



# Übungslektion 1 -- Python & JupyterLab

Informatik II

17.02.2026

# Willkommen!

## Polybox



Passwort: jschul

## Personal Website



[https://jschultev.github.io/personal\\_website/](https://jschultev.github.io/personal_website/)

# Über Mich

- Johannes (jschul)
- 6. Semester Maschinenbau
- Info II TA zum zweiten Mal
- Electronics und Controls Engineer bei Fokusprojekt PEGASUS (ARIS).  
Rotating Detonation Rocket Engine

# 1. Einführung

---

In diesen Slides präsentieren wir 3 Tools für Python:

- JupyterLab
- CodeExpert
- Terminal

# Warum diese Tools?

- JupyterLab ist ein weit verbreitetes Tool zum Managen und Schreiben von Python Skripts. Es wird in Vorlesungen und Projekten in späteren Semestern verwendet werden.
- CodeExpert wird für die Übungslektionen und die Hausaufgaben verwendet. Grund dafür ist das automatisierte Testsystem.
- Der Terminal wird verwendet um Python Code schnell und unkompliziert zu testen.

## 2. JupyterLab

---



# Was ist JupyterLab?

- JupyterLab ermöglicht das interaktive Schreiben und Ausführen von Code in einem sogenannten JupyterLab Notebook.
- In dieser Vorlesung verwenden wir (zusätzlich zu CodeExpert) JupyterLab zum Schreiben von Python-Programmen.
- Code-Beispiele in JupyterLab.

# Beispiel

```
inf2example.ipynb
```

```
[2]: def factorial(x):  
      if x == 0 or x == 1:  
          return 1  
      else:  
          return x * factorial(x-1)
```

```
[3]: q = factorial(6)  
      print(q)
```

```
720
```

```
[4]: print(q-3)
```

```
717
```

```
[ ]:
```

# Warum JupyterLab?

## JupyterLab

- zeigt Ihnen den Output verschiedener Versionen Ihres Codes;
- erlaubt es Ihnen, Ihre Arbeit einfach in einzelne Teile zu unterteilen; und
- unterstützt Kommentare, die sowohl Text als auch Diagramme enthalten. Dies erlaubt es anderen Personen Ihren Code zu verstehen.

Mehr Informationen finden Sie in der JupyterLab Dokumentation:

<https://jupyterlab.readthedocs.io/en/stable/index.html>

# Setup Überblick

1. Laden Sie Python 3.11 herunter und installieren Sie es.

Möglicherweise ist bereits eine ältere Python Version auf Ihrem Gerät installiert. Wir werden jedoch Version 3.11 installieren, um spätere Probleme aufgrund von Versionsunterschieden zu vermeiden.

2. Updaten des Python Package Managers: pip

pip wird zum Installieren öffentlicher Bibliotheken (packages) mit Python Code verwendet.

3. Installieren von JupyterLab und anderen Python packages die während des Semesters benötigt werden.

Speziell werden numpy (schnelle Vektorrechnungen), pandas (arbeiten mit Tabellen), und matplotlib (Graphen und Plots) installiert.

4. Starten und Stoppen von JupyterLab, Öffnen von Jupyter Notebooks.

In den folgenden Abschnitten werden Schritte 1-3 für Windows, macOS, und Ubuntu-Linux detailliert beschrieben. Danach geht es in Abschnitt 7 weiter.

## 3. Python & JupyterLab installieren (Windows)

---

- In diesem Abschnitt zeigen wir, wie man Python & JupyterLab unter Windows installiert. Eine andere Möglichkeit, JupyterLab auszuführen, besteht darin, es mit Anaconda zu installieren.
- Anaconda ist eine weit verbreitete, quelloffene Python-Distribution, die die Paketverwaltung und -bereitstellung vereinfacht.
- Wenn Sie Anaconda bereits installiert haben (was in einigen Computerlabors in CAB der Fall ist) oder es verwenden möchten, gehen Sie zu section 4.

# Schritt 1a: Python herunterladen (Windows)

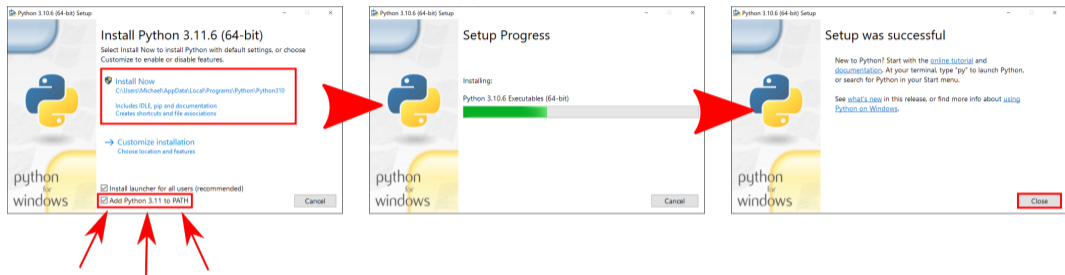
## Files

Version	Operating System	Description	MD5 Sum	File Size	GPG
<a href="#">Gzipped source tarball</a>	Source release		d76638ca8bf57e44ef0841d2cde557a0	25986768	<a href="#">SIG</a>
<a href="#">XZ compressed source tarball</a>	Source release		afc7e14f7118d10d1ba95ae8e2134bf0	19600672	<a href="#">SIG</a>
<a href="#">macOS 64-bit universal2 installer</a>	macOS	for macOS 10.9 and later	2ce68dc6cb870ed3beea8a20b0de71fc	40826114	<a href="#">SIG</a>
<a href="#">Windows embeddable package (32-bit)</a>	Windows		a62cca7ea561a037e54b4c0d120c2b0a	7608928	<a href="#">SIG</a>
<a href="#">Windows embeddable package (64-bit)</a>	Windows		37303f03e19563fa87722d9df11d0fa0	8585728	<a href="#">SIG</a>
<a href="#">Windows help file</a>	Windows		0aee63c8fb87dc71bf2bcc1f62231389	9329034	<a href="#">SIG</a>
<a href="#">Windows installer (32-bit)</a>	Windows		c4aa2cd7d62304c804e45a51696f2a88	27750096	<a href="#">SIG</a>
<a href="#">Windows installer (64-bit)</a>	Windows	Recommended	8f46453e68ef38e5544a76d84df3994c	28916488	<a href="#">SIG</a>

Laden Sie Python 3.11.9 herunter:

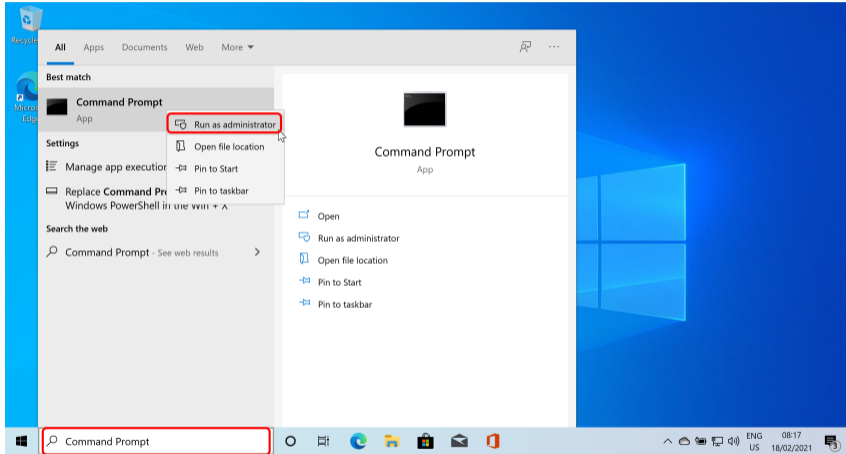
<https://www.python.org/downloads/release/python-3119/>.

# Schritt 1b: Installer ausführen (Windows)



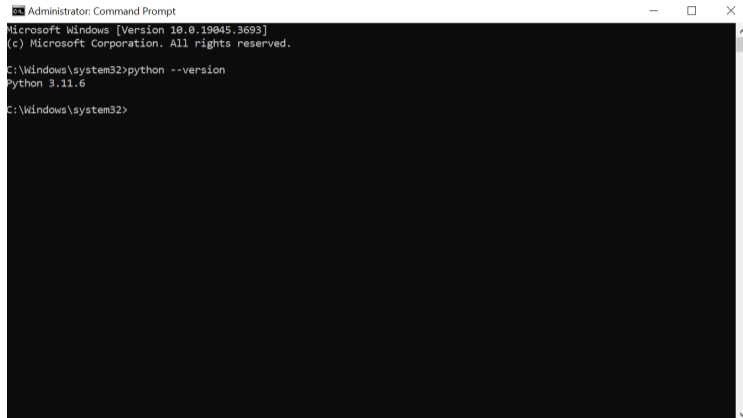
- Finden und führen Sie den Installer aus.
- **Wichtig!** Stellen Sie sicher, dass “Add Python 3.11 to PATH” ausgewählt ist.
- Schliessen Sie den Installer sobald die Installation abgeschlossen ist.

# Schritt 2a: Command Prompt öffnen (Windows)



Suchen und öffnen Sie die Windows Command Prompt als Administrator (Rechtsklick).

## Schritt 2b: Python-Version überprüfen (Windows)



```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.19045.3693]
(c) Microsoft Corporation. All rights reserved.

C:\Windows\system32>python --version
Python 3.11.6

C:\Windows\system32>
```

Stellen Sie sicher, dass Python korrekt installiert wurde indem Sie `python --version` eingeben und “Enter” drücken.

## Schritt 2c: Packages installieren (Windows)

Geben Sie die folgenden Befehle in der Command Prompt ein und drücken Sie "Enter":

- `python -m pip install --upgrade pip wheel`

Damit wird der Python Package Manager `pip` upgegraded und es wird `wheel` installiert. `wheel` ermöglicht das schnelle Installieren von Python Packages.

Wenn alles funktioniert hat sollten Sie "Successfully installed pip-23.0 wheel-0.38.4" sehen.

- `python -m pip install numpy matplotlib pandas jupyterlab timetable openpyxl seaborn[stats]`

Damit werden JupyterLab und andere Python Packages installiert. Falls alles funktioniert hat sollten Sie "Successfully installed ..." sehen. Fehler werden üblicherweise mit rotem Text kommuniziert. Sollte ein Fehler auftreten, schauen Sie sich die folgenden Abschnitte für mögliche Lösungen an.

# Fehlerbehebung Teil 1 (Windows)

- Sollten Sie den folgenden Fehler während des letzten Schrittes sehen

```
building 'winpty.pywinpty' extension
error: Microsoft Visual C++ 14.0 or greater is required. Get it with "Microsoft C++ Build Tools": https://visualstudio-microsoft.com/visual-cpp-build-tools/
error: failed building wheel for pywinpty
Running setup.py clean for pywinpty
Failed to build pywinpty
Installing collected packages: pywinpty, pandocfilters, nbclient, mistune, jupyterlab-pygments, jinja2, ipython, idna, entrypoints, defusedxml, cffi, bleach, jupyterlab
Running setup.py install for pywinpty ... error
```

sollten Sie Visual C++ 2015 Build Tools installieren:

<http://go.microsoft.com/fwlink/?LinkId=691126&fixForIE=.exe>

- Nachdem die Installation abgeschlossen ist, müssen Sie Ihren Computer neu starten und den folgenden Befehl ausführen  
`python -m pip install --upgrade setuptools`
- Sie können jetzt Schritt 2c erneut probieren.

# Fehlerbehebung Teil 2 (Windows)

Falls immer noch Fehler vorhanden sind (zum Beispiel 'winpty.h' is missing), probieren Sie folgendes:

- Führen Sie `python -m pip install pipwin` aus
- Danach, führen Sie folgende Befehle aus:
  - `python -m pipwin install numpy`
  - `python -m pipwin install matplotlib`
  - `python -m pipwin install pandas`
  - `python -m pipwin install jupyter`
  - `python -m pipwin install timetable`
  - `python -m pipwin install openpyxl`
  - `python -m pipwin install seaborn[stats]`

# Fehlerbehebung Teil 3 (Windows)

Sollte es immer noch nicht funktionieren müssen Sie das fehlende `pywinpty` File manuell installieren.

- Öffnen Sie `https://pypi.org/project/pywinpty/2.0.12/#files`.
- Laden Sie das zu Ihrer Python Installation passende `whl` File herunter. Falls Sie unsicher sind, versuchen Sie die `cp311` Version zuerst.
- Installieren Sie das File mit dem Befehl `python -m pip install [FILE]` wobei `[FILE]` der Pfad zum heruntergeladenen File ist.
- Führen Sie nun die Befehle aus Schritt 2c erneut aus.

# Schritt 3: Command Prompt neu öffnen (Windows)



```
Administrator: Command Prompt
C:\Windows\system32>exit
```

- Schliessen Sie die Command Prompt indem Sie `exit` eingeben und “Enter” drücken.
- Öffnen Sie eine neue Command Prompt. Dieses mal aber **nicht** als Administrator
- Fahren Sie mit Abschnitt 7 fort um JupyterLab zu starten.

## 4. Anaconda & JupyterLab installieren (Windows)

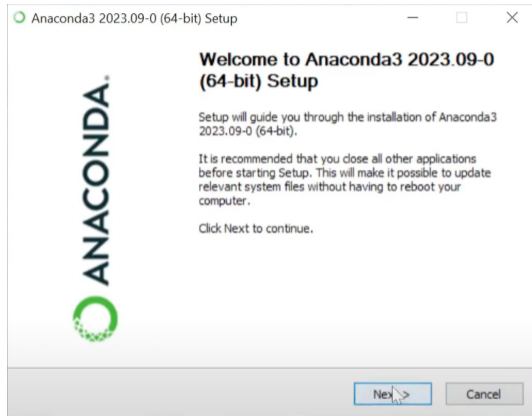
---

Wenn Anaconda bereits auf Ihrem Computer installiert ist, fahren Sie mit Schritt 2 zur Installation von JupyterLab fort. Auf Windows-Computern in CAB ist Anaconda bereits installiert. *Notiz: Wenn Sie bereits der normalen Python-Installation gefolgt sind, ist Anaconda nicht unbedingt nötig.*

# Schritt 1a: Anaconda herunterladen (Windows)

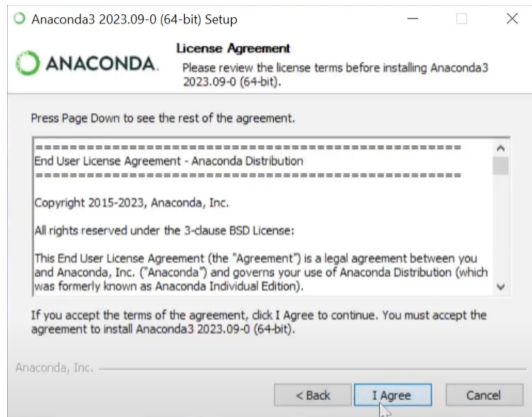
- Laden Sie die neuste Anaconda Version von <https://www.anaconda.com/download/success> herunter
- Navigieren Sie zum Ordner in welchen das File heruntergeladenen wurde (wahrscheinlich der *Downloads* Ordner).
- Das Files welches Sie heruntergeladenen haben sollte `Anaconda3-2023.09-0-Windows-x86_64.exe` oder ähnlich heissen. Je nachdem wann genau Sie dieser Anleitung folgen ist es möglich, dass das File einen leicht anderen Namen hat.

# Schritt 1b: Anaconda installieren (Windows)



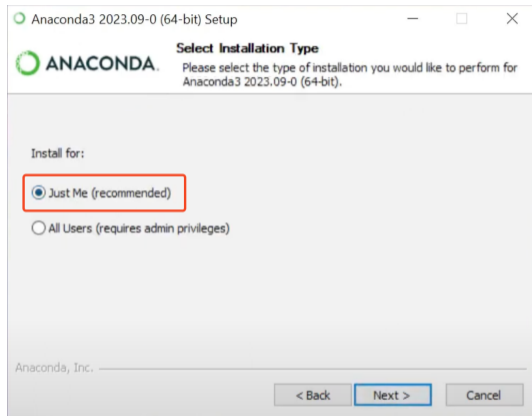
- Führen Sie den Installer aus.

# Schritt 1b: Anaconda installieren (Windows)



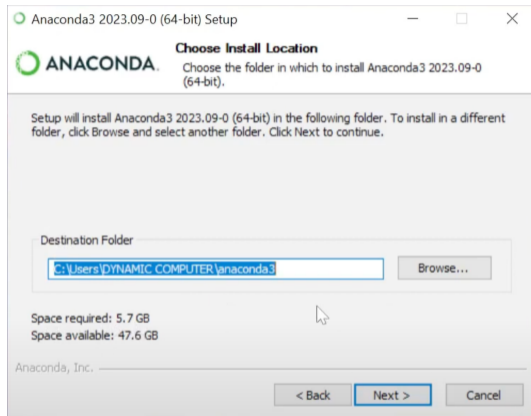
- Führen Sie den Installer aus.

# Schritt 1b: Anaconda installieren (Windows)



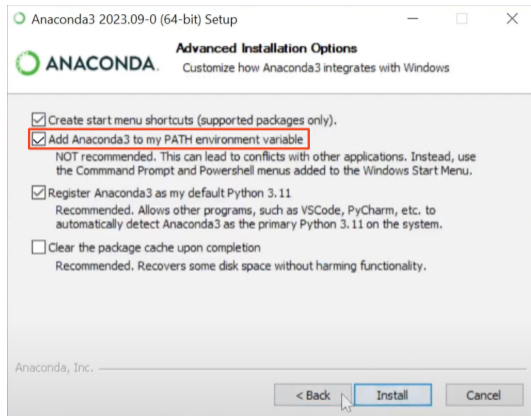
- Führen Sie den Installer aus.

# Schritt 1b: Anaconda installieren (Windows)



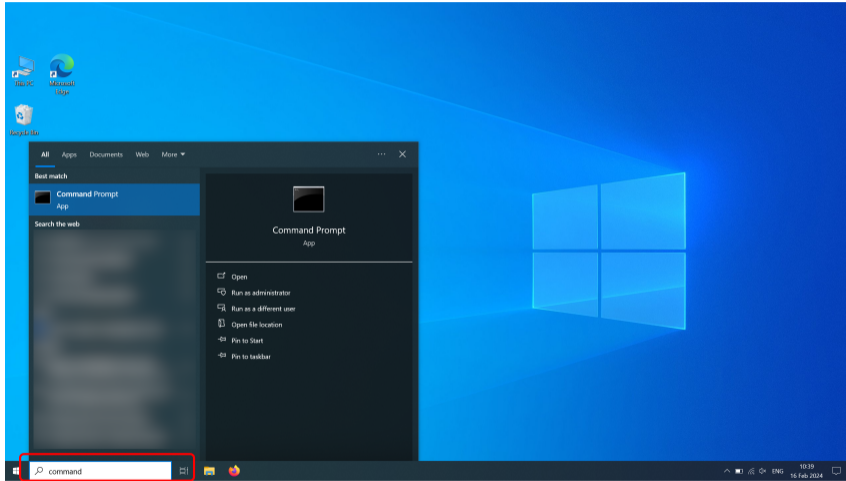
- Führen Sie den Installer aus.

# Schritt 1b: Anaconda installieren (Windows)



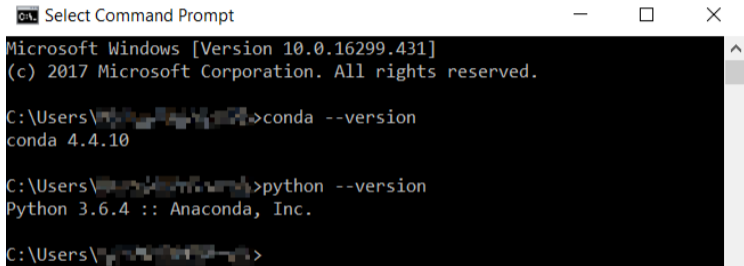
- Führen Sie den Installer aus.
- Wenn Sie das Installation Options Menü erreichen, markieren Sie 'Add Anaconda3 to my PATH environment variable' und wählen Sie **INSTALL** aus.

# Schritt 2a: Command Prompt öffnen (Windows)



Suchen und öffnen Sie die Windows Command Prompt.

## Schritt 2b: Python-Version überprüfen (Windows)



```
Microsoft Windows [Version 10.0.16299.431]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\>conda --version
conda 4.4.10

C:\Users\>python --version
Python 3.6.4 :: Anaconda, Inc.

C:\Users\>
```

- Stellen Sie sicher, dass Anaconda korrekt installiert wurde indem Sie `conda --version` eingeben und "Enter" drücken.
- Machen Sie dasselbe mit Python, indem Sie `python --version` eingeben.

## Schritt 2c: Neues conda Environment (Windows)

- Öffnen Sie einen neuen Command Prompt.
- Führen Sie `conda create --name inf2 python=3.11` aus. *inf2* ist der Name des erstellten Environments. Sie können einen beliebigen Namen auswählen.
- Führen Sie `conda activate inf2` aus um das Environment zu aktivieren. Die Kommandozeile sollte nun mit (*inf2*) ganz links beginnen.
- Führen Sie `conda install numpy matplotlib pandas jupyterlab openpyxl seaborn[stats]` und `pip install timetable` aus um die notwendigen Packages zu installieren.
- Schliessen Sie den Command Prompt.

## Schritt 3: JupyterLab ausführen (Windows)

- Öffnen Sie einen neuen Command Prompt.
- Führen Sie `conda activate inf2` aus um das Environment zu aktivieren.
- Geben Sie `jupyter lab` ein und drücken Sie "Enter" um JupyterLab zu starten.

Hinweis: Alternativ kann man auch den Anaconda Navigator öffnen und JupyterLab über die Benutzeroberfläche starten.

# Fehlerbehebung Teil 1 (Windows Anaconda)

**Problem:** `conda` wird nicht als interner oder externer Befehl erkannt.

- Dieser Fehler tritt auf, wenn Anaconda nicht zum System-PATH hinzugefügt wurde.
- Stellen Sie während der Installation sicher, dass das Kästchen "Add Anaconda to my PATH environment variable" aktiviert ist. Wenn Anaconda bereits installiert ist, müssen Sie möglicherweise Ihre Eingabeaufforderung oder Ihr System neu starten, damit die Änderungen wirksam werden.

## 5. Python & JupyterLab installieren (Mac)

# Schritt 1a: Python herunterladen (Mac)

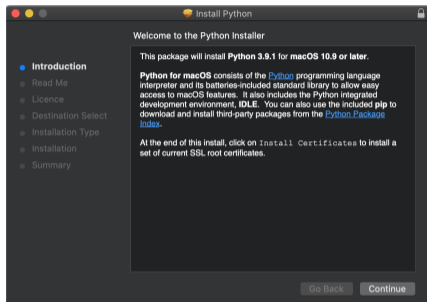
## Files

Version	Operating System	Description	MD5 Sum	File Size	GPG
<a href="#">Gzipped source tarball</a>	Source release		d76638ca8bf57e44ef0841d2cde557a0	25986768	<a href="#">SIG</a>
<a href="#">XZ compressed source tarball</a>	Source release		afc7e14f7118d10d1ba95ae8e2134bf0	19600672	<a href="#">SIG</a>
<a href="#">macOS 64-bit universal2 installer</a>	macOS	for macOS 10.9 and later	2ce68dc6cb870ed3beea8a20b0de71fc	40826114	<a href="#">SIG</a>
<a href="#">Windows embeddable package (32-bit)</a>	Windows		a62cca7ea561a037e54b4c0d120c2b0a	7608928	<a href="#">SIG</a>
<a href="#">Windows embeddable package (64-bit)</a>	Windows		37303f03e19563fa87722d9df11d0fa0	8585728	<a href="#">SIG</a>
<a href="#">Windows help file</a>	Windows		0aee63c8fb87dc71bf2bcc1f62231389	9329034	<a href="#">SIG</a>
<a href="#">Windows installer (32-bit)</a>	Windows		c4aa2cd7d62304c804e45a51696f2a88	27750096	<a href="#">SIG</a>
<a href="#">Windows installer (64-bit)</a>	Windows	Recommended	8f46453e68ef38e5544a76d84df3994c	28916488	<a href="#">SIG</a>

Laden Sie Python 3.11.9 herunter:

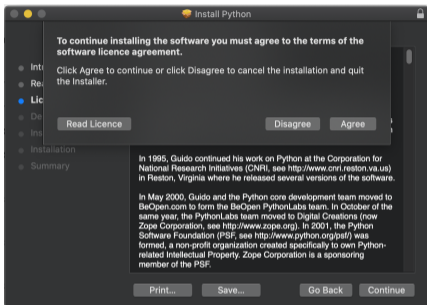
<https://www.python.org/downloads/release/python-3119/>.

# Schritt 1b: Installer ausführen (Mac)



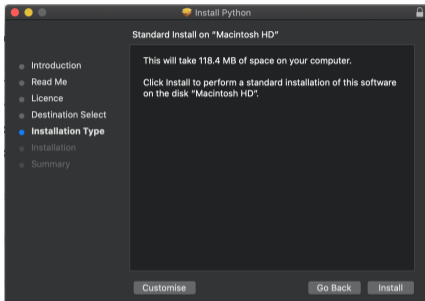
- Finden Sie den Installer und führen Sie ihn aus. Wenn Sie das Installation Type Menü erreichen, wählen Sie **INSTALL** aus (**CUSTOMIZE** sollte nicht notwendig sein).
- Schliessen Sie den Installer sobald dieser fertig ist.

# Schritt 1b: Installer ausführen (Mac)



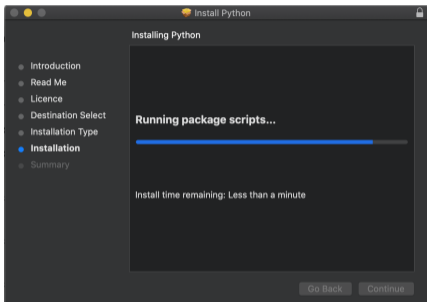
- Finden Sie den Installer und führen Sie ihn aus. Wenn Sie das Installation Type Menü erreichen, wählen Sie **INSTALL** aus (**CUSTOMIZE** sollte nicht notwendig sein).
- Schliessen Sie den Installer sobald dieser fertig ist.

# Schritt 1b: Installer ausführen (Mac)



- Finden Sie den Installer und führen Sie ihn aus. Wenn Sie das Installation Type Menü erreichen, wählen Sie **INSTALL** aus (**CUSTOMIZE** sollte nicht notwendig sein).
- Schliessen Sie den Installer sobald dieser fertig ist.

# Schritt 1b: Installer ausführen (Mac)



- Finden Sie den Installer und führen Sie ihn aus. Wenn Sie das Installation Type Menü erreichen, wählen Sie **INSTALL** aus (**CUSTOMIZE** sollte nicht notwendig sein).
- Schliessen Sie den Installer sobald dieser fertig ist.

# Schritt 2a: Terminal öffnen (Mac)

Öffnen Sie einen `Terminal` indem Sie entweder

- Mit `Spotlight` nach `Terminal` suchen und das oberste Resultat auswählen; oder
- `Go` -> `Utilities` -> `Terminal` auswählen.

Sollte das nicht funktionieren können Sie auch den `Finder` nutzen um `Applications` zu öffnen und dann `Utilities` -> `Terminal` auswählen.



## Schritt 2b: Python-Version überprüfen (Mac)

```
Last login: Mon Aug 22 13:19:57 on console
user@users-iMacPro $\dollar$ python3.11 --version
Python 3.11.6
```

Stellen Sie sicher, dass Python korrekt installiert ist indem Sie `python3.11 --version` eingeben und “Enter” drücken.

## Schritt 2c: JupyterLab installieren (Mac)

Geben Sie die folgenden Befehle im Terminal ein und drücken Sie “Enter”:

- `python3.11 -m pip install --upgrade pip wheel`

Damit wird der Python Package Manager `pip` upgedegrad und es wird `wheel` installiert. `wheel` ermöglicht das schnelle Installieren von Python Packages.

Falls alles funktioniert sollten Sie “Successfully installed pip-23.0 wheel-0.38.4” sehen.

- `python3.11 -m pip install numpy matplotlib pandas jupyterlab timetable openpyxl seaborn[stats]`

Damit werden JupyterLab und andere Python Packages installiert. Falls alles funktioniert hat sollten Sie “Successfully installed ...” sehen. Fehler werden üblicherweise mit rotem Text kommuniziert.

Fahren Sie mit Abschnitt 7 fort um JupyterLab zu starten.

# Fehlerbehebung (Mac)

- Sollten Sie den folgenden Fehler während des letzten Schrittes sehen

```
zsh: no matches found: seaborn[stats]
```

- Probieren Sie den Befehl mit Anführungszeichen um `seaborn[stats]` herum wie folgt aus:

```
python3.11 -m pip install numpy matplotlib pandas jupyterlab  
timetable openpyxl 'seaborn[stats]'
```

- Das liegt daran, dass die `zsh` Shell eckige Klammern für Globbing/Mustervergleiche verwendet.

## 6. Installieren von Python & JupyterLab (Ubuntu 20.04 & 22.04)

---

- Sollten Sie Linux verwenden gehen wir davon aus, dass Sie bereit sind Ihre eigene Zeit zu investieren um alles zum Laufen zu bringen.
- Da auf Ihr Betriebssystem sehr wahrscheinlich bereits eine Version von Python installiert ist, raten wir Ihnen ein Virtual Environment mit `conda`, `venv`, oder `pipenv` zu verwenden um Konflikte zu vermeiden.
- Auf den folgende Folien wird beschrieben wie Sie Python 3.11 mit `conda` installieren.

# Schritt 1a: Anaconda herunterladen (Ubuntu)

- Laden Sie die neuste Anaconda Version von <https://www.anaconda.com/> herunter
- Öffnen Sie einen Terminal und navigieren Sie zum Ordner in welchem das File heruntergeladenen wurde (wahrscheinlich der *Downloads* Ordner).
- Das Files welches Sie heruntergeladenen haben sollte `Anaconda3-2023.09-Linux-x86_64.sh` heissen. Je nachdem wann genau Sie dieser Anleitung folgen ist es möglich, dass das File einen leicht anderen Namen hat.

# Schritt 1b: Anaconda installieren (Ubuntu)

- Führen Sie `sudo chmod +x Anaconda3-2023.09-Linux-x86_64.sh` aus um das File ausführbar zum machen.
- Führen Sie `./Anaconda3-2023.09-Linux-x86_64.sh` aus um die Installation zu starten. **Führen Sie diesen Befehl nicht mit sudo aus!**
- Während der Installation werden Sie gefragt ob Sie die Anaconda Distribution initialisieren möchten indem `conda init` ausgeführt wird. Beantworten Sie dies mit **yes**.
- Schliessen Sie den Terminal sobald die Installation abgeschlossen ist.

## Schritt 2: Neues conda Environment (Ubuntu)

- Öffnen Sie einen neuen Terminal.
- Führen Sie `conda create --name inf2 python=3.11` aus. *inf2* ist der Name des erstellten Environments. Sie können einen beliebigen Namen auswählen.
- Führen Sie `conda activate inf2` aus um das Environment zu aktivieren. Die Kommandozeile sollte nun mit (*inf2*) ganz links beginnen.
- Führen Sie `conda install -c conda-forge numpy matplotlib pandas jupyterlab openpyxl seaborn[stats] pip install timetable` aus um die notwendigen Packages zu installieren.
- Schliessen Sie den Terminal.

## Schritt 3: JupyterLab ausführen (Ubuntu)

- Öffnen Sie einen neuen Terminal.
- Führen Sie `conda activate inf2` aus um das Environment zu aktivieren.
- Geben Sie `jupyter lab` ein und drücken Sie "Enter" um JupyterLab zu starten.

## 7. JupyterLab & Notebooks ausführen (Windows/Mac/Ubuntu)

---

# Schritt 1: JupyterLab ausführen

- Geben Sie den folgenden Befehl im Terminal ein und drücken Sie “Enter” um JupyterLab auszuführen.
  - `jupyter lab`, *oder*
  - `jupyter-lab`, *oder*
  - `python -m jupyterlab` (Windows), *oder*
  - `python3.11 -m jupyterlab` (Mac/Ubuntu).
- Damit sollte sich automatisch ein neuer Tab in Ihrem Browser öffnen. Sollte dies nicht der Fall sein, folgen Sie einem der Links die im Terminal ausgegeben wurden.

Obwohl JupyterLab im Web-Browser geöffnet wird benötigt es keine Internetverbindung. JupyterLab läuft lokal auf Ihrem Computer und wird nur im Web-Browser angezeigt.

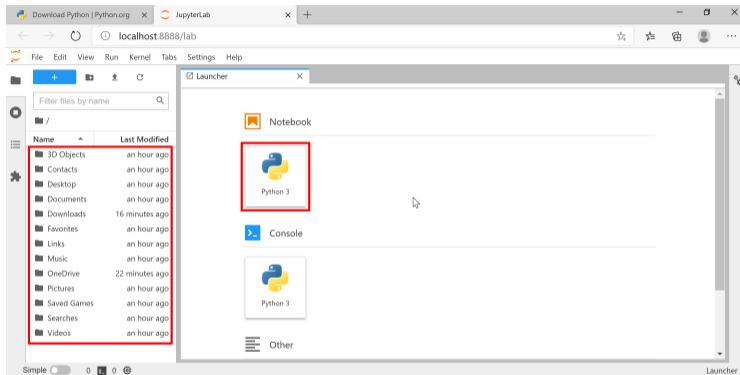
# Schritt 1: JupyterLab ausführen (Win/Mac/Ubuntu)

```
C:\Windows\system32\cmd.exe - jupyter lab

C:\Users\rouge>jupyter lab
[I 2021-02-17 00:03:05.511 ServerApp] jupyterlab | extension was successfully linked.
[W 2021-02-17 00:03:05.525 ServerApp] The 'min_open_files_limit' trait of a ServerApp instance expected an int, not the
NoneType None.
[W 2021-02-17 00:03:05.588 ServerApp] Terminals not available (error was No module named 'winpty.cywintpy')
[I 2021-02-17 00:03:05.588 LabApp] JupyterLab extension loaded from c:\users\rouge\appdata\local\programs\python\python3
9\lib\site-packages\jupyterlab
[I 2021-02-17 00:03:05.588 LabApp] JupyterLab application directory is C:\Users\rouge\AppData\Local\Programs\Python\Pyth
on39\share\jupyter\lab
[I 2021-02-17 00:03:05.588 ServerApp] jupyterlab | extension was successfully loaded.
[I 2021-02-17 00:03:06.666 ServerApp] nbclassic | extension was successfully loaded.
[I 2021-02-17 00:03:06.885 ServerApp] Serving notebooks from local directory: C:\Users\rouge
[I 2021-02-17 00:03:06.917 ServerApp] Jupyter Server 1.3.0 is running at:
[I 2021-02-17 00:03:06.917 ServerApp] http://localhost:8888/lab?token=60619289371e8526e1314207c95ccc17442855735719320f
[I 2021-02-17 00:03:06.917 ServerApp] or http://127.0.0.1:8888/lab?token=60619289371e8526e1314207c95ccc1744285573571932
0f
[I 2021-02-17 00:03:06.917 ServerApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirm
ation).
[C 2021-02-17 00:03:07.182 ServerApp]

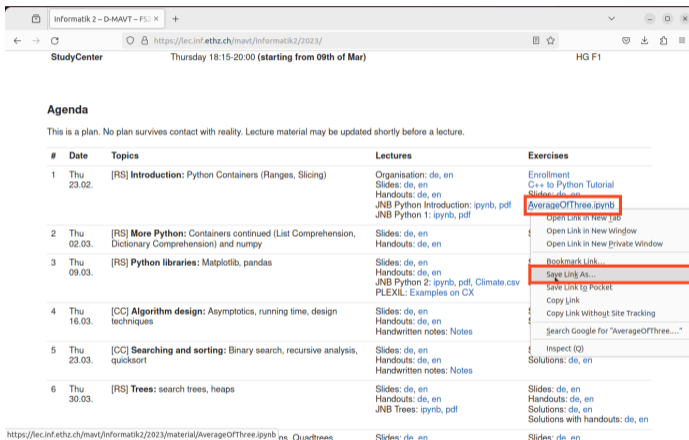
To access the server, open this file in a browser:
file:///C:/Users/rouge/AppData/Roaming/jupyter/runtime/jpserver-2768-open.html
Or copy and paste one of these URLs:
http://localhost:8888/lab?token=60619289371e8526e1314207c95ccc17442855735719320f
or http://127.0.0.1:8888/lab?token=60619289371e8526e1314207c95ccc17442855735719320f
[W 2021-02-17 00:03:17.846 LabApp] Could not determine jupyterlab build status without nodejs
```

# Schritt 2: Notebook erstellen (Win/Mac/Ubuntu)



- Klicken Sie auf die “Python 3” Schaltfläche im “Launcher” Tab um ein neues Python Notebook zu erstellen, oder
- Navigieren Sie zu einem Python Notebook auf Ihrem Computer und öffnen Sie es.

# Schritt 3a: Notebook Download (Win/Mac/Ubuntu)

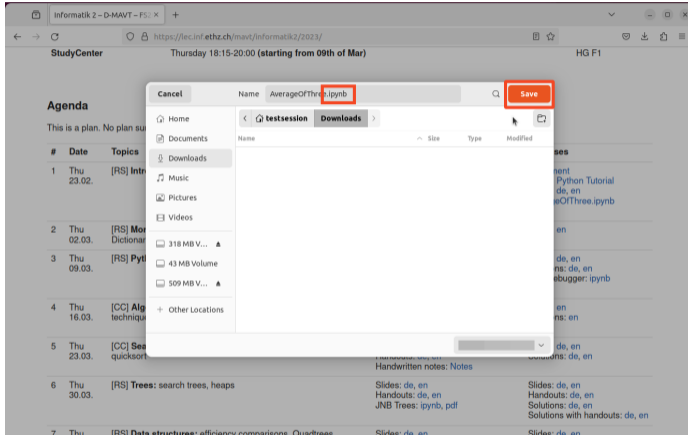


The screenshot shows a web browser window with the URL `https://lec.inf.ethz.ch/mavt/informatik2/2023/`. The page displays an "Agenda" for a course. A table lists several topics, each with associated materials. A context menu is open over the link "AverageOfThree.ipynb" in the first row of the table. The menu options include "Open Link in New Tab", "Open Link in New Window", "Open Link in New Private Window", "Bookmark Link...", "Save Link As..." (highlighted with a red box), "Save Link to Pocket", "Copy Link", "Copy Link Without Site Tracking", "Search Google for 'AverageOfThree....'", "Inspect (Q)", and "Solutions: de, en".

#	Date	Topics	Lectures	Exercises
1	Thu 23.02.	[RS] <b>Introduction:</b> Python Containers (Ranges, Slicing)	Organisation: <a href="#">de</a> , <a href="#">en</a> Slides: <a href="#">de</a> , <a href="#">en</a> Handouts: <a href="#">de</a> , <a href="#">en</a> JNB Python Introduction: <a href="#">ipynb</a> , <a href="#">pdf</a> JNB Python 1: <a href="#">ipynb</a> , <a href="#">pdf</a>	<a href="#">Enrollment</a> <a href="#">C++ to Python Tutorial</a> <a href="#">Slides: de, en</a> <a href="#">AverageOfThree.ipynb</a>
2	Thu 02.03.	[RS] <b>More Python:</b> Containers continued (List Comprehension, Dictionary Comprehension) and numpy	Slides: <a href="#">de</a> , <a href="#">en</a> Handouts: <a href="#">de</a> , <a href="#">en</a>	
3	Thu 09.03.	[RS] <b>Python libraries:</b> Matplotlib, pandas	Slides: <a href="#">de</a> , <a href="#">en</a> Handouts: <a href="#">de</a> , <a href="#">en</a> JNB Python 2: <a href="#">ipynb</a> , <a href="#">pdf</a> , <a href="#">Climate.csv</a> PLEXIL: <a href="#">Examples on CX</a>	
4	Thu 16.03.	[CC] <b>Algorithm design:</b> Asymptotics, running time, design techniques	Slides: <a href="#">de</a> , <a href="#">en</a> Handouts: <a href="#">de</a> , <a href="#">en</a> Handwritten notes: <a href="#">Notes</a>	
5	Thu 23.03.	[CC] <b>Searching and sorting:</b> Binary search, recursive analysis, quicksort	Slides: <a href="#">de</a> , <a href="#">en</a> Handouts: <a href="#">de</a> , <a href="#">en</a> Handwritten notes: <a href="#">Notes</a>	
6	Thu 30.03.	[RS] <b>Trees:</b> search trees, heaps	Slides: <a href="#">de</a> , <a href="#">en</a> Handouts: <a href="#">de</a> , <a href="#">en</a> JNB Trees: <a href="#">ipynb</a> , <a href="#">pdf</a>	Slides: <a href="#">de</a> , <a href="#">en</a> Handouts: <a href="#">de</a> , <a href="#">en</a> Solutions: <a href="#">de</a> , <a href="#">en</a> Solutions with handouts: <a href="#">de</a> , <a href="#">en</a>

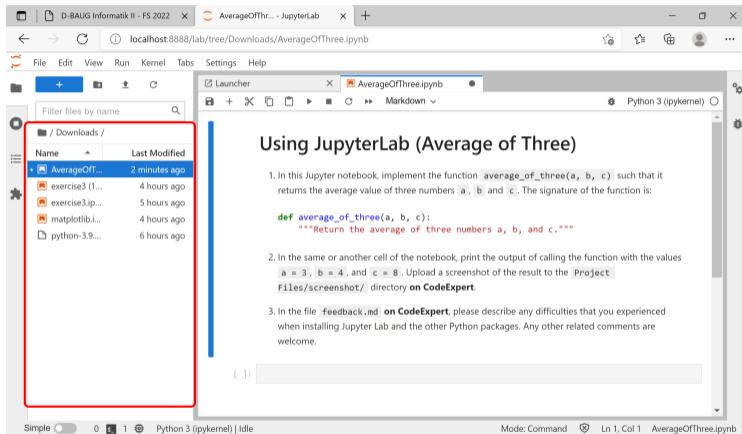
Navigieren Sie zur Kurs Website, klicken Sie das ipynb File mit der rechten Maustaste an und wählen Sie "Speichern unter" (oder etwas ähnliches) aus.

# Schritt 3b: Notebook Download (Win/Mac/Ubuntu)



Stellen Sie sicher, dass das File als '.ipynb' File und nicht als '.txt' File gespeichert wird.

# Schritt 3c: Notebook öffnen (Win/Mac/Ubuntu)



The screenshot shows a web browser window with a JupyterLab interface. The address bar shows the URL `localhost:8888/lab/tree/Downloads/AverageOfThree.ipynb`. The interface is divided into two main sections:

- File Browser (Left):** A sidebar showing a file tree for the `Downloads` directory. A table lists files with their names and last modified times. The file `AverageOfThree.ipynb` is highlighted in blue and enclosed in a red box.
- Notebook (Right):** The main area displays the content of the selected notebook, titled "Using JupyterLab (Average of Three)". It contains three numbered instructions and a code cell.

**File Browser Table:**

Name	Last Modified
AverageOfThree.ipynb	2 minutes ago
exercise3 (1...)	4 hours ago
exercise3.ip...	5 hours ago
matplotlib.i...	4 hours ago
python-3.9...	6 hours ago

**Notebook Content:**

## Using JupyterLab (Average of Three)

- In this Jupyter notebook, implement the function `average_of_three(a, b, c)` such that it returns the average value of three numbers `a`, `b` and `c`. The signature of the function is:  

```
def average_of_three(a, b, c):  
    """Return the average of three numbers a, b, and c."""
```
- In the same or another cell of the notebook, print the output of calling the function with the values `a = 3`, `b = 4`, and `c = 8`. Upload a screenshot of the result to the `Project Files/screenshot/` directory on **CodeExpert**.
- In the file `feedback.md` on **CodeExpert**, please describe any difficulties that you experienced when installing Jupyter Lab and the other Python packages. Any other related comments are welcome.

[ ]:

Navigieren Sie zum Notebook im File Browser auf der linken Seite und öffnen Sie es.

# Schritt 4: JupyterLab beenden (Win/Mac/Ubuntu)

```
C:\Windows\system32\cmd.exe
[W 2021-02-17 00:03:05.588 ServerApp] Terminals not available (error was No module named 'winpty.cywintpy')
[I 2021-02-17 00:03:05.588 LabApp] JupyterLab extension loaded from c:\users\rouge\appdata\local\programs\python\python3
9\lib\site-packages\jupyterlab
[I 2021-02-17 00:03:05.588 LabApp] JupyterLab application directory is C:\Users\rouge\AppData\Local\Programs\Python\Pyth
on39\share\jupyterlab
[I 2021-02-17 00:03:05.588 ServerApp] jupyterlab | extension was successfully loaded.
[I 2021-02-17 00:03:06.666 ServerApp] nbclassic | extension was successfully loaded.
[I 2021-02-17 00:03:06.885 ServerApp] Serving notebooks from local directory: C:\Users\rouge
[I 2021-02-17 00:03:06.917 ServerApp] Jupyter Server 1.3.0 is running at:
[I 2021-02-17 00:03:06.917 ServerApp] http://localhost:8888/lab?token=60619289371e8526e1314207c95ccc17442855735719320f
or http://127.0.0.1:8888/lab?token=60619289371e8526e1314207c95ccc17442855735719320f
[I 2021-02-17 00:03:06.917 ServerApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirm
ation).
[C 2021-02-17 00:03:07.182 ServerApp]

To access the server, open this file in a browser:
file:///C:/Users/rouge/AppData/Roaming/jupyter/runtime/jpserver-2768-open.html
Or copy and paste one of these URLs:
http://localhost:8888/lab?token=60619289371e8526e1314207c95ccc17442855735719320f
or http://127.0.0.1:8888/lab?token=60619289371e8526e1314207c95ccc17442855735719320f
[W 2021-02-17 00:03:17.846 LabApp] Could not determine jupyterlab build status without nodejs
[I 2021-02-17 00:04:17.900 ServerApp] Kernel started: 8a951803-1d5c-4972-b42b-6cbcafbd42
[I 2021-02-17 00:04:38.519 ServerApp] Starting buffering for 8a951803-1d5c-4972-b42b-6cbcafbd42:b020664b-9df7-4997-a73
2-3dc4767f64e7
[I 2021-02-17 00:04:45.256 ServerApp] Interrupted...
[I 2021-02-17 00:04:45.256 ServerApp] Shutting down 1 kernel
[I 2021-02-17 00:04:45.656 ServerApp] Kernel shutdown: 8a951803-1d5c-4972-b42b-6cbcafbd42
C:\Users\rouge>exit
```

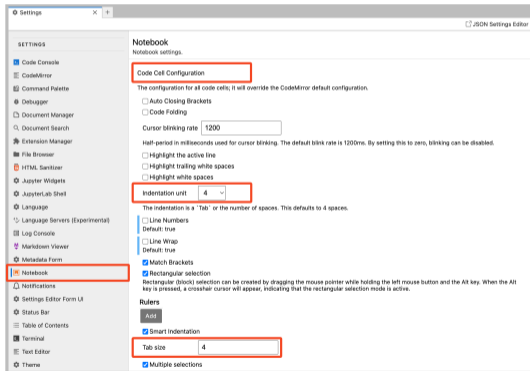
- Drücken Sie <Ctrl>+C auf Ihrer Tastatur (<Strg>+C auf Deutschen Tastaturen) um JupyterLab zu beenden.
- Führen Sie `exit` aus um den Command Prompt oder Terminal zu schliessen.

# Hinweis: Einrückung (Win/Mac/Ubuntu)

Wie Sie in diesem Kurs lernen werden ist Einrückung wichtig in Python. Üblicherweise wird davon ausgegangen, dass ein *Tab* gleich 4 Leerzeichen ist. CodeExpert hält sich nicht an diese Konvention und setzt ein *Tab* mit 2 Leerzeichen gleich. Dadurch entstehen Probleme wenn Code zwischen CodeExpert und JupyterLab hin und her kopiert wird.

# Hinweis: Einrückung (Win/Mac/Ubuntu)

Um JupyterLab mit CodeExpert kompatibel zu machen navigieren Sie zu *Settings* -> *Settings Editor* -> *Notebook* -> *Code Cell Configuration* und wählen Sie **Tab Size: 2** und **Indentation unit: 2** aus.



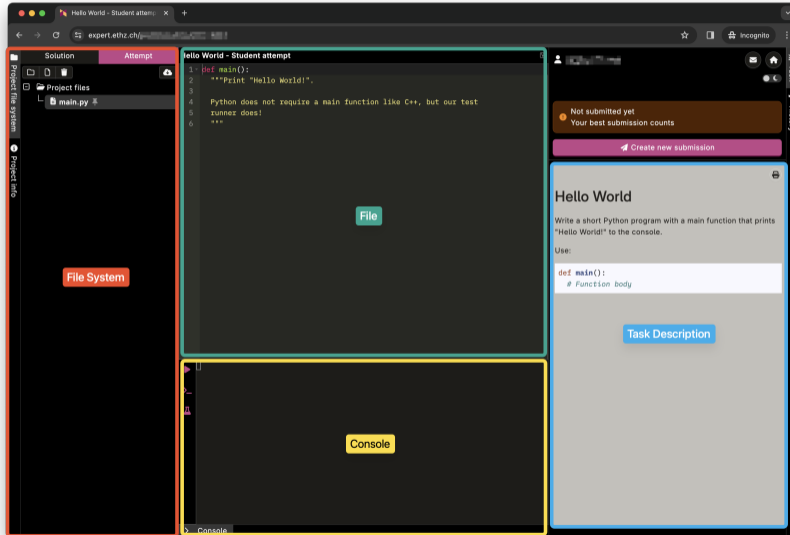
**Hinweis:** Sollte JupyterLab Teile Ihres Codes Rot markieren könnte dies ein Hinweis auf einen Einrückungsfehler sein.

# Hinweis: Shorcuts (Win/Mac/Ubuntu)

Hier einige nützliche shortcuts für JupyterLab:

- **CTRL + Z** Um Ihre letzte Aktion rückgängig zu machen
- **CTRL + SHIFT + Z** Um Ihre letzte rückgängig gemachte Aktion zu wiederholen
- **CTRL + F** Um nach Text zu suchen
- **CTRL + /** Um Zeilen zu kommentieren (oder unkommentieren)
- **SHIFT + ENTER** Um die ausgewählte Zelle auszuführen - wenn keine Zelle unterhalb vorhanden ist wird eine neue eingefügt.
- **ENTER** Um den Editiermodus der ausgewählten Zelle zu aktivieren
- **ESC** Um den Editiermodus der ausgewählten Zelle zu deaktivieren
- **A** Um eine neue Zelle oberhalb einzufügen
- **B** Um eine neue Zelle unterhalb einzufügen

# 8. Code Expert



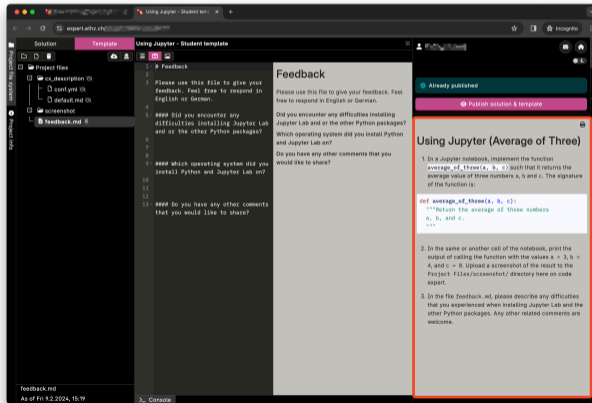
Um auf CodeExpert zuzugreifen navigieren Sie zu:

<https://expert.ethz.ch/>.

Sie müssen dazu nichts installieren. Loggen Sie sich lediglich mit Ihrem edu-ID Konto ein, welches Sie mit ihrem ETH Konto verbunden haben, siehe Webseite für Details.

# Übung: *Using Jupyter*

Gehen Sie auf Code Expert und die Übung “Using Jupyter” in “Exercise 1: Python I” lösen. Sie können das auf der Vorlesungswebsite bereitgestellte JupyterLab Notebook verwenden. Alternativ können Sie ein neues Notebook erstellen.



# CodeExpert Interface

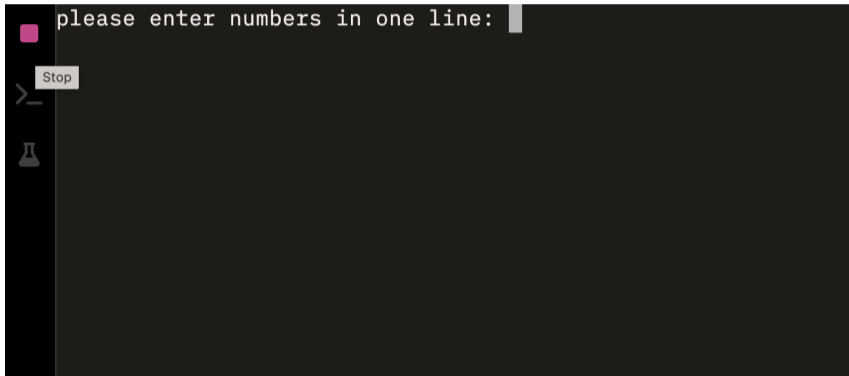
Jede CodeExpert Übung hat die selbe Struktur:

- **File System:** Zeigt die verfügbaren Files im Projekt.
- **File:** Bearbeiten des ausgewählten Files.
- **Task Description:** Zeigt die Anweisungen und Hinweise für die Übung.
- **Console:** Ausführen und Testen des Projekts. Mit der Play Schaltfläche können Sie das Projekt manuel ausführen und testen. Mit der Flaschen Schaltfläche wird das Projekt mit vordefinierten Tests überprüft.
- **Create new submission:** Drücken Sie diesen Knopf sobald Sie mit Ihrer Lösung zufrieden sind. Sie können sooft sie möchten eine neue Lösung einreichen. Nur die beste Abgabe zählt.

# CodeExpert Run



# CodeExpert Run



A screenshot of a CodeExpert IDE interface. The main area is a dark terminal window with the text "please enter numbers in one line:" followed by a white cursor. On the left sidebar, there is a pink square icon at the top, a "Stop" button, a ">\_" icon, and a flask icon.

# CodeExpert Test

```
▶ empty line : passed
single : passed
>_ pair : passed
🧪 mixed : passed
short : passed
Test longer : passed
rted : passed
Tests result: passed 7 of 7 / score 70 % [████████████████████]
]
```

# CodeExpert Interactive Shell



# CodeExpert Interaktive Konsole

```
▶ Welcome to the interactive Python shell.  
Your work here will be lost! The process runs for at most  
10 minutes.  
■  
⏱ Type "help", "copyright", "credits" or "license" for more  
information.  
>>>
```

# CodeExpert Main Funktion

**Wichtig:** Aufgrund des Aufbaus von CodeExpert wird immer eine main Funktion in Ihrem Code benötigt. Diese Funktion wird aufgerufen wenn Ihr Programm gestartet oder getestet wird.

# Übung: *Hello World*

Probieren Sie es aus!

Öffnen Sie das “Hello World” Code Beispiel in “Exercise 1: Python I”.  
Modifizieren Sie die main Funktion so, dass der beschriebene Text  
ausgegeben wird.

Drücken Sie auf die Test Schaltfläche und reichen Sie Ihre Lösung ein.

## 9. Terminal

---

Hier sehen Sie wie Sie schnell und einfach Python Code im Terminal ausführen.

# Navigieren

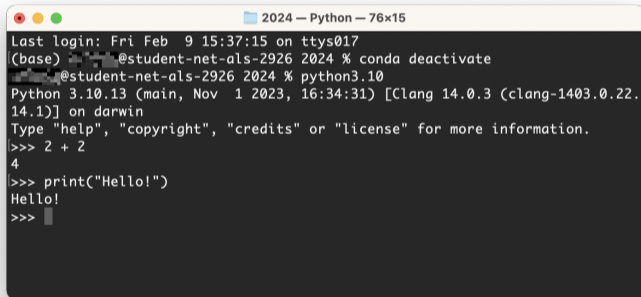
Öffnen Sie einen Terminal und navigieren Sie zu Ihrem Ordner. Verwenden Sie dazu die folgenden Befehle:

- Geben Sie `ls` oder `dir` ein um den Inhalt des momentanen Ordners anzuzeigen
- Geben Sie `cd` gefolgt vom Namen eines Ordners ein um in den genannten Ordner zu wechseln
- Drücken Sie `<TAB>` um File- und Ordnernames automatisch zu vervollständigen

Probieren Sie es aus!

# Python Starten

Geben Sie `python3.11` ein um Python zu starten. Versuchen Sie nun einige simple Python operationen im Terminal auszuführen.

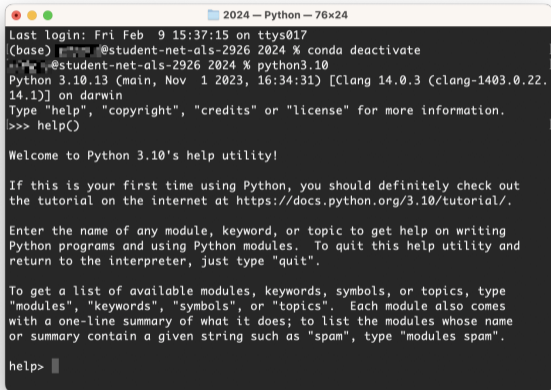
A terminal window titled "2024 - Python - 76x15" showing the process of starting Python. The terminal output includes the login message, the command to deactivate conda, the command to start python3.10, the Python version and platform information, and the execution of simple arithmetic and print statements.

```
2024 - Python - 76x15
Last login: Fri Feb  9 15:37:15 on ttys017
(base) [redacted]@student-net-als-2926 2024 % conda deactivate
[redacted]@student-net-als-2926 2024 % python3.10
Python 3.10.13 (main, Nov  1 2023, 16:34:31) [Clang 14.0.3 (clang-1403.0.22.14.1)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> 2 + 2
4
>>> print("Hello!")
Hello!
>>> █
```

Mit `<CTRL>+d` können Sie Python beenden.

# Hilfe

Geben Sie `help()` oder `help(<name>)` ein um Hilfe zu bekommen.

A terminal window titled "2024 - Python - 76x24" showing the execution of the Python help utility. The user has deactivated conda and started Python 3.10. The terminal displays the standard Python help text, including a welcome message, a link to the Python tutorial, and instructions on how to use the help utility. The prompt is currently at "help>".

```
2024 - Python - 76x24
Last login: Fri Feb  9 15:37:15 on ttys017
(base) [REDACTED]@student-net-als-2926 2024 % conda deactivate
[REDACTED]@student-net-als-2926 2024 % python3.10
Python 3.10.13 (main, Nov  1 2023, 16:34:31) [Clang 14.0.3 (clang-1403.0.22.14.1)] on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>> help()

Welcome to Python 3.10's help utility!

If this is your first time using Python, you should definitely check out
the tutorial on the internet at https://docs.python.org/3.10/tutorial/.

Enter the name of any module, keyword, or topic to get help on writing
Python programs and using Python modules.  To quit this help utility and
return to the interpreter, just type "quit".

To get a list of available modules, keywords, symbols, or topics, type
"modules", "keywords", "symbols", or "topics".  Each module also comes
with a one-line summary of what it does; to list the modules whose name
or summary contain a given string such as "spam", type "modules spam".

help> █
```

Geben Sie `q` ein um die Hilfe zu beenden.

# Dir

Benutzen Sie `help(<object>)` oder `dir(<object>)`, um die verschiedenen Attribute eines bestimmten Objekts zu erhalten.

```
[>>> family = ["Vedran", "Nina", "Marko"]
]>>> dir(family)
['_add__', '__class__', '__class_getitem__', '__contains__', '__delattr__', '__delitem__', '__dir__', '__doc__', '__eq__', '__format__', '__ge__', '__getattr__', '__getitem__', '__gt__', '__hash__', '__iadd__', '__imul__', '__init__', '__init_subclass__', '__iter__', '__le__', '__len__', '__lt__', '__mul__', '__ne__', '__new__', '__reduce__', '__reduce_ex__', '__repr__', '__reversed__', '__rmul__', '__setattr__', '__setitem__', '__sizeof__', '__str__', '__subclasshook__', 'append', 'clear', 'copy', 'count', 'extend', 'index', 'insert', 'pop', 'remove', 'reverse', 'sort']
```

Probieren Sie es aus!

# Letzter Wert

Benutzen Sie `_` um den letzten Wert aufzurufen.

```
[>>> 3 + 4  
7  
[>>> x = _  
[>>> x  
7
```

Probieren Sie es aus!

# Kopieren und Einfügen im Terminal

Im Terminal wird `<CTRL>+c` verwendet um Prozesse zu beenden. Daher kann dieser Shortcut nicht zum Kopieren benutzt werden. Alternativ können Sie `<CTRL>+<SHIFT>+c` oder `Rechtsklick -> Kopieren` verwenden. Zum Einfügen können Sie `<CTRL>+<SHIFT>+v` oder `Rechtsklick -> Einfügen` verwenden.

## 10. Troubleshooting & FAQ

---

Einige zusätzliche Problemlösungen bei der Installation oder Ausführung von Python in JupyterLab oder im Terminal.

# Problem: Kein Schreibzugriff in cmd unter Windows

**Problem:** die Eingabeaufforderung öffnet sich in einem vom Administrator geschützten Pfad wie `C:/Program Files/` oder `C:/Windows` ohne Schreibzugriff.

**Mögliche Ursache:** Die Installation der CAD-Software NX aus einem früheren Kurs hat möglicherweise eine Terminal-Verknüpfung erstellt, die sich in einem geschützten Systemordnerpfad ohne Schreibzugriff öffnet. Diese Terminal-Verknüpfung taucht oft als erstes auf, wenn man in der Windows-Suche nach "Terminal" sucht.

## Lösungen:

- Navigieren Sie zu einem Ordner mit Schreibzugriff (z. B. dem Benutzerordner), indem Sie Folgendes ausführen: `cd %USERPROFILE%` oder `cd ~` (für Powershell)
- Öffnen Sie erneut das "korrekte" Terminal aus dem Startmenü (oder Windows-Taste + R und geben Sie "cmd" ein)