

# LU03c - PyCharm - Projekt erstellen

Erstellen Sie für das Modul 319 ein neues Projekt in PyCharm.

Entweder via `File > New > Project`



oder über den Projekte-Dialog:



- 
1. Wählen Sie Pure Python als Projekt-Art aus
  2. Geben Sie bei „Location“ den Pfad `C:\PythonProjects\PROJEKTNAME` an, wobei Sie statt PROJEKTNAME einen aussagekräftigen Namen angeben.

Wählen Sie `New environment using virtualenv`



Im Feld „Location“ unterhalb von „New environment using“ muss der gleiche Pfad ergänzt um `\venv` wie oben stehen. Also `C:\BZZ\Python\PROJEKTNAME\venv`. Falls dies nicht so ist, korrigieren Sie den Pfad.

Klicken Sie anschliessen `Create`

---

PyCharm erstellt Ihnen nun Ihr ersten Python-Projekt.



Verwenden Sie den `Play` Button oben rechts um Ihre Applikation laufen zu lassen.



Im `run`-Fenster sehen Sie nun die Ausgabe des Programms.



## Grundgerüst eines Python-Programms

Wenn Sie ein Python-Projekt erstellen oder eine Vorlage für eine Aufgabe kopieren, sind bereits einige Zeilen vorhanden. Diese Zeilen bilden das Grundgerüst unserer Programme und sind meistens identisch:

<code>def main():</code>	Wir definieren eine Funktion mit dem Namen <code>main</code>
--------------------------	--

<code># write the program logic here and remove the line "pass"</code>	Dieser Kommentar zeigt, wo Sie ihren Programmcode schreiben sollen
<code>pass</code>	Der Befehl <code>pass</code> macht nichts, ist aber notwendig für die Vorlage.
<code>if __name__ == '__main__':</code>	Diese Codezeile vereinfacht es uns, das Programm auszuführen
<code>main()</code>	Wir rufen die oben definierte Funktion <code>main</code> auf.

Einfache Python-Programme könnten auch ohne dieses Grundgerüst geschrieben werden. Damit wir aber nicht später ein neues Gerüst erlernen müssen, schreiben wir von Anfang an alle Programme auf dieser Basis.

## Hinweise

### main

Die Funktion, die auf der ersten Zeile beginnt, muss nicht zwingend `main` heißen. Wir könnten einen beliebigen Namen wählen. Der Name `main` für die Hauptverarbeitung hat sich aber in vielen Programmiersprachen eingebürgert.

### pass

Der Befehl `pass` dient als Platzhalter in einem Programmblock. Python erwartet, dass in jeder Funktion mindestens 1 Befehl geschrieben wird. Andernfalls würde unserer Entwicklungsumgebung einen Fehler anzeigen. Deshalb schreiben wir in der Vorlage den Befehl `pass`, damit kein Fehler angezeigt wird. Sobald Sie beginnen, die eigentlichen Befehle wie `print` zu schreiben, können Sie `pass` entfernen.

[m319-LU03](#), [m319-D1G](#)



© Kevin Maurizi

From:

<https://wiki.bzz.ch/> - **BZZ - Modulwiki**

Permanent link:

<https://wiki.bzz.ch/de/modul/m319/learningunits/lu03/pycharm?rev=1750657503>

Last update: **2025/06/23 07:45**

