

## Efl Data Science Kurse - Installationsanleitungen

Hallo lieber Kursteilnehmer!

Schön, dass du dich entschlossen hast, in unseren Kursen etwas über Data Science mit Python zu lernen. Damit du eine optimale Kurserfahrung genießen kannst, sollten allerdings ein paar technische Grundvoraussetzungen auf deinem Privatlaptop geschaffen werden. Dazu zählen:

- Eine funktionierende **Python 3.7+** Installation
- Eine Installation der Entwicklungsumgebung **Spyder**, sowie der dazugehörenden Paketmanagement-Software **Anaconda**
- Verschiedene **Python-Paketinstallationen (z.B. Pandas, Numpy, Keras)**, welche innerhalb deiner Python Entwicklungsumgebung nutzbar sind und dir das Programmieren erleichtern.

Diese Voraussetzungen ermöglichen dir ein rundes Python-Programmiererlebnis! Dann schaffen wir mal diese Voraussetzungen.

### Inhalt

Efl Data Science Kurse - Installationsanleitungen.....	1
Betriebssystem .....	2
Windows-10: .....	2
Mac OS: .....	2
Python Installation: .....	3
Windows 10: .....	3
Mac OS: .....	3
Installation von Spyder und Anaconda .....	4
Windows 10 / MacOS: .....	4
Python/ Anaconda Paketinstallationen .....	5
Keine Installationsmöglichkeit vorhanden – Google Colab .....	8
Weitere Vorbereitungsmöglichkeiten .....	8

## Betriebssystem

Du hast entweder ein Windows-Laptop, oder Apple Macbook?

### Windows-10:

Stelle sicher, dass du ein Windows 64-Bit Betriebssystem hast. Dies sollte zwar Standardmäßig mittlerweile so sein, es kommt jedoch vor, dass noch vereinzelt 32-bit Versionen laufen. Prüfen kannst du das so:

[https://www.youtube.com/watch?v=E\\_rlxF9aDLw](https://www.youtube.com/watch?v=E_rlxF9aDLw)

Falls du eine 32-bit version hast, kannst du nur 32-bit Software installieren. Bitte dies bei der Auswahl der Spyder IDE, sowie Python Installation beachten!

**ACHTUNG:** Für Tag 3 (Intro to Data Science) wird leider eine 64-bit Installation vorausgesetzt! Dies sollte für den Großteil aller Notebooks kein Problem darstellen, dennoch bitten wir darum, dies zu überprüfen.

### Mac OS:

Big Suprise: Mac User haben es tendenziell einfacher. Eine funktionierende, möglichst aktuelle MacOS-Version genügt.

## Python Installation:

Bitte folge einem der folgenden Links für dein jeweiliges Betriebssystem. Das gesamte Vorgehen wird dabei Schritt-für-Schritt erklärt.

Windows 10:

<https://www.youtube.com/watch?v=qridM5Oox3c>

Mac OS:

<https://www.youtube.com/watch?v=0hGzGdRQeak>

## Installation von Spyder und Anaconda

Ein wichtiges Tool für Data Scientists, sowie im allgemeinen Programmierer sind die Entwicklungsumgebungen, in denen man programmiert. Hierzu zählt **Spyder**. **Anaconda** ist der dazugehörige **Software-Paketmanager**, mit dem man bestimmte zusätzliche Software-Funktionalitäten in seine Python-Installation integrieren kann. Diese Pakete ermöglichen damit eine leichtere Programmierung komplexer Funktionalitäten. Bitte folge den folgenden Links, um Spyder und Anaconda zu installieren!

### Windows 10 / MacOS:

Zunächst kannst du dich über Spyder und Anaconda hier informieren:

<https://pypi.org/project/spyder/>

Im Anschluss wähle bitte die passende Installation für dein Betriebssystem aus:

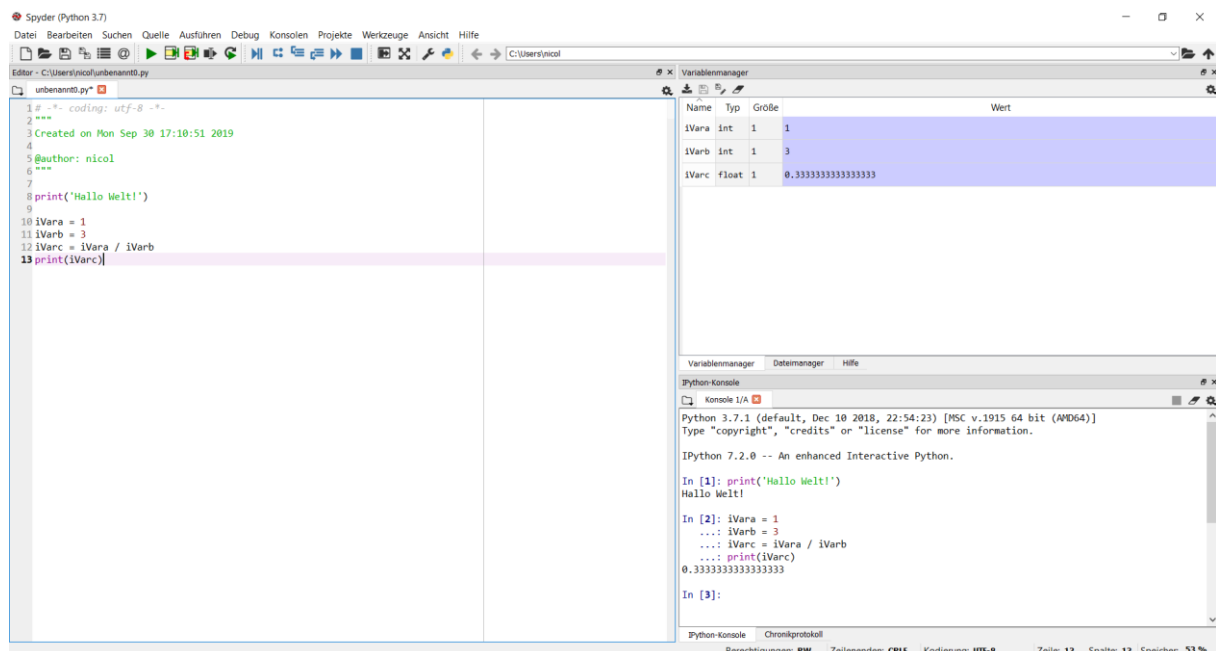
<https://www.anaconda.com/distribution/#download-section>

Für Windows 10 32-bit → <https://repo.anaconda.com/archive/Anaconda3-2019.07-Windows-x86.exe>

Für Windows 10 64-bit → [https://repo.anaconda.com/archive/Anaconda3-2019.07-Windows-x86\\_64.exe](https://repo.anaconda.com/archive/Anaconda3-2019.07-Windows-x86_64.exe)

Für MacOS → [https://repo.anaconda.com/archive/Anaconda3-2019.07-MacOSX-x86\\_64.pkg](https://repo.anaconda.com/archive/Anaconda3-2019.07-MacOSX-x86_64.pkg)

Wenn alles bis hierhin richtig installiert wurde, solltest du ohne Probleme Spyder öffnen können und programmieren können. Das Ganze könnte dann so aussehen:

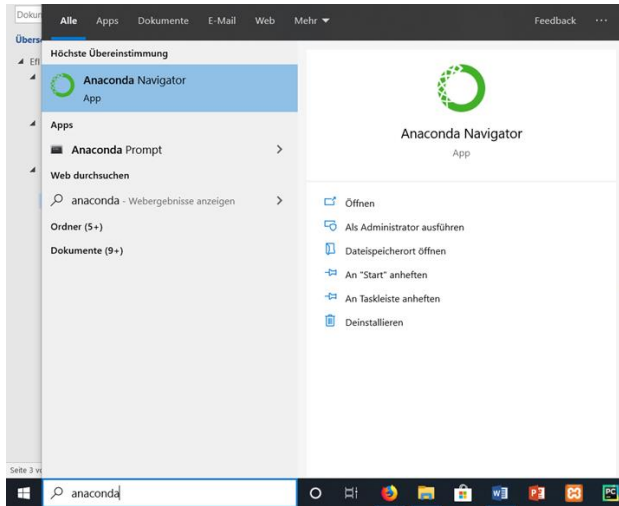


Super! Du kannst ja schonmal ein bisschen in Python rechnen oder dich anderweitig ausprobieren...

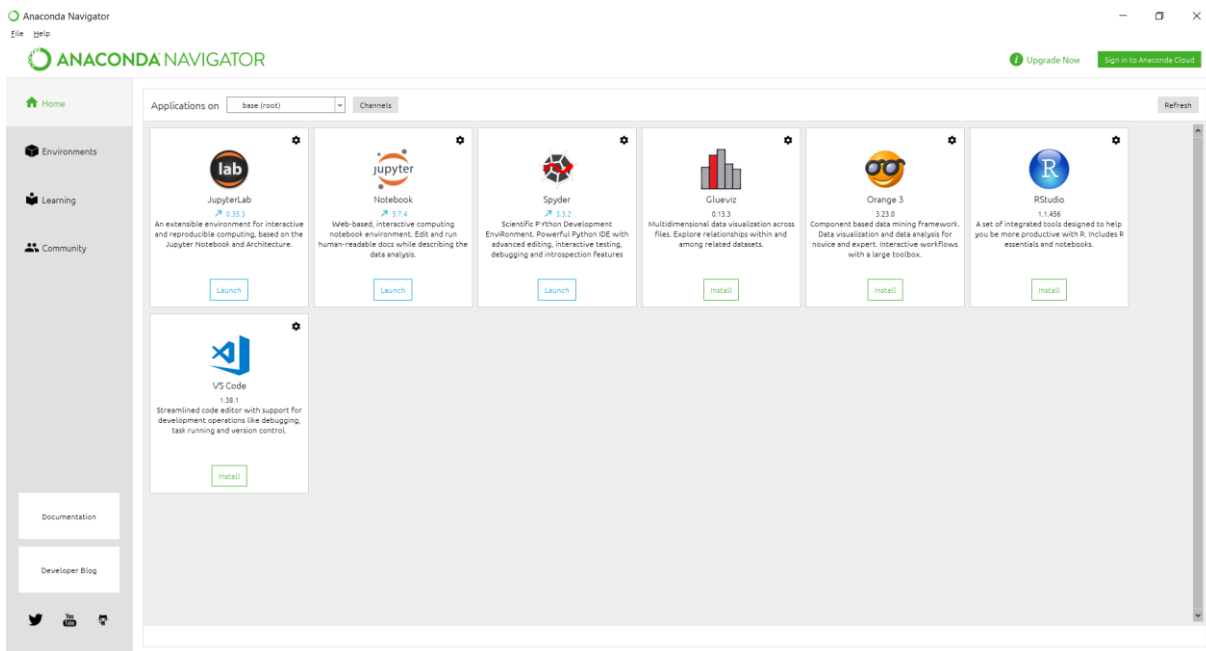
## Python/ Anaconda Paketinstallationen

Python-Pakete lassen sich relativ einfach in der conda Umgebung installieren, sodass sie dann in Spyder (eurer Programmierumgebung) verfügbar gemacht werden.

Hierzu öffne bitte zunächst den Anaconda Navigator.



Nach dem öffnen des Anaconda Navigator findest du eine Nutzeroberfläche.



Hier wählst du bitte „**Environments**“ aus. Du befindest dich standardmäßig im root Environment. Dies ist die Umgebung, in welcher deine Python-Pakete zu finden sind und welche standardmäßig für Projekte in Spyder genutzt werden.

Anaconda Navigator

File Help

ANACONDA NAVIGATOR

Upgrade Now Sign in to Anaconda Cloud

Home

Environments

Learning

Community

Documentation

Developer Blog

Create Clone Import Remove

Search Environments

base (root)

Not installed Channels Update index... Search Packages

Name	Description	Version
<input type="checkbox"/> _anaconda_depends		5.3.1
<input type="checkbox"/> _libarchive_static...		3.3.3
<input type="checkbox"/> _libgcc_mutex		0.1
<input type="checkbox"/> _low_priority		1.0
<input type="checkbox"/> _mkl_mxnet	Mklx package to pin a variant of mxnet conda package	0.0.40
<input type="checkbox"/> _py-igboost-mutex		2.0
<input type="checkbox"/> _pytorch_select		1.2.0
<input type="checkbox"/> _r-mutex		1.0.0
<input type="checkbox"/> _r-igboost-mutex		2.0
<input type="checkbox"/> _tflow_1100_select		0.0.3
<input type="checkbox"/> _tflow_190_select		0.0.2
<input type="checkbox"/> _tflow_select		2.3.0
<input type="checkbox"/> abseil-py	abseil python common libraries, see https://github.com/abseil/abseil-py.	0.7.1
<input type="checkbox"/> enum	advanced enumerations (compatible with python's stdlib enum), namedtuples.	2.2.1
<input type="checkbox"/> affine	Matrices describing affine transformation of the plane.	2.2.2
<input type="checkbox"/> agate	a data analysis library that is optimized for humans instead of machines.	1.6.1
<input type="checkbox"/> agate-dbf	agate-dbf adds read support for dbf files to agate.	0.2.1
<input type="checkbox"/> agate-excel	agate-excel adds read support for excel files (xls and xlsx) to agate.	0.2.3
<input type="checkbox"/> anaconda-nod	anaconda-nod adds nod2pandaslike support to anaconda	0.1.4

5680 packages available

Als nächstes selektierst du im rechten Bereich die „**not installed**“ packages. Hier werden alle Pakete aufgelistet, welche du noch nicht installiert hast und noch installieren kannst.

Jetzt sollst du folgende Pakete installieren:

- Tensorflow

<input type="checkbox"/> tensorflow	Tensorflow is a machine learning library.	1.9.0
-------------------------------------	---	-------

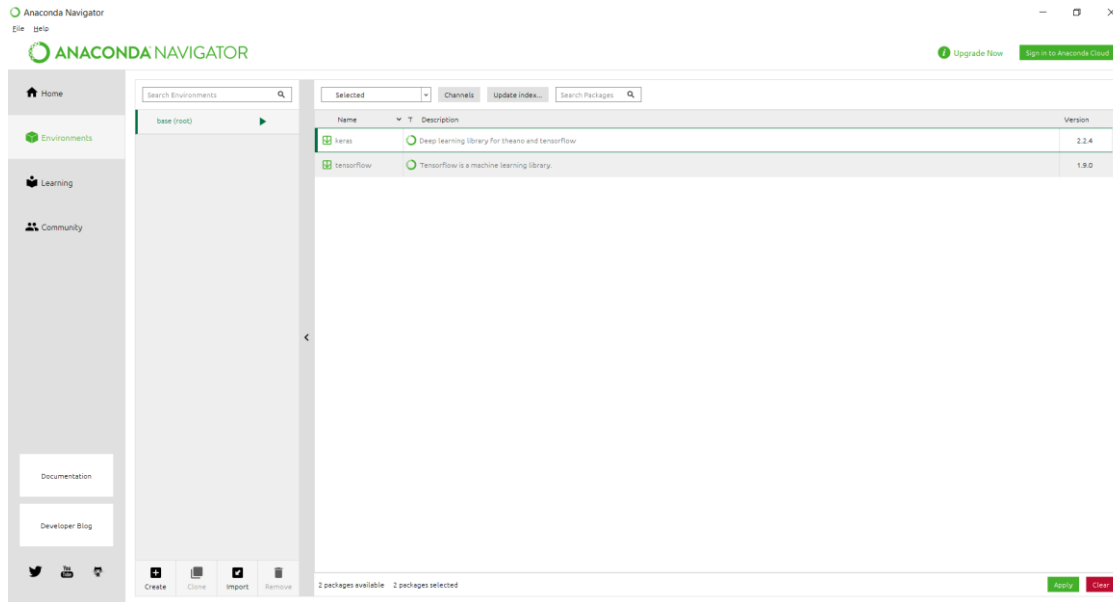
- Keras

<input type="checkbox"/> keras	Deep learning library for theano and tensorflow	2.2.4
--------------------------------	---	-------

- Plotly

<input checked="" type="checkbox"/> plotly	An interactive, browser-based graphing library for python	4.1.1
--	---	-------

Hierzu gibst du den Suchbegriff in die Suchleiste ein und wählst die Pakete über die Checkboxes aus.



Abschließend klickst du auf **apply**. Dann sollten die Packages installiert werden.

Wer sich dazu nochmal ein Video ansehen will, findet es hier:

<https://www.youtube.com/watch?v=V9cDjjRXS08>

**ACHTUNG:** Wer eine 32-bit Windows 10 Version haben sollte, bei dem wird TensorFlow nicht installiert werden können! Hierzu bieten wir die PCs in unserem PC-Pool als Ausweichmöglichkeit an.

Super! Deine Entwicklungsumgebung sollte jetzt bereit für den Kurs sein.

## Keine Installationsmöglichkeit vorhanden – Google Colab

Falls du keine Möglichkeit hast, Spyder und Anaconda zu installieren, gibt es eine weitere Möglichkeit wie du an unserem Kurs teilnehmen kannst: Google Colab.

Im Prinzip ist Google Colab nicht mehr als eine Online-Entwicklungsumgebung, in der man mit Jupyter Notebooks / Ipython-Notebooks arbeitet (siehe hier: <https://jupyter-notebook.readthedocs.io/en/stable/notebook.html#notebook-user-interface>).

Eine Intro dazu, wie Colab funktioniert, findest du hier:  
<https://colab.research.google.com/notebooks/intro.ipynb#>

Das einzige was du brauchst ist:

- Eine funktionierende Internetverbindung
- Ein Google-Konto
- Du musst dich initial mit dem Google Colab Service verbinden, dann kannst du direkt loslegen.

Ein großer Unterschied zwischen Colab und Spyder ist der fehlende Variableninspektor. Da man jedoch die Zellen in Colab beliebig oft ausführen kann, lässt sich dieses Problem umgehen indem man sich in einer neuen Zelle einfach die gewünschte Variable ausgeben lässt (denn im Prinzip ist der Variableninspektor auch einfach eine GUI, welche alle aktiven Variablen in der aktuellen Umgebung anzeigt).

Ein weiterer kleiner Vorteil ist: die meisten Packages, welche wir verwenden werden sind schon vorinstalliert.

Falls ihr also Google Colab verwenden müsst, sollten diese ersten Schritte ausreichen damit ihr vorbereitet seid. Bitte setzt euch allerdings vor dem Kurs mit der GUI auseinander!

## Weitere Vorbereitungsmöglichkeiten

Folgende Webseiten kannst du dir schon einmal als Desktop-Verknüpfung/Favoriten anlegen, da sie hilfreich sein werden:

- <https://www.sli.do/> - Für Umfragen während des Kurses
- <https://stackoverflow.com/> - Für Coding Beispiele
- <https://docs.python.org/3/tutorial/> - Dokumentation von Python
- <https://pandas.pydata.org/pandas-docs/stable/> - Dokumentation von Pandas
- <https://docs.scipy.org/doc/> - Dokumentation von Numpy

Wir freuen uns auf dich!