

# **Préambule**

**Tristan Crolard**

Laboratoire CEDRIC  
Equipe « Systèmes Sûrs »

**`tristan.crolard@cnam.fr`**

**`cedric.cnam.fr/sys/crolard`**

# Organisation

- ▶ **Modalité : « formation à distance planifiée »**
  - séances pré-enregistrées (cours et correction des TP)
  - séances interactives à distance planifiées (regroupements)
- ▶ **12 séances**
  - 4 séances sur la [programmation fonctionnelle](#) en Python
  - 4 séances sur JavaScript/TypeScript et React
  - 4 séances sur le projet (développé en Python)
- ▶ **Evaluation**
  - TP noté optionnel
  - projet à rendre (source et rapport)

# Modalités d'évaluation

note\_finale =

**si** note\_de\_projet > 10 **alors** note\_de\_projet

**sinon**

**si**  $(2 \times \text{note\_de\_projet} + \text{note\_de\_TP}) / 3 > 10$  **alors** 10

**sinon** note\_de\_projet

En seconde session, même formule (avec la même `note_de_TP`) et le *même sujet* mais le *barème prend en compte le temps supplémentaire*.

# Python

## Que dire de Python?

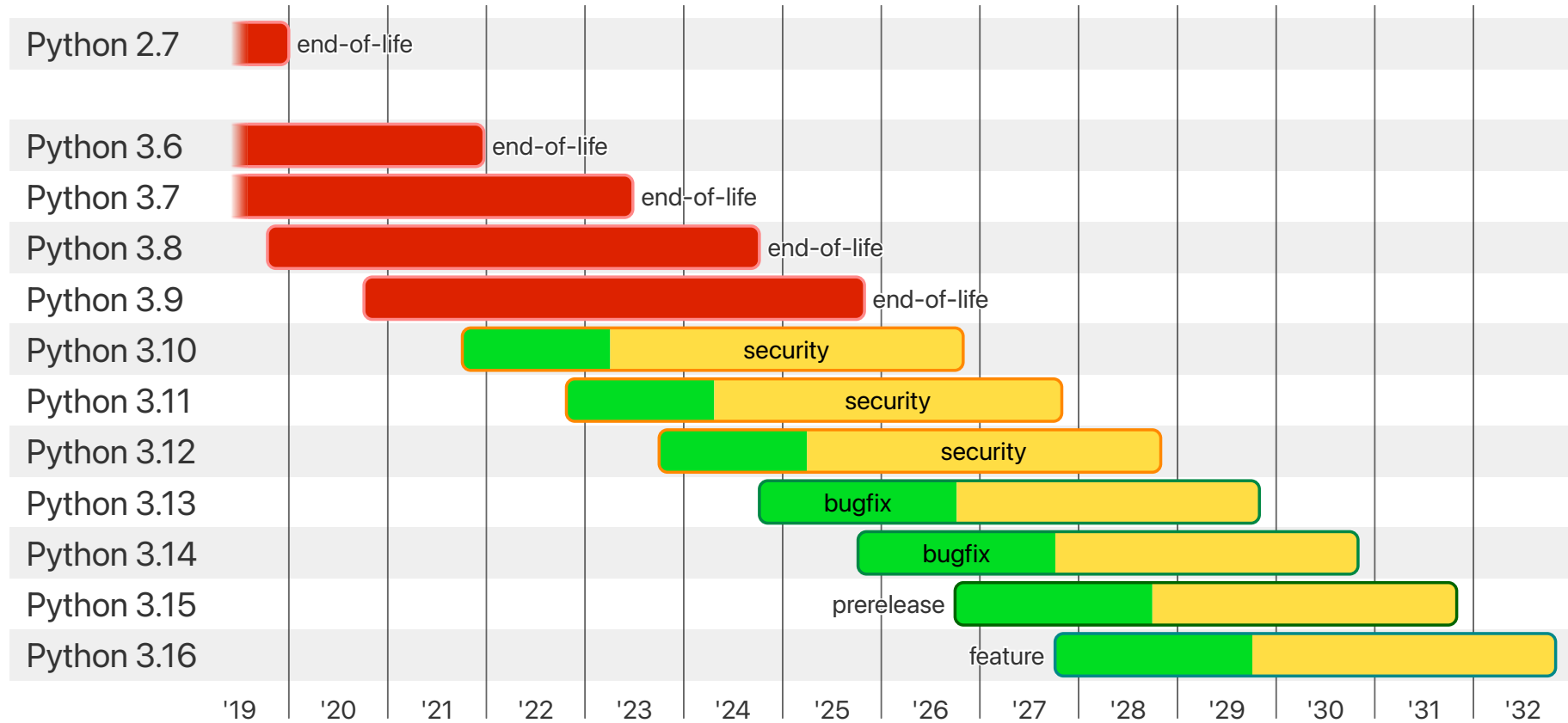
- ▶ langage de programmation généraliste de haut niveau
- ▶ compilation vers du bytecode, qui est interprété par une VM par défaut (mais des compilateurs vers du code natif existent)
- ▶ autorise les principaux paradigmes (imperative, fonctionnel, orienté objets)

## Disponibilité

- ▶ Python is gratuit (et même open source)
- ▶ Python est multi-plateformes (fonctionne sur Windows, Linux/Unix, Mac OS, ...)

# Versions de Python

<https://devguide.python.org/versions>



# Quoi de nouveau dans Python ?

- ▶ <https://peps.python.org/pep-0484>  
(Type Hints - 2014)
- ▶ <https://peps.python.org/pep-0526>  
(Syntax for Variable Annotations - 2016)
- ▶ <https://peps.python.org/pep-0589>  
(TypedDict - 2019)
- ▶ <https://docs.python.org/3/whatsnew/3.10.html>  
(PEP 634: Structural Pattern Matching)
- ▶ <https://docs.python.org/3.12/whatsnew/3.12.html>  
(PEP 695: Type Parameter Syntax)

# Python : un langage moderne ?

	Java	(Java/Type)Script	Python
records	Y (java 17) [2021]	Y	Y (dataclasses)
pattern matching	Y (java 21) [2023]	Y (partial)	Y (3.10) [2021]
exceptions	Y	Y	Y
object-oriented	Y	Y	Y
functional programming	Y (java 8) [2014]	Y	Y
static type checking	Y	N/Y (gradual)	Y (gradual) [2014]
proper generics	Y (java 5) [2009]	Y	Y (3.12) [2023]

Dans ce cours, uniquement :

- **programmation fonctionnelle** avec **pattern matching** (3.10)
- **python moderne** avec **types explicites** et **genericité** (3.12)

# Environnement Python standard

Une installation standard Python vient avec CPython:

- un interpréteur (un compilateur vers du bytecode & une VM pour l'exécuter).
- cet interpréteur peut aussi être utilisé de manière interactive :  
REPL: Read Eval Print Loop

## Remarques

- ▶ Il n'y a pas d'étape de compilation explicite (cette étape est cachée).
- ▶ Les annotations de type sont ignorées par l'interpréteur.

```
>>> x: int = "ok"
```

```
>>> type(x)
```

```
<class 'str'>
```

# Typage statique

## Outils de vérification statique du typage

- **Mypy** (implementation de référence)
  - ▶ <https://mypy.readthedocs.io/en/stable/index.html>  
(documentation)
  - ▶ [https://mypy.readthedocs.io/en/stable/cheat\\_sheet\\_py3.html](https://mypy.readthedocs.io/en/stable/cheat_sheet_py3.html)  
(synthèse)
  - ▶ [https://mypy.readthedocs.io/en/stable/builtin\\_types.html](https://mypy.readthedocs.io/en/stable/builtin_types.html)  
(types primitifs)
- **Pyright** (inclus avec l'extension VS Code)
  - ▶ <https://github.com/microsoft/pyright>
- **Pytype** (google), **Pyre** (facebook), **PyCharm** (JetBrains), ...

# Cours en ligne sur Python

- ▶ Support du cours *Python for Computational Science*<sup>1</sup>  
**Attention:** pas d'annotations de types dans ce cours
- ▶ *Introduction to Python for Computational Science and Engineering*  
Livre (2022)
- ▶ *Python for Computational Science*  
Transparents des cours (2024)

---

1. *Python for Computational Science*, Hans Fangohr et al. University of Southampton

# Documentation sur Python

Documentation officielle :

- ▶ Python documentation page
- ▶ Python Standard Library
- ▶ Python tutorial
- ▶ Python Package Index (pip repository)

Documentation en ligne sur les types, par exemple :

- ▶ Mypy documentation
- ▶ Type hints cheat sheet
- ▶ Built-in types
- ▶ Pyright

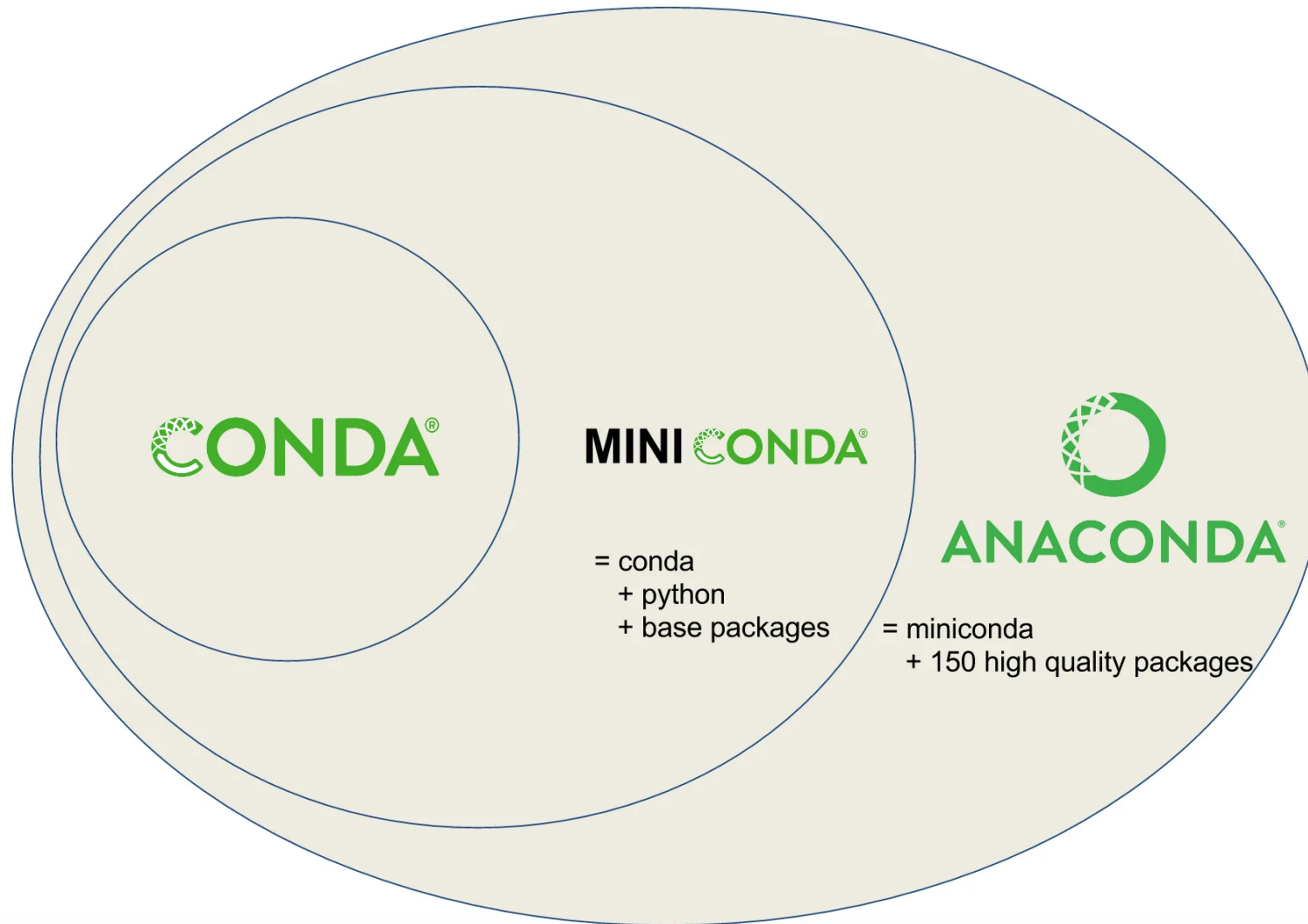
# Installation de Python

## Plusieurs options

- ▶ Gestionnaire de package du système (droits d'administrateur requis)
- ▶ Distribution officielle (droits d'administrateur requis)  
<https://www.python.org/downloads>
- ▶ Anaconda (distribution populaire chez les scientifiques)  
<https://www.anaconda.com/download>
- ▶ Miniconda (distribution minimale de anaconda)  
<https://docs.conda.io/en/main/miniconda.html>

**Remarque.** Anaconda (ou miniconda) permet d'installer différentes versions de Python et de passer facilement de l'une à l'autre (et les droits d'administrateur ne sont pas requis).

# Les distributions Conda



# Installation de packages

**Avec conda ou pip (gestionnaire de package officiel)**

*Dans le terminal, saisir la commande*

```
conda install some-package
```

*ou*

```
pip install some-package
```

**Remarque.** Il est préférable d'éviter de mélanger les deux types de packages. Toutefois, si vous n'avez pas le choix, vous devez installer les packages conda d'abord, puis les packages pip<sup>2</sup>.

---

2. <https://www.anaconda.com/blog/using-pip-in-a-conda-environment>

# Extension Python pour VS Code

The screenshot shows the Visual Studio Code interface with the Python extension page open in the marketplace. The left sidebar displays a list of extensions, with the Python extension selected. The main panel shows the details for the Python extension, including its logo, name, publisher (Microsoft), download count (186,381,800), and rating (4.5 stars from 618 reviews). The extension description highlights its support for IntelliSense (Pylance), debugging, and environment management. The page also includes sections for installation details, marketplace history, and categories.

**EXTENSIONS: MARKETPLACE**

Microsoft Python @sort:installs

- Python** (Microsoft) - Python language support with ext...
- Pylance** (Microsoft) - A performant, feature-rich langua... (491ms)
- Jupyter** (Microsoft) - Jupyter notebook support, intera... (96.2M, 2.5 stars)
- Python Debugger** (Microsoft) - Python Debugger extension using...
- IntelliCode** (Microsoft) - AI-assisted development
- IntelliCode API Usage Examples** (Microsoft) - See relevant code examples from ...
- isort** (Microsoft) - Import organization support for P... (24.6M, 2.5 stars)
- Python Environments** (Microsoft) - Provides a unified python environ...

## Python

Microsoft [microsoft.com](#) | 186,381,800 | ★★★★★ (618)

Python language support with extension access points for IntelliSense (Pylance), ...

[Disable](#) [Uninstall](#) [Switch to Pre-Release Version](#)  Auto Update

[DETAILS](#) [FEATURES](#) [CHANGELOG](#) [EXTENSION PACK](#)

### Python extension for Visual Studio Code

A [Visual Studio Code extension](#) with rich support for the [Python language](#) (for all [actively supported Python versions](#)), providing access points for extensions to seamlessly integrate and offer support for IntelliSense (Pylance), debugging (Python Debugger), formatting, linting, code navigation, refactoring, variable explorer, test explorer, environment management (**NEW** Python Environments Extension).

### Support for [vscode.dev](#)

The Python extension does offer [some support](#) when running on [vscode.dev](#) (which includes [github.dev](#)). This includes partial IntelliSense for open files in the editor.

### Installed extensions

The Python extension will automatically install the following extensions by default to provide the best Python development experience in VS Code:

### Installation

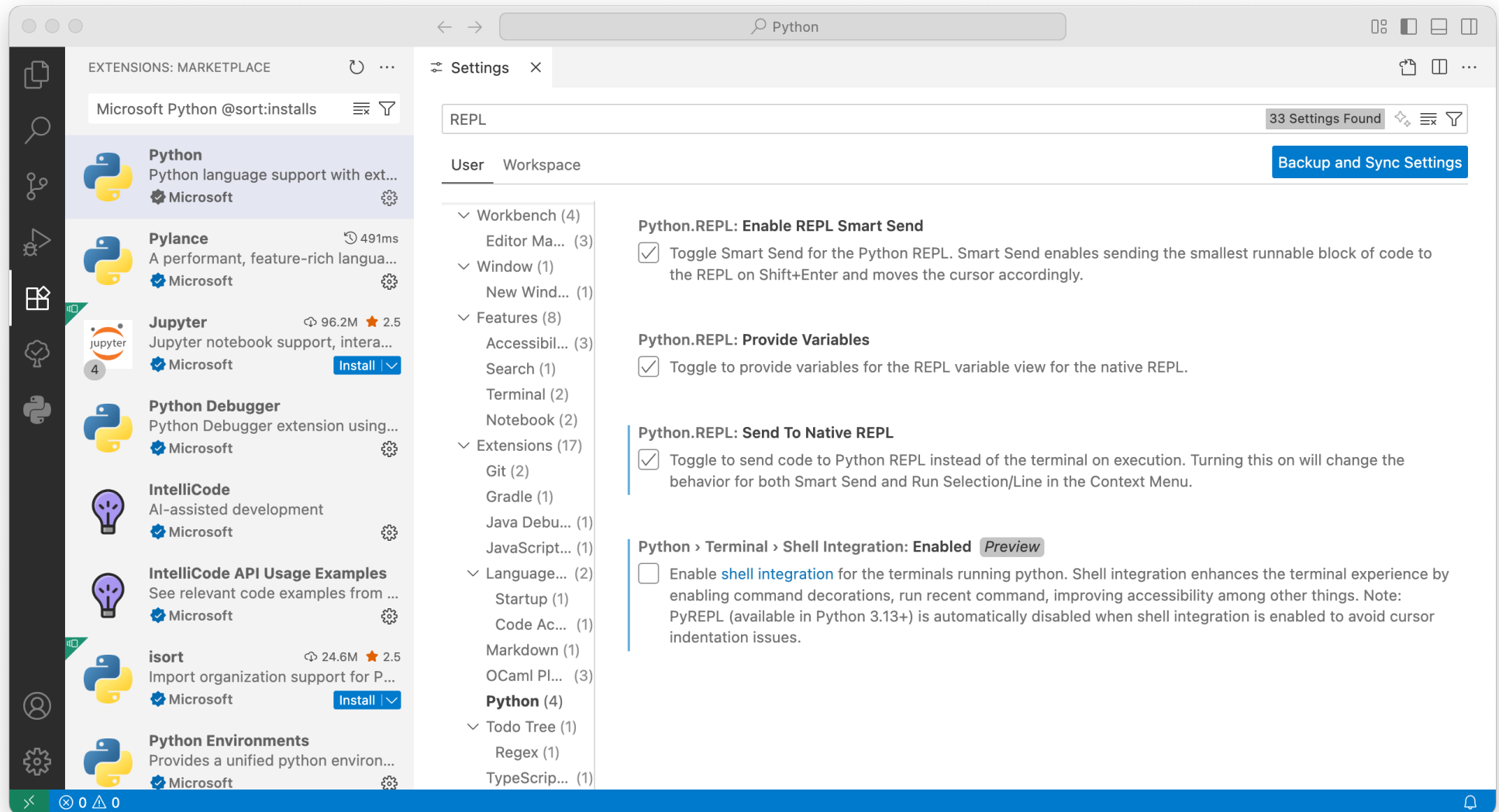
Identifier	ms-python.python
Version	2025.14.0
Last Updated	2025-09-13, 15:08:39
Size	29.13MB
Cache	4.67KB

### Marketplace

Published	2016-01-19, 16:03:11
Last Released	2025-10-02, 12:58:27

### Categories

# Configuration de l'extension Python



# Extension autopep8 pour VS Code

The screenshot shows the VS Code Extensions Marketplace interface. On the left, a list of extensions is visible, including autopep8, Pylint, Flake8, Data Wrangler, ROS, Mypy Type Checker, Tensorboard, and IntelliCode Completions. The main panel displays the details for the autopep8 extension, which is a Microsoft extension with 4,667,257 downloads and a 4-star rating. The extension is currently in a 'Preview' state. The page title is 'Formatter extension for Visual Studio Code using autopep8'. The description states: 'A Visual Studio Code extension with support for the autopep8 formatter. The extension ships with autopep8=2.3.2.' A note indicates that the minimum supported version of autopep8 is 2.3.2. The installation details show the identifier 'ms-python.autopep8', version '2025.2.0', last updated on '2025-03-06', and a size of '2.52MB'. The marketplace information shows it was published on '2022-10-04' and last released on '2025-08-15'. The categories include 'Programming'.

EXTENSIONS: MARKETPLACE

Microsoft Python @sort:installs

autopep8  
Formatting support for Python file...  
Microsoft

Pylint  
Linting support for Python files us...  
Microsoft

Flake8  
Linting support for Python files us...  
Microsoft

Data Wrangler  
Data viewing, cleaning and prepar...  
Microsoft

ROS  
Develop Robot Operating System ...  
Microsoft

Mypy Type Checker  
Type checking support for Python...  
Microsoft

Tensorboard  
Launch and view Tensorboards wi...  
Microsoft

IntelliCode Complet...  
IntelliCode Completions: AI-drive...  
Microsoft

Extension: autopep8

autopep8 **Preview**

Microsoft [microsoft.com](https://microsoft.com) | 4,667,257 | ★★★★★ (15)

Formatting support for Python files using the autopep8 formatter.

Disable Uninstall Switch to Pre-Release Version  Auto Update

DETAILS FEATURES CHANGELOG DEPENDENCIES

## Formatter extension for Visual Studio Code using autopep8

A Visual Studio Code extension with support for the autopep8 formatter. The extension ships with `autopep8=2.3.2`.

Note: The minimum supported version of autopep8 is `2.3.2`. If you have any issues with formatting with autopep8, please report it to [this issue tracker](#) as this extension is just a wrapper around autopep8.

This extension supports for all [actively supported versions](#) of the Python language.

For more information on autopep8, see <https://pypi.org/project/autopep8/>.

## Usage and Features

The autopep8 extension for Visual Studio Code provides formatting support for your Python files. Check out the [Settings section](#) for more details on how to customize this

### Installation

Identifier	ms-python.autopep8
Version	2025.2.0
Last Updated	2025-03-06, 07:51:05
Size	2.52MB

### Marketplace

Published	2022-10-04, 21:55:19
Last Released	2025-08-15, 12:28:09

### Categories

Programming

# Mode « ini » pour « pyproject.toml »

