



# الدليل الشامل لإعداد بيئة عمل بايثون المتكاملة

دليلك الأكاديمي خطوة بخطوة لبناء بيئة بايثون احترافية على جهازك الشخصي، بدءاً من تثبيت النظيف حتى التجهيز الكامل للتحليل والبرمجة



## ⚠ تنبيه هام قبل البدء

نجاح إعداد البيئة يعتمد كلياً على الالتزام بالترتيب. تجاوز أي خطوة قد يؤدي لمشاكل مستقبلية. يرجى القراءة بعناية واتباع كل التعليمات.

✓ اتبع الخطوات بالترتيب المذكور تماماً

✓ لا تتجاوز أي تفاصيل مهما بدت بسيطة

✓ كل خطوة مبنية على التي تسبقها



# المرحلة الأولى: تأسيس الأساس

تثبيت بايثون بشكل نظيف وصحيح هو أساس البيئة البرمجية. التثبيت السليم يمنع معظم الأعطال والأخطاء لاحقاً.

- ✓ التثبيت الصحيح يمنع 90% من المشاكل المستقبلية المتعلقة بالمكتبات
- ✓ ضبط المسارات (PATH) بشكل صحيح يسهل استخدام الأوامر من أي مكان
- ✓ البدء بنظام نظيف من النسخ المتعددة يمنع تعارض الإصدارات
- ✓ التحقق من نجاح التثبيت ضروري قبل الانتقال للخطوات اللاحقة



## >فحص وتنظيف الجهاز من نسخ بايثون القديمة

ضرورة حذف كل نسخ بايثون القديمة أو المتعارضة. يمنع تعارض المسارات والأدوات. استخدم الأمر التالي في PowerShell:

```
Windows PowerShell

PS C:\Users\user> where.exe python
INFO: Could not find files for the given pattern(s).

PS C:\Users\user> where.exe python
C:\Users\YourName\AppData\Local\Programs\Python\Python312\python.exe

PS C:\Users\user> where.exe python
C:\Users\YourName\AppData\Local\Programs\Python\Python312\python.exe
C:\Users\YourName\Anaconda3\python.exe
C:\Program Files\Python37\python.exe
```

### ❌ الحالة الخطيرة

تظهر مسارات متعددة، يجب إزالة جميع النسخ لتجنب تعارض المكتبات.

### ⚠️ الحالة المتوسطة

يظهر مسار واحد فقط، تحقق من رقم الإصدار (3.8 أو أحدث).

### ✅ الحالة المثالية

لا توجد نسخ مثبتة، رسالة: Could not find files. جهازك نظيف وجاهز للتثبيت.

# تنزيل وتثبيت بايثون بالطريقة الصحيحة

## ١ تنزيل ملف التثبيت

اذهب إلى الموقع الرسمي: <https://www.python.org/downloads/>  
اضغط على الزر الأصفر الكبير Download Python لتنزيل أحدث إصدار.

## ٢ عملية التثبيت (اللحظة الحاسمة)

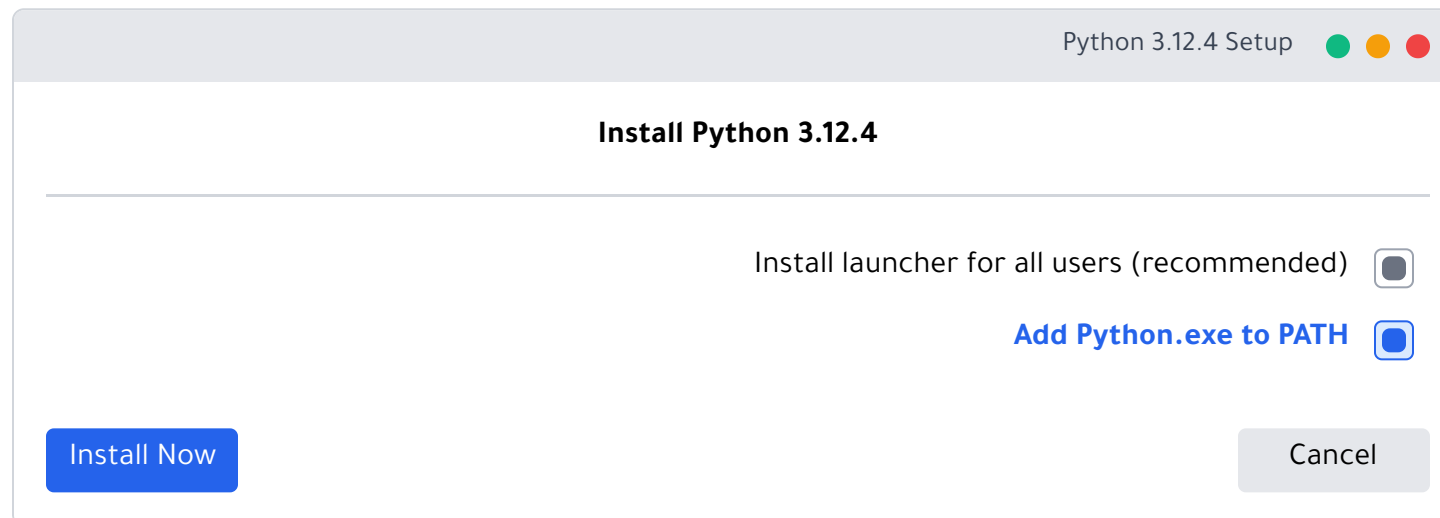
بعد تشغيل ملف التثبيت، ستظهر نافذة التثبيت. **انتبه للخيار المهم جداً:**

### يجب تحديد خيار:

Add Python.exe to PATH

هذا الخيار يخبر نظام ويندوز بمكان وجود بايثون، مما يسمح بتشغيل الأوامر من أي مكان في الطرفية.  
**عدم تحديد هذا الخيار هو سبب 99% من مشاكل المبتدئين!**

بعد التأكد من وضع علامة الصح، اضغط على **Install Now**



## التحقق النهائي من نجاح التثبيت

للتأكد 100% من نجاح تثبيت بايثون، افتح نافذة PowerShell جديدة تماماً ونفذ الأوامر التالية:

```
Windows PowerShell

PS C:\Users\user> python --version

Python 3.12.4

PS C:\Users\user> pip --version

pip 24.0 from C:\Users\YourName\AppData\Local\Programs\Python\Python312\lib\site-packages\pip (python 3.12)

PS C:\Users\user> where.exe python

C:\Users\YourName\AppData\Local\Programs\Python\Python312\python.exe
```

### النتائج المتوقعة للتثبيت الناجح

- ✓ **الأمر الأول:** يجب أن يظهر رقم إصدار بايثون (مثلاً Python 3.12.4)
  - ✓ **الأمر الثاني:** يظهر رقم إصدار pip ومسار تثبيته المرتبط بنفس نسخة بايثون
  - ✓ **الأمر الثالث:** يجب أن يظهر مسار واحد فقط لنسخة بايثون التي قمت بتثبيتها
- 💡 إذا نجحت جميع هذه الاختبارات، فتهانينا! لقد أنجزت أهم جزء من الإعداد بنجاح.

## ✂️ المرحلة الثانية: إعداد ورشة العمل

بعد تثبيت "المحرك" (بايثون)، نحتاج إلى "ورشة العمل" التي سنكتب فيها الأكواد. سنقوم بتثبيت أداتين أساسيتين يستخدمهما المحترفون:

### Jupyter Notebook

لتحليل البيانات والذكاء الاصطناعي

- ✓ بيئة تفاعلية تدعم تنفيذ الكود على شكل خلايا
- ✓ عرض المخرجات والرسوم البيانية مباشرة
- ✓ الأداة رقم واحد في علم البيانات والبحث العلمي

### Visual Studio Code

للبرمجة العامة وتطوير المشاريع

- ✓ محرر أكواد قوي ومجاني من مايكروسوفت
- ✓ يدعم التكامل مع Git وإضافات متنوعة
- ✓ مناسب للمشاريع الكبيرة ومتعددة الملفات





## تنزيل المحرر

قم بزيارة الموقع الرسمي لـ VS Code: <https://code.visualstudio.com> سيكتشف الموقع تلقائياً نظام التشغيل ويعرض زر التنزيل المناسب. اضغط عليه لتنزيل برنامج التثبيت.

## الخيارات المهمة أثناء التثبيت

### Add to PATH ✓

ضروري للغاية. يسمح بتشغيل VS Code من أي نافذة طرفية مباشرة.

### Add "Open with Code" to Explorer context menu ✓

يضيف خيار "فتح بواسطة Code" عند النقر بزر الماوس الأيمن على أي ملف أو مجلد.

### Register Code as an editor for supported file types ✓

يجعل VS Code المحرر الافتراضي لأنواع الملفات البرمجية.

## الفوائد

بيئة تطوير متكاملة وخفيفة

دعم كامل للغة بايثون عبر الإضافات

واجهة مخصصة قابلة للتوسيع والتعديل




# تثبيت إضافات بايثون الأساسية على VS Code

الإضافات (Extensions) هي أدوات صغيرة تضيف ميزات قوية لـ VS Code وتجعل تجربة البرمجة أكثر احترافية. لإضافة إضافة جديدة، انقر على أيقونة المربعات الأربعة في الشريط الجانبي.

 **Python** مايكروسوفت أساسية

الإضافة الرئيسية للغة بايثون، توفر الإكمال التلقائي، تصحيح الأخطاء، تشغيل الأكواد، والكثير من الميزات الأساسية.

 **Jupyter** مايكروسوفت أساسية

تسمح لك بتشغيل ملفات Jupyter Notebooks التفاعلية مباشرة داخل VS Code، مثالية لتحليل البيانات والعرض البصري.

 **Prettier - Code formatter** موصى بها

تقوم بترتيب وتنسيق الكود الخاص بك تلقائياً ليصبح نظيفاً ومنظماً وسهل القراءة.

 **Arabic Language Pack** مايكروسوفت اختيارية

إذا كنت تفضل أن تكون واجهة البرنامج باللغة العربية، قم بتثبيت هذه الحزمة اللغوية.



# تثبيت Jupyter Notebook و JupyterLab

Jupyter هي الأداة رقم واحد في عالم علم البيانات والذكاء الاصطناعي. تسمح لك بكتابة الكود وتشغيله ورؤية النتائج فوراً في مكان واحد.

```
Windows PowerShell

PS C:\Users\user> pip install notebook jupyterlab

Collecting notebook
  Downloading notebook-7.0.6-py3-none-any.whl (13.8 MB)
    |████████████████████████████████████████| 13.8 MB 6.4 MB/s
Collecting jupyterlab
  Downloading jupyterlab-4.0.9-py3-none-any.whl (9.6 MB)
    |████████████████████████████████████████| 9.6 MB 8.2 MB/s
...
Successfully installed ipykernel-6.26.0 jupyter-client-8.5.0 jupyterlab-4.0.9 notebook-7.0.6 ... (وآخرى)
```

## طرق تشغيل Jupyter بعد التثبيت

### JupyterLab 🧪

الواجهة الحديثة مع ميزات متقدمة أكثر

```
PS C:\Users\user> jupyter lab
```

سيفتح متصفحك على واجهة JupyterLab المتطورة. من الشاشة الرئيسية، اختر **Python 3** تحت قسم **Notebook**.

#### 💡 اختبار التثبيت:

اكتب ("يعمل بنجاح JupyterLab") في الخلية واضغط **Shift + Enter**

### Jupyter Notebook 📖

الواجهة الكلاسيكية الأبسط والأكثر شيوعاً

```
PS C:\Users\user> jupyter notebook
```

سيفتح متصفحك تلقائياً على واجهة Jupyter الكلاسيكية. اضغط على زر **New** ثم اختر **Python 3** لإنشاء دفتر ملاحظات جديد.

#### 💡 اختبار التثبيت:

اكتب ("يعمل بنجاح Jupyter Notebook") في خلية واضغط **Shift + Enter**

## المرحلة الثالثة: تجهيز صندوق الأدوات

قبل البدء بتثبيت المكتبات الأساسية، هناك بعض الخطوات الوقائية الضرورية لضمان عملية تثبيت سلسلة وخالية من المشاكل:

### التأكد من وجود مساحة كافية



تحقق من وجود مساحة فارغة لا تقل عن 10 جيجابايت على القرص الرئيسي C:.  
بعض المكتبات مثل TensorFlow وPyTorch كبيرة جداً وتحتاج مساحة تخزين كبيرة.

### إيقاف مكافح الفيروسات مؤقتاً



أحياناً يمنع برنامج الحماية (Windows Defender) عملية التثبيت عن طريق الخطأ.  
قم بإيقافه مؤقتاً فقط خلال مرحلة التثبيت ثم أعد تفعيله بعد الانتهاء.

### تشغيل PowerShell كمسؤول



لتجنب أخطاء الصلاحيات، ابحث عن PowerShell في قائمة ابدأ، انقر بزر الماوس الأيمن، واختار "Run as administrator".

### نصيحة هامة



تحل بالصبر الشديد! تثبيت المكتبات سيستغرق وقتاً طويلاً (15-30 دقيقة حسب سرعة الإنترنت والحاسوب). لا تقاطع العملية أو تغلق النافذة أثناء التثبيت.



# تثبيت الحزمة الرئيسية للمكتبات الأساسية

استخدم الأمر التالي من PowerShell (كمدير) لتثبيت جميع المكتبات الأساسية دفعة واحدة:

Windows PowerShell (Administrator)

```
PS C:\Users\user> pip install numpy pandas matplotlib seaborn scikit-learn scikit-fuzzy tensorflow torch pygame  
deap Flask experta
```

Collecting numpy

Downloading numpy-1.24.3-cp312-cp312-win\_amd64.whl (15.7 MB)

15.7/15.7 MB 8.2 MB/s eta 0:00:00

Collecting pandas

Downloading pandas-2.0.1-cp312-cp312-win\_amd64.whl (10.4 MB)

10.4/10.4 MB 9.1 MB/s eta 0:00:00

...

WARNING: You are using pip version 22.3.1; however, version 23.0.1 is available.  
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

Successfully installed numpy-1.24.3 pandas-2.0.1 matplotlib-3.7.1 seaborn-0.12.2 scikit-learn-1.3.0 scikit-fuzzy-0.4.2  
tensorflow-2.12.0 torch-2.0.1 pygame-2.5.0 deap-1.4.0 Flask-2.3.2 experta-1.9.4

## تحديثات ضمان التوافق

### تحديث مكتبة frozendict

```
pip install --upgrade frozendict
```

### تحديث أداة pip

```
python.exe -m pip install --upgrade pip
```

### هام جداً!

بعد انتهاء التثبيت، ارجع وقم بإعادة تفعيل الحماية في الوقت الحقيقي (Real-time protection) التي أوقفتها في الخطوة السابقة.

## المرحلة الرابعة: إجراء الاختبار النهائي

اختبر عمل المكتبات في VS Code أو Jupyter. شغل ملف اختبار الأكواد وتأكد من عدم وجود رسائل خطأ، وكل المكتبات تعمل بسلاسة.

### الاختبار باستخدام Jupyter Notebook

الأمثل لتحليل البيانات والاستكشاف التفاعلي

١ شغل Jupyter Notebook أو JupyterLab

عبر كتابة jupyter notebook في PowerShell

٢ تصفح المجلدات حتى تجد ملف الاختبار test\_lib.ipynb

٣ انقر على الخلية الأولى، ثم اضغط Shift + Enter لكل خلية

٤ حلل النتيجة:

✓ النتيجة الناجحة: مخرجات الخلايا تظهر أسفلها دون أخطاء

```
matplotlib v3.7.1
scikit-learn v1.3.0
جميع المكتبات تعمل بشكل صحيح
```

### </> الاختبار باستخدام Visual Studio Code

الأنسب للمشاريع الكاملة وتطوير البرامج

١ افتح VS Code وافتح المجلد الذي يحتوي مواد الدورة

باستخدام File > Open Folder

٢ تصفح المجلدات حتى تجد ملف الاختبار test\_lib.py

٣ اضغط على أيقونة التشغيل (المثلث) في الزاوية العليا

٤ حلل النتيجة في الطرفية (Terminal) بالأسفل:

✓ النتيجة الناجحة: عرض لإصدارات المكتبات وبدون أخطاء

```
NumPy v1.24.3
Pandas v2.0.1
TensorFlow v2.12.0
<< . كل شيء يعمل بشكل صحيح! بيئة العمل جاهزة >>
```

# مبروك! أنجزت إعداد بيئة بايثون باحترافية



إذا وصلت إلى هنا وحصلت على نتيجة ناجحة في الاختبار النهائي، فأنت قد قمت بإعداد بيئة تطوير بايثون كاملة واحترافية بنجاح 100%. أنت الآن مستعد تماماً للانطلاق في رحلتك البرمجية.



## الخطوات القادمة

### طبق مشاريع عملية

حاول تطبيق المعرفة في مشاريع صغيرة



### استكشف خدمات السحابة

جرب Google Colab لتنفيذ مشاريع متقدمة



### استكشف المكتبات

ابدأ بتعلم NumPy وPandas لمعالجة البيانات



### انضم للمجتمع

شارك في منتديات بايثون وGitHub للتعلم

