

Environnement Python — Guide d'Installation

Adservio | Dr Olivier Vitrac

2026-02-03

Contents

1 Environnement Python — Guide d'Installation	1
1.1 Pourquoi Python ?	1
1.2 Choix de la Distribution	2
1.2.1 Anaconda vs Miniconda vs Miniforge	2
1.3 Installation de Miniforge	2
1.3.1 Windows	2
1.3.2 macOS	2
1.3.3 Linux	3
1.4 Création d'un Environnement	3
1.4.1 Principe	3
1.4.2 Commandes Essentielles	3
1.4.3 Environnement depuis un fichier	3
1.5 Environnement par Parcours	4
1.5.1 Parcours A — Testing Agent	4
1.5.2 Parcours B — RAG Code/Docs	4
1.5.3 Parcours C — MCP Server	4
1.6 Bonnes Pratiques	4
1.6.1 1. Un environnement par projet	4
1.6.2 2. Figer les versions	5
1.6.3 3. Ne jamais installer dans base	5
1.6.4 4. Utiliser pip dans conda	5
1.7 Dépannage	5
1.7.1 “conda: command not found”	5
1.7.2 Conflit de versions	5
1.7.3 Lenteur de conda	5
1.8 Vérification Finale	6
1.9 Ressources	6

1 Environnement Python — Guide d'Installation

1.1 Pourquoi Python ?

Python est le langage dominant pour l'IA et le Machine Learning. Les raisons :

Aspect	Python	Java
Écosystème IA/ML	PyTorch, TensorFlow, LangChain, etc.	Limité

Aspect	Python	Java
Prototypage	Rapide, interactif (REPL, notebooks)	Plus verbeux
Intégration LLM	APIs natives, SDKs officiels	Wrappers souvent en retard
Data Science	pandas, numpy, scipy — standards	Pas d'équivalent mature

Pour ce workshop : tous les parcours utilisent Python. Les outils MCP, RAG et agents de test sont en Python.

1.2 Choix de la Distribution

1.2.1 Anaconda vs Miniconda vs Miniforge

Distribution	Taille	Licence	Recommandation
Anaconda	~3 Go	Commerciale (>200 employés)	□ Éviter en entreprise
Miniconda	~50 Mo	Commerciale (>200 employés)	△ Vérifier licence
Miniforge	~50 Mo	BSD (libre)	□ Recommandé

Pourquoi Miniforge ? - Distribution **libre** (conda-forge par défaut) - Compatible avec tous les packages conda - Pas de restriction commerciale - Mêmes commandes que conda

1.3 Installation de Miniforge

1.3.1 Windows

```
# Télécharger depuis GitHub
# https://github.com/conda-forge/miniforge/releases

# Ou via winget
winget install --id CondaForge.Miniforge3

# Vérifier l'installation
conda --version
```

1.3.2 macOS

```
# Homebrew (recommandé)
brew install miniforge

# Ou téléchargement direct
curl -L -O "https://github.com/conda-forge/miniforge/releases/latest/download/Miniforge3-MacOSX-$(_uname -m).sh"
bash Miniforge3-MacOSX-$(_uname -m).sh
```

```
# Redémarrer le terminal puis vérifier  
conda --version
```

1.3.3 Linux

```
# Téléchargement direct  
curl -L -O "https://github.com/conda-forge/miniforge/releases/latest/download/Miniforge3-Linux-x86_64.sh  
bash Miniforge3-Linux-x86_64.sh  
  
# Suivre les instructions (accepter licence, chemin par défaut)  
# Redémarrer le terminal  
  
conda --version
```

1.4 Création d'un Environnement

1.4.1 Principe

Un **environnement** isole les dépendances d'un projet. Équivalent conceptuel : - Java : différentes versions de JDK + Maven dependencies - Node : nvm + node_modules - Python : conda env + pip packages

1.4.2 Commandes Essentielles

```
# Crée un environnement pour le workshop  
conda create -n workshop-agents python=3.11
```

```
# Activer l'environnement  
conda activate workshop-agents
```

```
# Le prompt change : (workshop-agents) $
```

```
# Installer des packages  
pip install pytest langchain faiss-cpu
```

```
# Désactiver  
conda deactivate
```

```
# Lister les environnements  
conda env list
```

```
# Supprimer un environnement  
conda env remove -n workshop-agents
```

1.4.3 Environnement depuis un fichier

Créer environment.yml :

```
name: workshop-agents  
channels:  
  - conda-forge  
  - defaults  
dependencies:
```

```
- python=3.11
- pip
- pip:
  - pytest
  - pytest-cov
  - langchain
  - faiss-cpu
  - sentence-transformers
  - mcp

# Créer depuis le fichier
conda env create -f environment.yml

# Mettre à jour
conda env update -f environment.yml
```

1.5 Environnement par Parcours

1.5.1 Parcours A — Testing Agent

```
conda create -n track-a-testing python=3.11
conda activate track-a-testing
pip install pytest pytest-cov pytest-asyncio
pip install anthropic # Claude API
```

1.5.2 Parcours B — RAG Code/Docs

```
conda create -n track-b-rag python=3.11
conda activate track-b-rag
pip install faiss-cpu sentence-transformers
pip install rank-bm25 # Recherche lexique
pip install langchain langchain-community
```

1.5.3 Parcours C — MCP Server

```
conda create -n track-c-mcp python=3.11
conda activate track-c-mcp
pip install mcp
pip install ruff # Linter moderne
pip install radon # Métriques de code
```

1.6 Bonnes Pratiques

1.6.1 1. Un environnement par projet

```
# Projet A
conda activate projet-a
# Projet B (autre terminal ou après deactivate)
conda activate projet-b
```

1.6.2 2. Figer les versions

```
# Exporter les versions exactes  
pip freeze > requirements.txt
```

```
# Réinstaller exactement  
pip install -r requirements.txt
```

1.6.3 3. Ne jamais installer dans base

```
# ☐ Mauvais  
conda activate base  
pip install quelque-chose
```

```
# ☐ Bon  
conda create -n mon-projet python=3.11  
conda activate mon-projet  
pip install quelque-chose
```

1.6.4 4. Utiliser pip dans conda

```
# Préférer pip pour les packages Python purs  
# Conda pour les dépendances système (CUDA, etc.)  
  
conda install cudatoolkit # Dépendance système  
pip install torch # Package Python
```

1.7 Dépannage

1.7.1 “conda: command not found”

```
# Réinitialiser le shell  
source ~/miniforge3/etc/profile.d/conda.sh  
conda init bash # ou zsh  
  
# Redémarrer le terminal
```

1.7.2 Conflit de versions

```
# Recréer l'environnement proprement  
conda env remove -n mon-env  
conda create -n mon-env python=3.11
```

1.7.3 Lenteur de conda

```
# Utiliser mamba (drop-in replacement plus rapide)  
conda install mamba -n base -c conda-forge
```

```
# Puis utiliser mamba au lieu de conda  
mamba create -n mon-env python=3.11  
mamba install package
```

1.8 Vérification Finale

```
# Environnement activé ?  
echo $CONDA_DEFAULT_ENV  
  
# Python correct ?  
python --version  
which python  
  
# Packages installés ?  
pip list | grep pytest
```

1.9 Ressources

- Miniforge GitHub
 - Conda Documentation
 - Conda Cheat Sheet
-

Document préparatoire — À consulter avant le workshop.