

# 1. Anaconda

## 1.1 Anaconda Installation

Siehe Link: <https://www.anaconda.com/products/individual>

Unter Windows oder Linux die 64 bit Version von *Anaconda* installieren und ausführen. Während der Installation der Installationsanleitung folgen, bis zur Fertigstellung.

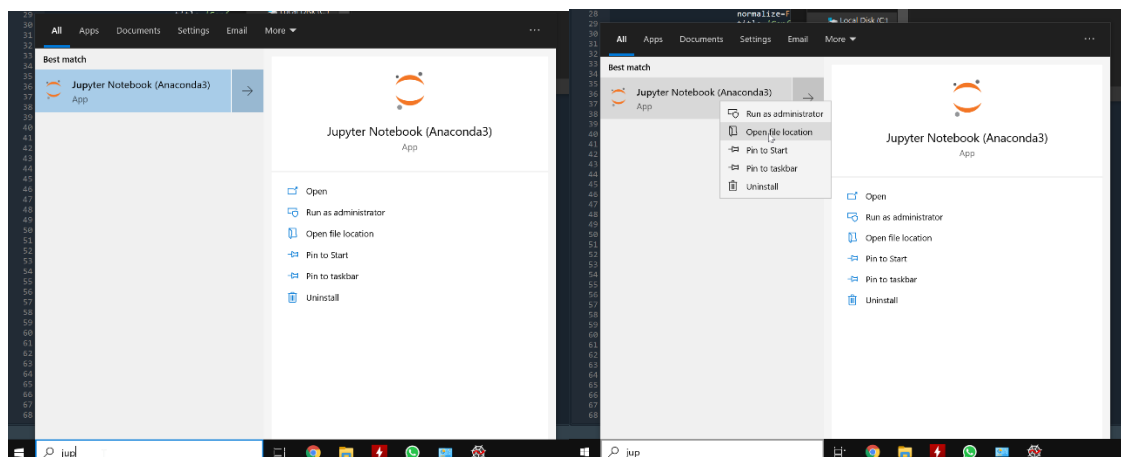


### Installation von Jupyter Notebook:

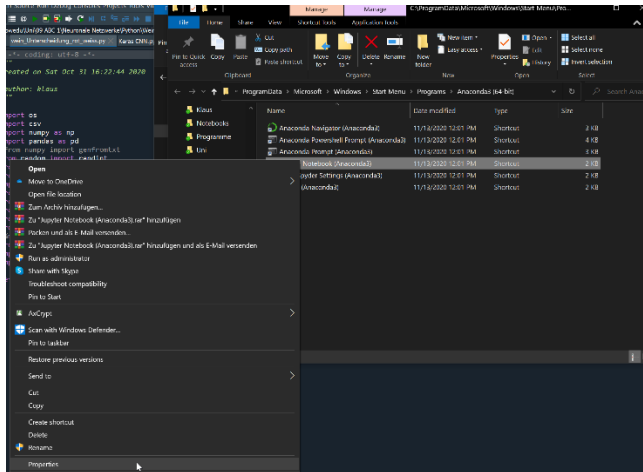
*Jupyter Notebook* wird automatisch als ein Teil von *Anaconda* installiert, allerdings ist die Dateipfad Zuordnung auf den Benutzerordner festgelegt, was ggf. zu Problemen bei langen Dateipfaden führen kann. Wir schlagen wir einen Fix vor.

Ausführung der Zuweisung eines neuen Datenpfades:

Eingabe von *Jupyter Notebook* in der Taskleiste bei Windows, dann rechts auf das Symbol von *Jupyter Notebook* klicken und im Anschluss den exakten Dateipfad öffnen.

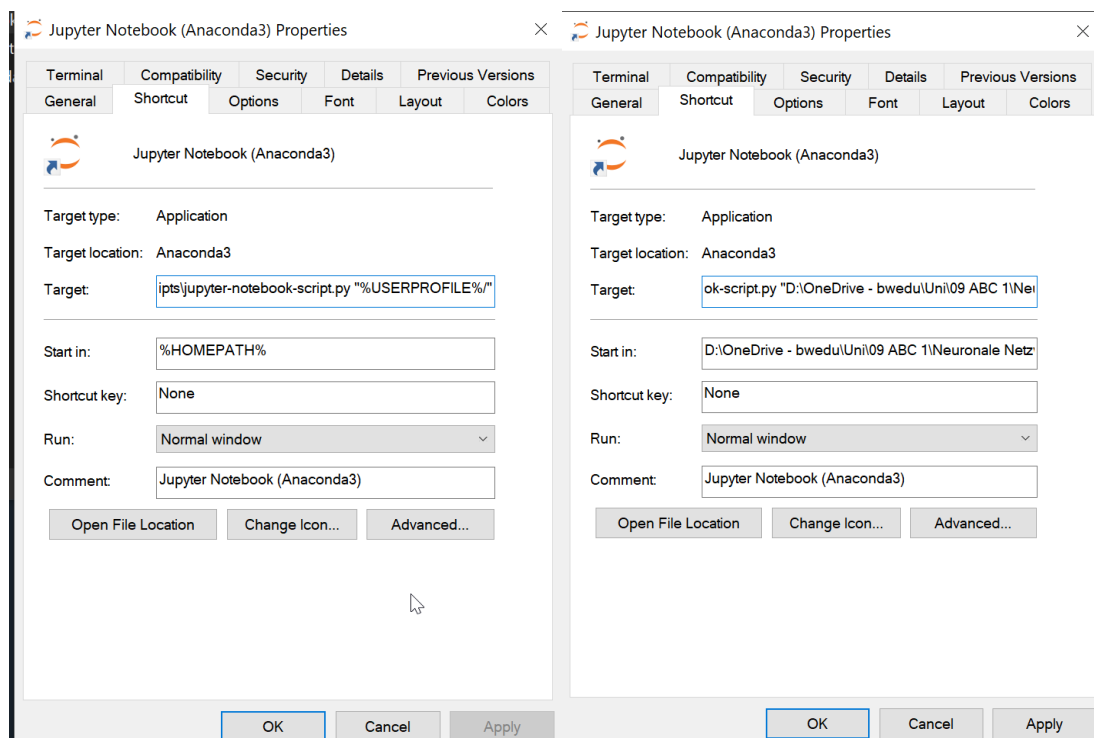


Nachdem sich der Ordner zum Dateipfad geöffnet hat wird die Datei *Jupyter Notebook* [Anaconda] durch einen rechtsklick ausgewählt und das Eigenschaftsmenü aufgerufen.



Unter den Einstellungen wird nun der Reiter Shortcut bzw. Verknüpfung ausgewählt. In diesem Reiter werden jetzt ein neues Target ausgewählt. %USERPROFILE% wird gelöscht und durch das gewünschte Zielverzeichnis ersetzt. Achtung die Anführungszeichen müssen bestehen bleiben!

Eine online Anweisung ist hier zu finden: <https://stackoverflow.com/questions/15680463/change-ipynb-jupyter-notebook-working-directory>

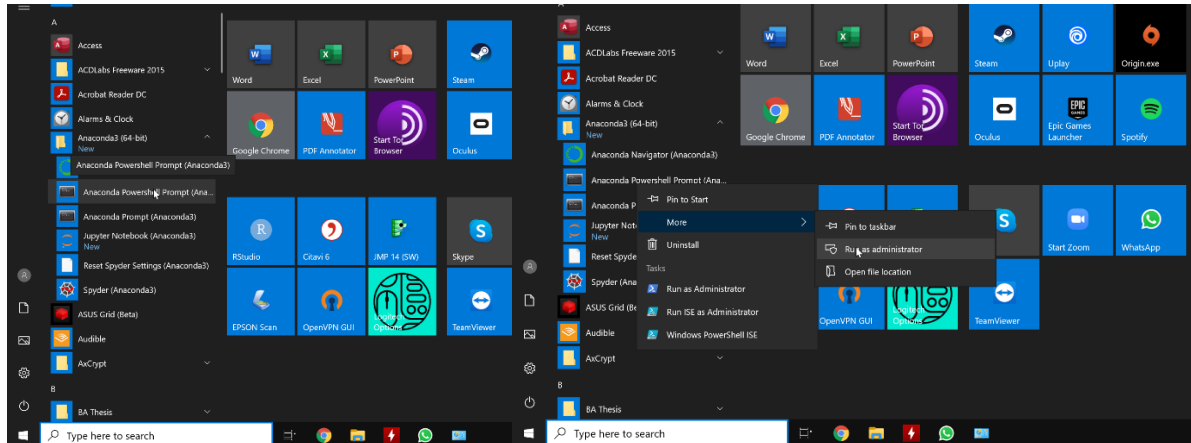


## 2. TensorFlow

### 2.1 Installation von Tensorflow

Als ersten Schritt muss alles was mit *Python* und *Anaconda* in Zusammenhang steht beendet werden, da die Installation sonst nicht erfolgen kann.

Nun kann in der Taskleiste nach *Anaconda Powershell Prompt* gesucht werden. Durch einen Rechtsklick kann diese Anwendung als Administrator ausgeführt werden.



Nach dem Ausführen öffnet sich eine Konsole, in die der folgende Befehl eingegeben wird.

```
pip install tensorflow
```

Dieser Befehl wird über die Eingabe bestätigt und die Installation beginnt. Die Installation ist abgeschlossen, wenn der Maus Cursor wieder blinkt. Im Anschluss kann die Anwendung geschlossen werden.

Link für eine weitere schriftliche und visuelle Anleitung:

<https://deeplizard.com/learn/video/IubEtS2JAIY>

**Installation ist beendet falls keine Grafikkarten support erwünscht ist!**

### 2.2. Installation von CudaCore

Zu beachten vor der Installation von Cuda:

Installation der aktuellen Nvidia Treiber. Es werden keine AMD Grafikkarten unterstützt!

Siehe Link: <https://www.nvidia.de/Download/index.aspx?lang=de>

Überprüfen der aktuellen Versionen von Cuda bezogen auf TensorFlow

Siehe Link: [https://www.tensorflow.org/install/gpu#software\\_requirements](https://www.tensorflow.org/install/gpu#software_requirements)


Installation von Visual Studio, auf kostenloser Download in der Community Version klicken und das Programm ganz normal installieren. Es braucht nur die Basisversion, ohne Zusätze.

Visual Studio schließen, bevor die Installation von Cuda gestartet wird.

Siehe Link: <https://visualstudio.microsoft.com/de/downloads/>

## Downloads

[Auswahlhilfe](#)



### Visual Studio 2019

Version 16.8

[Anmerkungen zu dieser Version >](#)

Umfassend ausgestattete integrierte Entwicklungsumgebung (IDE) für Android, iOS, Windows, Web und Cloud

[Editionen im Vergleich >](#)[Vorgehensweise bei der Offlineinstallation >](#)

### Community

Leistungsstarke IDE für Studierende, Mitwirkende an Open-Source-Projekten und Einzelentwickler

[Kostenloser Download](#)

### Professional

Professionelle IDE, die am besten für kleine Teams geeignet ist

[Kostenlose Testversion](#)

### Enterprise

Skalierbare End-to-End-Lösungen für Teams beliebiger Größe

[Kostenlose Testversion](#)

### Visual Studio Vorschau

[Anmerkungen zu dieser Version >](#)

Early Access auf Features, die noch nicht in der Hauptversion enthalten sind

[Weitere Informationen >](#)

#### 4) Download der korrekten Cuda Version (10.1 Update 2 /August 2019)

Richtige Version wählen Windows 10 Exe local

Fast 2,5 GB Daten müssen heruntergeladen werden. Dies kann unter Umständen dauern.

Siehe Link: <https://developer.nvidia.com/cuda-toolkit-archive>

#### CUDA Toolkit Archive

Previous releases of the CUDA Toolkit, GPU Computing SDK, documentation and developer drivers can be found using the links below, and be sure to check [www.nvidia.com/drivers](http://www.nvidia.com/drivers) for more recent production drivers appropriate for your hardware configuration

[Download Latest CUDA Toolkit](#)

[Learn More about CUDA Toolkit 11](#)

##### Latest Release

CUDA Toolkit 11.1.1 (Oct 2020), [Versioned Online Documentation](#)

##### Archived Releases

CUDA Toolkit 11.1.0 (Sept 2020), [Versioned Online Documentation](#)  
CUDA Toolkit 11.0 Update 1 (Aug 2020), [Versioned Online Documentation](#)  
CUDA Toolkit 11.0 (May 2020), [Versioned Online Documentation](#)  
CUDA Toolkit 10.2 (Nov 2019), [Versioned Online Documentation](#)  
CUDA Toolkit 10.1 update2 (Aug 2019), [Versioned Online Documentation](#)  
CUDA Toolkit 10.1 update1 (May 2019), [Versioned Online Documentation](#)  
CUDA Toolkit 10.1 (Feb 2019), [Online Documentation](#)  
CUDA Toolkit 10.0 (Sept 2018), [Online Documentation](#)  
CUDA Toolkit 9.2 (May 2018), [Online Documentation](#)  
CUDA Toolkit 9.1 (Dec 2017), [Online Documentation](#)  
CUDA Toolkit 9.0 (Sept 2017), [Online Documentation](#)  
CUDA Toolkit 8.0 GA2 (Feb 2017), [Online Documentation](#)  
CUDA Toolkit 8.0 GA1 (Sept 2016), [Online Documentation](#)  
CUDA Toolkit 7.5 (Sept 2015)  
CUDA Toolkit 7.0 (March 2015)  
CUDA Toolkit 6.5 (August 2014)  
CUDA Toolkit 6.0 (April 2014)  
CUDA Toolkit 5.5 (July 2013)  
CUDA Toolkit 5.0 (Oct 2012)  
CUDA Toolkit 4.7 (April 2012)  
CUDA Toolkit 4.1 (Jan 2012)  
CUDA Toolkit 4.0 (May 2011)  
CUDA Toolkit 3.2 (Nov 2010)  
CUDA Toolkit 3.1 (June 2010)  
CUDA Toolkit 3.0 (March 2010)  
OpenCL 1.0 Release (Sept 2009)

#### CUDA Toolkit 10.1 update2 Archive

#### Select Target Platform

Click on the green buttons that describe your target platform. Only supported platforms will be shown.

Operating System	Windows	Linux	Mac OS
Architecture	x86_64		
Version	10	8.1	8
Installer Type	exe-Standalone	exe-Bundled	

#### Download Installer for Windows 10 x86\_64

The base installer is available for download below.

[Base Installer](#)[Download \(2.5 GB\)](#)

Installer Instructions

- Double click cuda\_10.1.242\_420.00\_windows.exe
- Follow on-screen prompts

The checksums for the installer and patches can be found in [Installer Checksums](#).  
For further information, see the [Installation Guide for Microsoft Windows](#) and the [CUDA Quick Start Guide](#).

#### Die Installation von Cuda

Datei ausführen und warten, bis sich das Fenster zum Ausführen öffnet und der Anleitung im Programm Folge leisten.

Installationspfad merken ist nachher noch wichtig

cuDNN wird als zusätzliches Tool benötigt.

Hierfür ist es nötig einen Account zu erstellen und sich anzumelden.

Siehe Link: <https://developer.nvidia.com/cudnn>

Achtung: Auf archivierte Cuda Versionen gehen und die Version Cuda v.7.6.5 for cuda 10.1 verwenden. Falls die falsche Version gewählt wurde, dann wird eine Fehlermeldung in TensorFlow angezeigt.

(Wer eine andere Version verwendet, muss genau suchen welche Version benötigt wird)

Herunterladen der cuDNN library for win 10.

## cuDNN Download

NVIDIA cuDNN is a GPU-accelerated library of primitives for deep neural networks.

☒ I Agree To the Terms of the [cuDNN Software License Agreement](#)

Note: Please refer to the [Installation Guide](#) for release prerequisites, including supported GPU architectures and compute capabilities, before downloading.

For more information, refer to the cuDNN Developer Guide, Installation Guide and Release Notes on the [Deep Learning SDK Documentation](#) web page.

[Download cuDNN v8.0.5 \(November 9th, 2020\), for CUDA 11.1](#)

[Download cuDNN v8.0.5 \(November 9th, 2020\), for CUDA 11.0](#)

[Download cuDNN v8.0.5 \(November 9th, 2020\), for CUDA 10.2](#)

[Download cuDNN v8.0.5 \(November 9th, 2020\), for CUDA 10.1](#)

[Archived cuDNN Releases](#)

Download cuDNN v8.0.3 (August 26th, 2020), for CUDA 11.0
Download cuDNN v8.0.3 (August 26th, 2020), for CUDA 10.2
Download cuDNN v8.0.3 (August 26th, 2020), for CUDA 10.1
Download cuDNN v8.0.2 (July 24th, 2020), for CUDA 11.0
Download cuDNN v8.0.2 (July 24th, 2020), for CUDA 10.2
Download cuDNN v8.0.2 (July 24th, 2020), for CUDA 10.1
Download cuDNN v8.0.1 RC2 (June 26th, 2020), for CUDA 11.0
Download cuDNN v8.0.1 RC2 (June 26th, 2020), for CUDA 10.2
Download cuDNN v7.6.5 (November 18th, 2019), for CUDA 10.2
<a href="#">Download cuDNN v7.6.5 (November 5th, 2019), for CUDA 10.1</a>
Download cuDNN v7.6.5 (November 5th, 2019), for CUDA 10.0
Download cuDNN v7.6.5 (November 5th, 2019), for CUDA 9.2
Download cuDNN v7.6.5 (November 5th, 2019), for CUDA 9.0
Download cuDNN v7.6.4 (September 27, 2019), for CUDA 10.1
Download cuDNN v7.6.4 (September 27, 2019), for CUDA 10.0
Download cuDNN v7.6.4 (September 27, 2019), for CUDA 9.2
Download cuDNN v7.6.4 (September 27, 2019), for CUDA 9.0

### Library for Windows, Mac, Linux, Ubuntu and RedHat/Centos(x86\_64architecture)

cuDNN Library for Windows 7

[cuDNN Library for Windows 10](#)

cuDNN Library for Linux

cuDNN Library for OSX

cuDNN Runtime Library for Ubuntu18.04 (Deb)

cuDNN Developer Library for Ubuntu18.04 (Deb)

cuDNN Code Samples and User Guide for Ubuntu18.04 (Deb)

cuDNN Runtime Library for Ubuntu16.04 (Deb)

cuDNN Developer Library for Ubuntu16.04 (Deb)

cuDNN Code Samples and User Guide for Ubuntu16.04 (Deb)

cuDNN Runtime Library for Ubuntu14.04 (Deb)

cuDNN Developer Library for Ubuntu14.04 (Deb)

cuDNN Code Samples and User Guide for Ubuntu14.04 (Deb)

Nachdem die Datei heruntergeladen wurde kann das ZIP Packet entpackt werden und nach der folgenden Installationsanleitung installiert werden:

Siehe Link: <https://docs.nvidia.com/deeplearning/cudnn/install-guide/index.html#installwindows>

Es muss nun genau darauf geachtet werden, dass die richtigen Daten in die gleichnamigen Ordner kopiert werden, dies muss hier manuell erfolgen.

Die Ordnerstruktur muss exakt mit der in der Anleitung übereinstimmen.

Alles was unter bin steht muss in bin etc. immer mit den Administrator Rechten.

PC > Local Disk (C:) > Users > klaus > Downloads > cudnn-10.1-windows10-x64-v7.6.5.32 > cuda >				
Name	Date modified	Type	Size	
bin	11/13/2020 12:39 PM	File folder		
include	11/13/2020 12:39 PM	File folder		
lib	11/13/2020 12:39 PM	File folder		
NVIDIA_SLA_cuDNN_Support.txt	10/27/2019 8:16 AM	Text Document	39 KB	

So muss es gemacht werden.

Copy <installpath>\cuda\bin\cudnn\*.dll to C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\vx.x\bin.

Copy <installpath>\cuda\include\cudnn\*.h to C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\vx.x\include.

Copy <installpath>\cuda\lib\x64\cudnn\*.lib to C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\vx.x\lib\x64.

Abschließend muss das Programm noch aufgerufen und gestartet werden.

Setzen der folgenden Umgebungsvariablen so, dass sie auf die Stelle zeigen, an der cuDNN installiert wurde. Um auf den Wert der Umgebungsvariablen \$(CUDA\_PATH) zuzugreifen, führt man die folgenden Schritte aus:

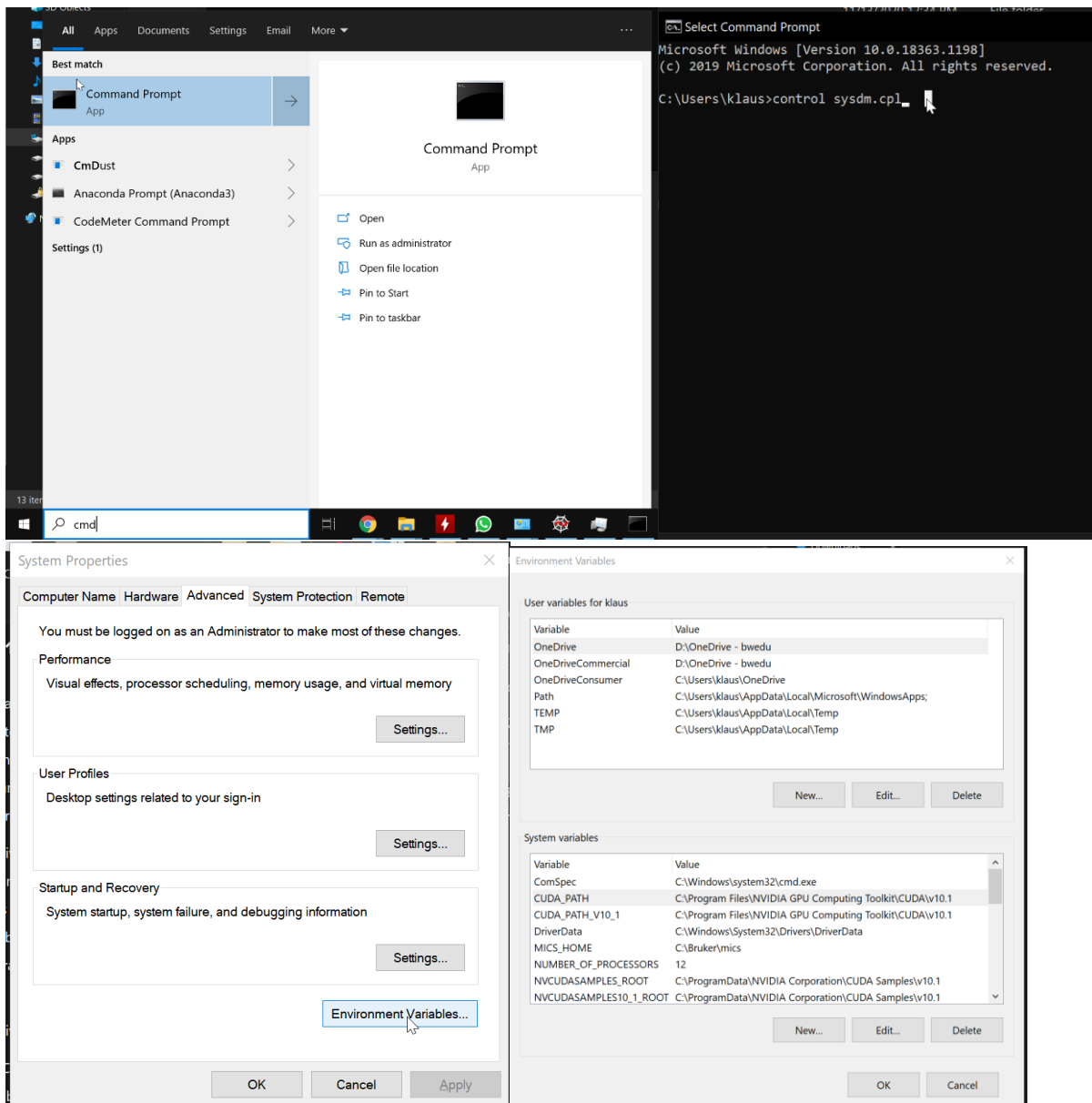
- Öffne über das Startmenü eine Eingabeaufforderung (CMD).
- Gebe run ein und drücken Sie die Eingabetaste.
- Gebe den Befehl control sysdm.cpl ein.
- Wähle die Registerkarte Erweitert im oberen Teil des Fensters.
- Klicke unten im Fenster auf Umgebungsvariablen.
- Stelle sicher, dass die folgenden Werte eingestellt sind:

Name der Variablen: CUDA\_PATH (Sehr wichtig!)

Wert der Variable: C:\Programdateien\NVIDIA GPU Computing Toolkit\CUDA\vx.x

Siehe Link <https://docs.nvidia.com/deeplearning/cudnn/install-guide/index.html#installwindows>>

Nach all diesen Schritten wird ein Neustart empfohlen.





### 3. Keras-tuner

#### 3.1 Installation des Keras-tuner

Um den Keras-tuner zu installieren, muss zunächst die Anaconda Powershell Prompt als Administrator geöffnet werden (Abbildung 1 und Abbildung 2).

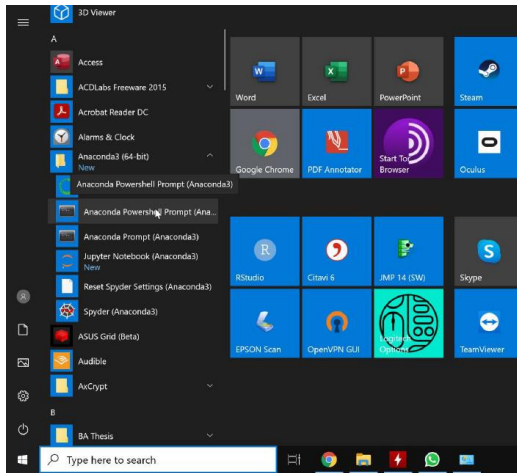


Abbildung 1: Öffnen der Anaconda Powershell Prompt 1

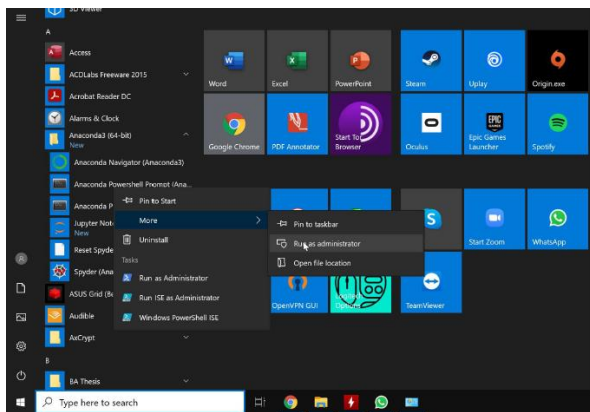
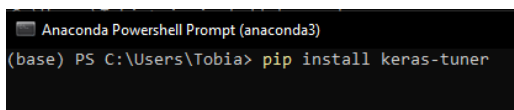


Abbildung 2 Öffnen der Anaconda Powershell Prompt 2

Anschließend den Befehl

*pip install keras-tuner*

eingeben und mit Enter bestätigen.



Der keras-tuner wurde nach Abschließen des Befehls erfolgreich installiert.

## 4. Tensorboard

### 4.1 Installation von Tensorboard

Wichtig: Bevor Tensorboard installiert werden kann muss zuerst Tensorflow installiert werden.

Um Tensorboard zu installieren, muss zunächst die Anaconda Powershell Prompt als Administrator geöffnet werden (Abbildung 1 und Abbildung 2).

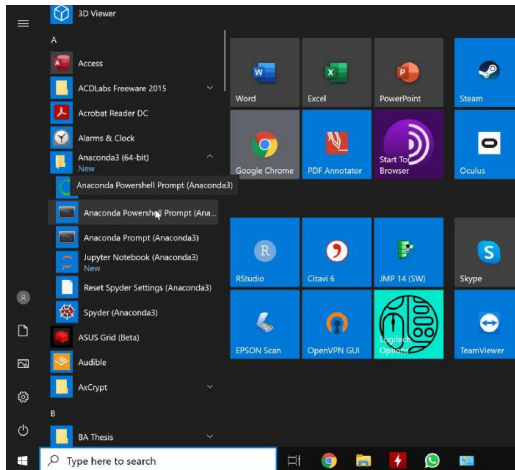


Abbildung 3: Öffnen der Anaconda Powershell Prompt 1

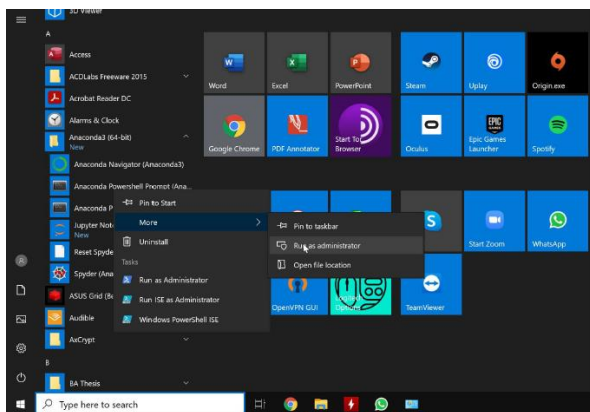


Abbildung 4 Öffnen der Anaconda Powershell Prompt 2

Anschließend müssen folgende Befehle eingegeben werden (jede Zeile einzeln und nacheinander mit Enter bestätigen):

```
conda create -n tf tensorflow
```

```
proceed ([y]/n)? mit y bestätigen
```

```
conda activate tf
```

```
conda install -c conda-forge tensorboard
```

```
proceed ([y]/n)? mit y bestätigen
```

*conda deactivate*

Tensorboard ist nun installiert und kann verwendet werden.

## 4.2 Benutzen von Tensorboard

Merke: Um Tensorboard anwenden zu können muss zuerst das neuronale Netz (mit Callback-Befehl) in Python einmal ausgeführt werden.

Callback-Befehl im neuronalen Netz: „`tensorboard_cb = keras.callbacks.TensorBoard(run_logdir)`“

Um Tensorboard zu benutzen muss wieder die Anaconda Powershell Prompt als Administrator ausgeführt werden (Abbildung 3 und 4).

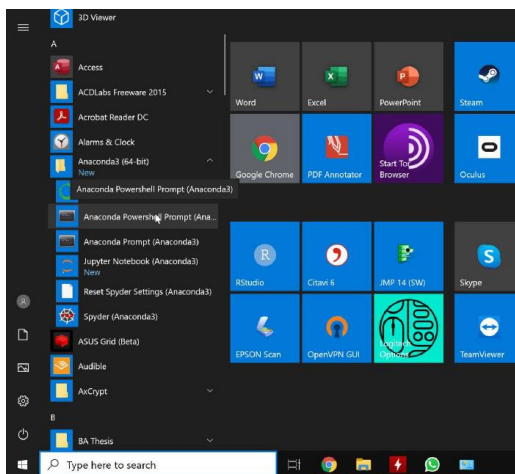


Abbildung 5: Öffnen der Anaconda Powershell Prompt 1

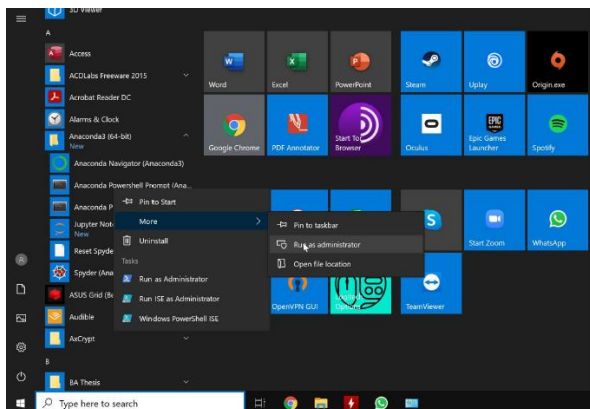


Abbildung 6 Öffnen der Anaconda Powershell Prompt 2

In der Anaconda Powershell Prompt muss nun folgender Befehl eingegeben werden, um die Tensorflow-Umgebung zu aktivieren:

Anmerkung: Die Graphische Darstellung der Befehle in der Anaconda Powershell Prompt sind in Abbildung 7 dargestellt.

```
conda activate tf
```

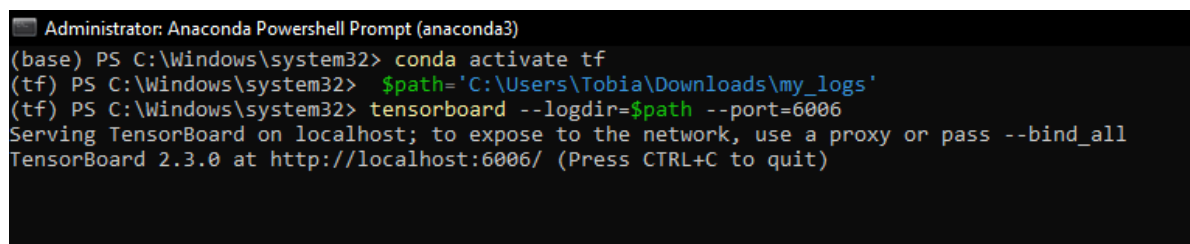
Im nächsten Schritt wird der Ordner der Log-Files definiert.

Der Dateipfad wird wie folgt angegeben (ACHTUNG: DER DATEIPFAD KANN VON COMPUTER ZU COMPUTER VARIIEREN UND MUSS ENTSPRECHEND ANGEPASST WERDEN!):

```
$path='C:\Users\Tobia\Downloads\my_logs'
```

Anschließend erfolgt die Angabe des Ports

```
tensorboard --logdir=$path --port=6006
```



```
Administrator: Anaconda Powershell Prompt (anaconda3)
(base) PS C:\Windows\system32> conda activate tf
(tf) PS C:\Windows\system32> $path='C:\Users\Tobia\Downloads\my_logs'
(tf) PS C:\Windows\system32> tensorboard --logdir=$path --port=6006
Serving TensorBoard on localhost; to expose to the network, use a proxy or pass --bind_all
TensorBoard 2.3.0 at http://localhost:6006/ (Press CTRL+C to quit)
```

Abbildung 7: Codezeilen zur Nutzung von Tensorboard

Nach Erfolgreicher Ausführung obenstehender Befehle kann im Browser (Chrom, etc.) in die Adresszeile <http://localhost:6006/> eingegeben werden. Durch Enter bestätigt erreicht man folgende Tensorboard-Startseite (Abbildung 8):

Hier ist z.B. die Accuracy sowie der Loss der einzelnen Epochen dargestellt.

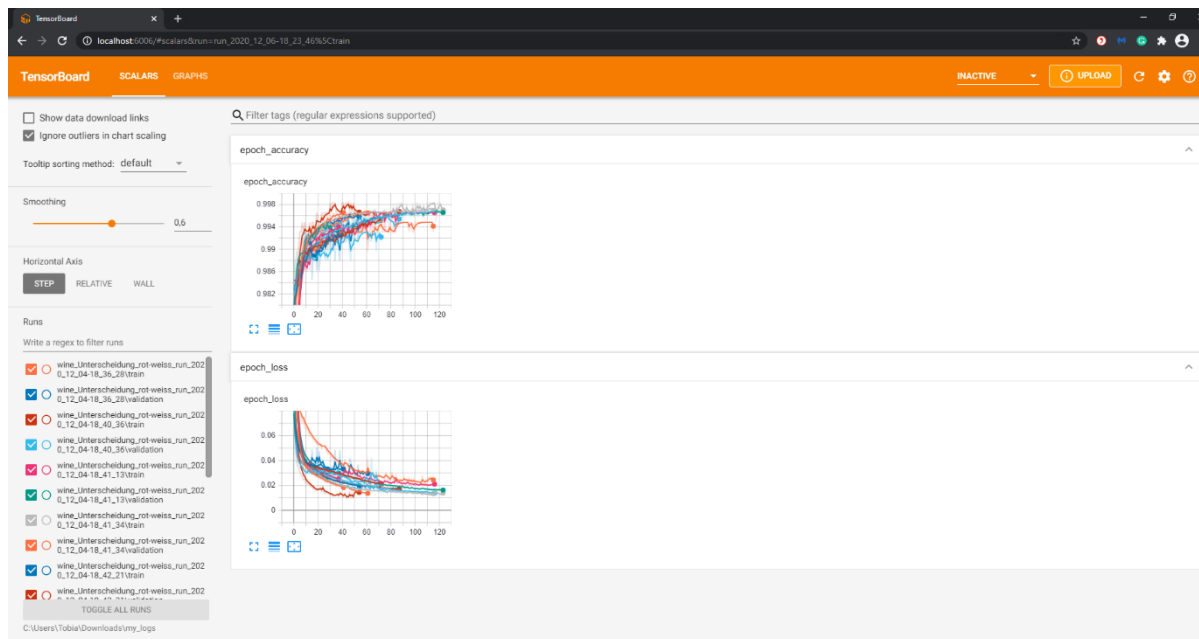


Abbildung 8: Startseite von Tensorboard

Nachdem die Arbeit mit Tensorboard beendet ist, wird tensorflow durch simultanes Drücken von strg + c beendet.

Anschließend wird die Conda-Oberfläche durch folgenden Befehl beendet.

*conda deactivate*

Alle notwendigen Codes zur Nutzung von Tensorboard sind in Abbildung 9 vom Starten bis zum Beenden dargestellt.

```
Administrator: Anaconda Powershell Prompt (anaconda3)
(base) PS C:\Windows\system32> conda activate tf
(tf) PS C:\Windows\system32> $path='C:\Users\Tobia\Downloads\my_logs'
(tf) PS C:\Windows\system32> tensorboard --logdir=$path --port=6006
Serving TensorBoard on localhost; to expose to the network, use a proxy or pass --bind_all
TensorBoard 2.3.0 at http://localhost:6006/ (Press CTRL+C to quit)
(tf) PS C:\Windows\system32> conda deactivate
(base) PS C:\Windows\system32>
```

Abbildung 9: Alle Befehle zur Nutzung von Tensorboard in der Anaconda Powershell Prompt

## 5. Perceptilabs

### 5.1 Installation von Perceptilabs

Achtung: Evtl. kann es bei der Installation zu Problemen kommen, wenn zusätzlich eine weitere (neuere) Python-Version auf dem Computer installiert ist.

Wichtig: Die Powershell sollte als Administrator ausgeführt werden.

#### Installation von Perceptilabs

##### 1. Deaktivieren der Anaconda Umgebung

```
PS C:\Users\Tobias> conda deactivate
```

##### 2. Installation von Python 3.7

```
PS C:\Users\Tobias> conda create -n pi python=3.7
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##
  environment location: C:\Users\Tobias\anaconda3\envs\pi
  added / updated specs:
    - python=3.7

The following NEW packages will be INSTALLED:

ca-certificates pkgs/main/win-64::ca-certificates-2020.12.8-ha95532_0
certifi          pkgs/main/win-64::certifi-2020.12.5-py37ha95532_0
openssl         pkgs/main/win-64::openssl-1.1.1i-h2b6ff1b_0
pip             pkgs/main/win-64::pip-20.3.3-py37ha95532_0
python          pkgs/main/win-64::python-3.7.9-h60c2a47_0
setuptools      pkgs/main/win-64::setuptools-51.0.0-py37ha95532_2
sqlite          pkgs/main/win-64::sqlite-3.33.0-h2ad8f80_0
vc              pkgs/main/win-64::vc-14.2-h21ff451_1
vs2015_runtime  pkgs/main/win-64::vs2015_runtime-14.27.26016-h5e8377_2
wheel           pkgs/main/noarch::wheel-0.36.2-pyhd3eb1b9_0
wincertstore    pkgs/main/win-64::wincertstore-0.2-py37_0
zlib            pkgs/main/win-64::zlib-1.2.11-h62dc097_4

Proceed ([y]/n)? y
```

Mit Ja bestätigen

```
Proceed ([y]/n)? y
Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#   $ conda activate pi
#
# To deactivate an active environment, use
#
#   $ conda deactivate
#
```

##### 3. Aktivieren der Perceptilabs-Umgebung

```
PS C:\Users\Tobias> conda activate pi
```

##### 4. Installation von Perceptilabs

```

(p1) PS C:\Users\tobias> pip install perceptilabs
Collecting perceptilabs
  Downloading perceptilabs-0.11.5-cp37-cp37m-win_amd64.whl (26.8 MB)
    |#####| 26.8 MB 269 KB/s
Requirement already satisfied: certifi>=2020.4.5.1 in c:\users\tobias\anaconda3\envs\pillib\site-packages (from perceptilabs) (2020.12.5)
Collecting azure-eventhub>=5.1.0
  Downloading azure-eventhub-5.1.0-py2.py3-none-any.whl (97 kB)
    |#####| 97 kB 2.1 MB/s
Collecting azure-storage>=0.36.0
  Downloading azure-storage-0.36.0-py2.py3-none-any.whl (190 kB)
    |#####| 190 kB ...
Collecting colorama>=0.4.3
  Downloading colorama-0.4.3-py2.py3-none-any.whl (15 kB)
Collecting jsonschema>=3.2.0
  Downloading jsonschema-3.2.0-py2.py3-none-any.whl (56 kB)
    |#####| 56 kB 4.1 MB/s
Requirement already satisfied: setuptools in c:\users\tobias\anaconda3\envs\pillib\site-packages (from jsonschema>=3.2.0->perceptilabs) (51.0.0.post20201207)
Collecting mixpanel>=4.7.0-py2.py3-none-any.whl (8.2 kB)
  Downloading mixpanel-4.7.0-py2.py3-none-any.whl (8.2 kB)
Collecting pillow>=7.0.0
  Downloading pillow-7.0.0-cp37-cp37m-win_amd64.whl (2.0 MB)
    |#####| 2.0 MB 6.8 MB/s
Collecting sentry-sdk>=0.14.3
  Downloading sentry-sdk-0.14.3-py2.py3-none-any.whl (103 kB)
    |#####| 103 kB 6.4 MB/s
Collecting tensorflow>=1.15
  Downloading tensorflow-1.15.0-cp37-cp37m-win_amd64.whl (295.1 MB)
    |#####| 295.1 MB 6.4 KB/s
Requirement already satisfied: wheel>=0.26 in c:\users\tobias\anaconda3\envs\pillib\site-packages (from tensorflow>=1.15->perceptilabs) (0.36.2)
Collecting gast>=0.2.2
  Downloading gast-0.2.2.tar.gz (10 kB)
Collecting tensorflow-estimator>=1.15.1
  Downloading tensorflow-estimator-1.15.1-py2.py3-none-any.whl (503 kB)
    |#####| 503 kB 3.3 MB/s
Collecting wrllib3>=1.25.10
  Downloading wrllib3-1.25.10-py2.py3-none-any.whl (127 kB)
    |#####| 127 kB ...
Collecting absl-py>=0.7.0
  Using cached absl_py-0.7.0-py3-none-any.whl (127 kB)
Collecting asgiref>=3.2.10
  Downloading asgiref-3.2.10-py3-none-any.whl (19 kB)
Collecting astor>=0.8.0
  Downloading astor-0.8.1-py2.py3-none-any.whl (27 kB)
Collecting attrs>=17.4.0
  Downloading attrs-20.3.0-py2.py3-none-any.whl (49 kB)
    |#####| 49 kB 1.5 MB/s
Collecting azure-common>=1.1.5
  Downloading azure-common-1.1.26-py2.py3-none-any.whl (12 kB)
Collecting azure-core>=2.0.0, <=1.5.0
  Downloading azure-core-1.0.0-py2.py3-none-any.whl (124 kB)
    |#####| 124 kB 6.4 MB/s
Collecting azure-mspkg>=2.0.0
  Downloading azure_mspkg-2.0.2-py3-none-any.whl (1.5 kB)
Collecting blessings>=1.7
  Downloading blessings-1.7-py3-none-any.whl (18 kB)
Collecting boltons>=19.3.0
  Downloading boltons-20.2.1-py2.py3-none-any.whl (170 kB)
    |#####| 170 kB 6.4 MB/s
Collecting boto>=2.49.0
  Downloading boto-2.49.0-py2.py3-none-any.whl (1.4 MB)

```

## 5.2. Starten von Perceptilabs

### 1. Aktivieren der Perceptilabs-Umgebung

```
PS C:\Users\tobias> conda activate pl
```

### 2. Starten von Perceptilabs

```

(p1) PS C:\Users\tobias> perceptilabs
Perceptilabs: Your environment does not have atari.py installed, so some functionality may not be available
Perceptilabs: To install it, please follow the directions at https://github.com/KojoJey/ataripy and then install gym through 'pip install gym[atari]'
INFORMATION: Es konnten keine Dateien mit den angegebenen
Pustern gefunden werden.
Perceptilabs: Your environment does not have git installed, so interactions with GitHub will not be available
Perceptilabs: Starting
Perceptilabs: Waiting for services to listen on these ports:
Perceptilabs:   kernel on port 5000
Perceptilabs:   Waiting for services to listen on these ports:
Perceptilabs:   kernel on port 5000
Perceptilabs: Waiting for services to listen on these ports:
Perceptilabs:   kernel on port 5000
Perceptilabs: Perceptilabs Started
Perceptilabs: Perceptilabs is running at http://localhost:8080/?token=fthab566_r0M7ZDCHZn9VLvKIDTHRAuuqKX-46Pc-Y
Perceptilabs: Use Control+C to stop this server and shut down all Perceptilabs processes.

```

Perceptilabs durch Drücken von strg + c beenden