

Updated on 01/09/2026

[Register](#)

Jetbrains DataSpell Training

3 days (21 hours)

Overview

JetBrains DataSpell is an IDE dedicated to data science that accelerates the exploration, analysis, and production of Python notebooks and scripts. The training helps you structure your workflows, make your environments more reliable, and save time on debugging, visualization, and automation.

This training aims to make your data projects more reproducible through operational mastery of DataSpell: project management, Python environments, notebook execution, data inspection, Git integration, and code quality. You will learn how to move from rapid exploration to maintainable code that is ready to be shared with your team.

The approach is 100% practical: guided workshops, step-by-step demos, refactoring and diagnostic exercises (environment errors, dependencies, performance).

Objectives

- Set up a DataSpell project and organize a data workspace.
- Create and manage Python environments (venv/Conda) and dependencies.
- Run, debug, and profile notebooks and scripts.
- Explore datasets and produce usable visualizations.
- Version control, share, and ensure reliability of work with Git and inspections.

Target audience

- Data analysts and data scientists.
- Data-oriented Python developers.
- ML/AI engineers and R&D profiles.

Prerequisites

- Solid understanding of Python (variables, functions, modules).
- Basic knowledge of data manipulation (pandas recommended).
- Understanding of Python environments and dependencies.
- Basic knowledge of Git (commit, branch, merge).

Technical requirements

- PC/Mac with at least 8 GB RAM (16 GB recommended).
- Windows 10/11, macOS, or Linux.
- JetBrains DataSpell installed (stable version).
- Python 3.10+ and access to an environment manager (venv/Conda).

Jetbrains DataSpell training program

[Day 1 - Morning]

Getting started with DataSpell and setting up a Data Science environment

- Overview of the IDE: projects, tool windows, navigation, and search
- Creating a Python project: structure, interpreter, dependency management
- Configuring environments: venv/Conda, kernel selection, package management
- Productivity settings: themes, keymap, inspections, formatting, and linting
- Hands-on workshop: Create a DataSpell project with a Conda environment and run a validation script.

[Day 1 - Afternoon]

Notebooks in DataSpell: execution, exploration, and visualization

- Creating and managing notebooks: cells, outputs, kernel restart, variables
- Data exploration: CSV/Parquet import, preview, filtering, and quick statistics
- Visualizations: Matplotlib/Seaborn, inline rendering, saving and exporting results
- Quality and reproducibility: execution order, seeds, notebook dependency management
- Hands-on workshop: Analyze a dataset (pandas) and produce a mini-report with 3 graphs.

[Day 2 - Morning]

Data-oriented debugging, testing, and refactoring

- Python debugging: breakpoints, step-by-step, variable inspection, and watch
- Profiling and performance: identifying bottlenecks (pandas), simple optimizations
- Assisted refactoring: function extraction, safe renaming, navigation to uses
- Unit testing: pytest configuration, fixtures, targeted execution, and reports
- Hands-on workshop: Debugging a data preparation pipeline and adding 3 pytest tests.

[Day 2 - Afternoon]

Git integration and collaboration: Data Science workflow

- Git in the IDE: cloning, committing, branching, rebasing/merging, and conflict resolution
- Notebook best practices: cleaning outputs, managing diffs, and versioning
- Dependency management: requirements.txt/conda.yml, locking, and reproducibility
- Review and sharing: changelists, annotations, history, and version comparison
- Hands-on workshop: Setting up a Git workflow (feature branch) and delivering a PR with a cleaned-up notebook.

[Day 3 - Morning]

SQL and data access: queries, connections, and analysis

- Connecting to sources: PostgreSQL/MySQL/SQLite, drivers, settings, and security
- SQL editor: autocompletion, execution by selection, results, and export
- Rapid modeling: exploring schemas, keys, joins, and aggregations
- SQL & Python round trips: loading to pandas, validation, and checks
- Hands-on workshop: Querying a database, building an analysis view, and loading the results into a notebook.

[Day 3 - Afternoon]

Light industrialization: execution, packaging, and deliverables

- Structuring a pipeline: scripts, modules, parameters, logs, and error handling
- Reproducible executions: run configurations, environment variables, .env files
- Minimal packaging: code organization, dependencies, entry points, and documentation
- Deliverables: notebook export (HTML), report generation, and delivery checklist
- Hands-on workshop: Transform a notebook into a configurable script and produce a reproducible HTML deliverable.

Companies involved

This training is aimed at both individuals and companies, large or small, wishing to train their teams in new advanced IT technology or to

acquire specific professional knowledge or modern methods.

Placement at the start of training

The placement test at the start of the training course complies with Qualiopi quality criteria. Once they have finalized their registration, learners receive a self-assessment questionnaire that allows us to assess their estimated level of proficiency in different types of technologies, as well as their expectations and personal objectives for the upcoming training course, within the limits imposed by the selected format. This questionnaire also allows us to anticipate certain connection or internal security issues within the company (intra-company or virtual classroom) that could be problematic for the monitoring and smooth running of the training session.

Teaching methods

Practical training: 60% practical, 40% theory. Training materials distributed in digital format to all participants.

Organization

The course alternates between theoretical input from the trainer, supported by examples and reflection sessions, and group work.

Validation

At the end of the session, a multiple-choice questionnaire is used to verify that the skills have been correctly acquired.

Certification

A certificate will be issued to each trainee who has completed the entire training course.