

Mis à jour le 04/02/2026

S'inscrire

# Formation JetBrains DataSpell

3 jours (21 heures)

## Présentation

JetBrains DataSpell est un IDE dédié à la data science qui accélère l'exploration, l'analyse et la mise en production de notebooks et scripts Python. La formation vous aide à structurer vos workflows, fiabiliser vos environnements et gagner du temps sur le debug, la visualisation et l'automatisation.

Cette formation vise à rendre vos projets data plus reproductibles grâce à une maîtrise opérationnelle de DataSpell : gestion de projets, environnements Python, exécution de notebooks, inspection de données, intégration Git et qualité de code. Vous apprendrez à passer d'une exploration rapide à un code maintenable, prêt pour le partage en équipe.

L'approche est 100% pratique : ateliers guidés, démos pas à pas, exercices de refactoring et de diagnostic (erreurs d'environnement, dépendances, performances).

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

## Objectifs

- Configurer un projet DataSpell et organiser un workspace data.
- Créer et gérer des environnements Python (venv/Conda) et dépendances.
- Exécuter, déboguer et profiler notebooks et scripts.
- Explorer des jeux de données et produire des visualisations exploitables.
- Versionner, partager et fiabiliser le travail avec Git et inspections.

## Public visé

- Data analysts et data scientists.

- Développeurs Python orientés data.
- Ingénieurs ML/IA et profils R&D.

## Pré-requis

- Notions solides de Python (variables, fonctions, modules).
- Connaissances de base en manipulation de données (pandas recommandé).
- Compréhension des environnements et dépendances Python.
- Notions de Git (commit, branche, merge).

## Pré-requis techniques

- PC/Mac avec 8 Go RAM minimum (16 Go recommandé).
- Windows 10/11, macOS ou Linux.
- JetBrains DataSpell installé (version stable).
- Python 3.10+ et accès à un gestionnaire d'environnements (venv/Conda).

## Programme de formation JetBrains DataSpell

[Jour 1 - Matin]

### Prise en main de DataSpell et configuration d'un environnement Data Science

- Tour d'horizon de l'IDE : projets, fenêtres d'outils, navigation et recherche
- Création d'un projet Python : structure, interpréteur, gestion des dépendances
- Configuration des environnements : venv/Conda, sélection du kernel, gestion des packages
- Paramétrage productivité : thèmes, keymap, inspections, formatage et linting
- Atelier pratique : Créer un projet DataSpell avec un environnement Conda et exécuter un script de validation.

[Jour 1 - Après-midi]

### Notebooks dans DataSpell : exécution, exploration et visualisation

- Création et gestion de notebooks : cellules, outputs, redémarrage kernel, variables
- Exploration de données : import CSV/Parquet, aperçu, filtrage et statistiques rapides
- Visualisations : Matplotlib/Seaborn, rendu inline, sauvegarde et export des résultats
- Qualité et reproductibilité : ordre d'exécution, seeds, gestion des dépendances du notebook
- Atelier pratique : Analyser un dataset (pandas) et produire un mini-rapport avec 3 graphiques.

[Jour 2 - Matin]

## Débogage, tests et refactoring orientés Data

- Débogage Python : breakpoints, step-by-step, inspection des variables et watch
- Profiling et performance : identification des goulots (pandas), optimisations simples
- Refactoring assisté : extraction de fonctions, renommage sûr, navigation vers usages
- Tests unitaires : configuration pytest, fixtures, exécution ciblée et rapports
- Atelier pratique : Déboguer une pipeline de préparation de données et ajouter 3 tests pytest.

[Jour 2 - Après-midi]

## Intégration Git et collaboration : workflow Data Science

- Git dans l'IDE : clone, commit, branches, rebase/merge et résolution de conflits
- Bonnes pratiques notebooks : nettoyage des outputs, gestion des diffs et versioning
- Gestion des dépendances : requirements.txt/conda.yml, verrouillage et reproductibilité
- Revue et partage : changelists, annotations, historique et comparaison de versions
- Atelier pratique : Mettre en place un workflow Git (feature branch) et livrer une PR avec notebook nettoyé.

[Jour 3 - Matin]

## SQL et accès aux données : requêtes, connexions et analyse

- Connexion aux sources : PostgreSQL/MySQL/SQLite, drivers, paramètres et sécurité
- Éditeur SQL : autocomplétion, exécution par sélection, résultats et export
- Modélisation rapide : exploration de schémas, clés, jointures et agrégations
- Allers-retours SQL & Python : chargement vers pandas, validation et contrôles
- Atelier pratique : Interroger une base, construire une vue d'analyse et charger le résultat dans un notebook.

[Jour 3 - Après-midi]

## Industrialisation légère : exécution, packaging et livrables

- Structurer une pipeline : scripts, modules, paramètres, logs et gestion des erreurs
- Exécutions reproductibles : run configurations, variables d'environnement, fichiers .env
- Packaging minimal : organisation du code, dépendances, entypoints et documentation
- Livrables : export notebook (HTML), génération de rapports et checklist de livraison
- Atelier pratique : Transformer un notebook en script paramétrable et produire un livrable HTML reproductible.

## 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.