- Quantificateurs : ? (intro), ? (exists), extraction par cases
- Fonctions et lambda : preuves sur des fonctions, extensionnalité (funext)
- Raisonnement sur Nat : calcul, inégalités simples, normalisation par simp
- Utiliser la bibliothèque : trouver un lemme, naviguer dans Mathlib, #check, #find
- Atelier pratique : Formaliser des énoncés avec ?? et prouver des propriétés simples sur Nat.

[Jour 2 - Après-midi]

Induction et preuves structurées

- Induction sur Nat : induction, cas base, hypothèse d'induction
- Récursion et définitions : écrire des fonctions et prouver leurs propriétés
- Chaînage de réécritures : calc, ring (si disponible), simplifications ciblées
- Gestion des objectifs : simp?, aesop (si activé), stratégies de débogage
- Atelier pratique : Prouver par induction des propriétés classiques (somme, double, associativité sur Nat).

[Jour 3 - Matin]

Structures, types inductifs et preuves par cas

- Types inductifs : définition, constructeurs, élimination par cases
- Listes : nil/cons, preuves par induction sur List
- Structures et records : champs, projections, égalité de structures
- Automatisation raisonnable : règles simp, attributs, lemmes auxiliaires
- Atelier pratique : Définir une fonction sur listes et prouver une propriété (longueur, append, map).

[Jour 3 - Après-midi]

Mini-projet : formaliser un énoncé et produire une preuve robuste

- Choisir un énoncé réaliste (logique + induction) et le découper en lemmes
- Écrire une preuve maintenable : noms, sections, commentaires, factorisation
- Stabiliser avec simp/rw : éviter les boucles et les réécritures fragiles
- Revue et amélioration : lisibilité, performance, dépendances minimales
- Atelier pratique : Réaliser un mini-projet complet (spécification, lemmes, preuve finale) et le présenter.

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.