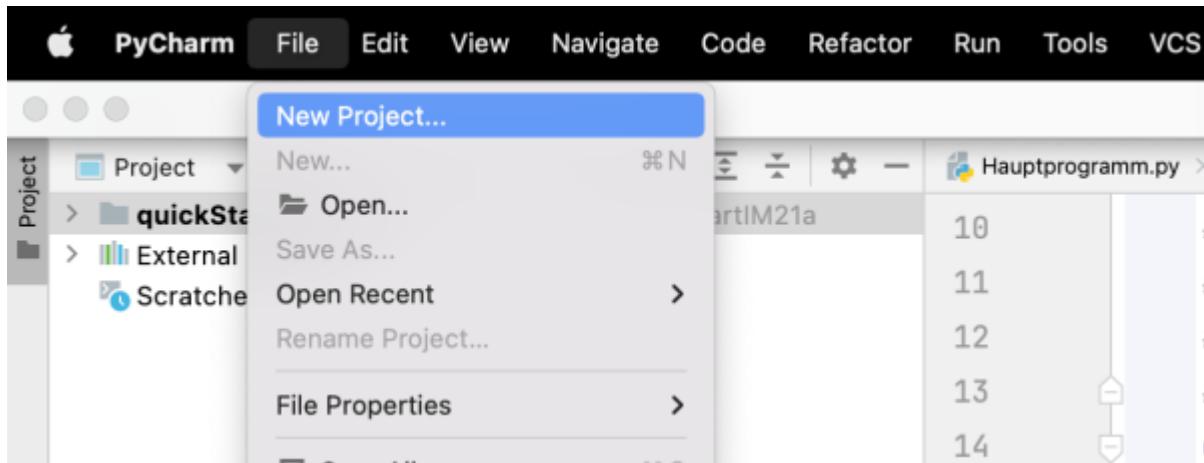


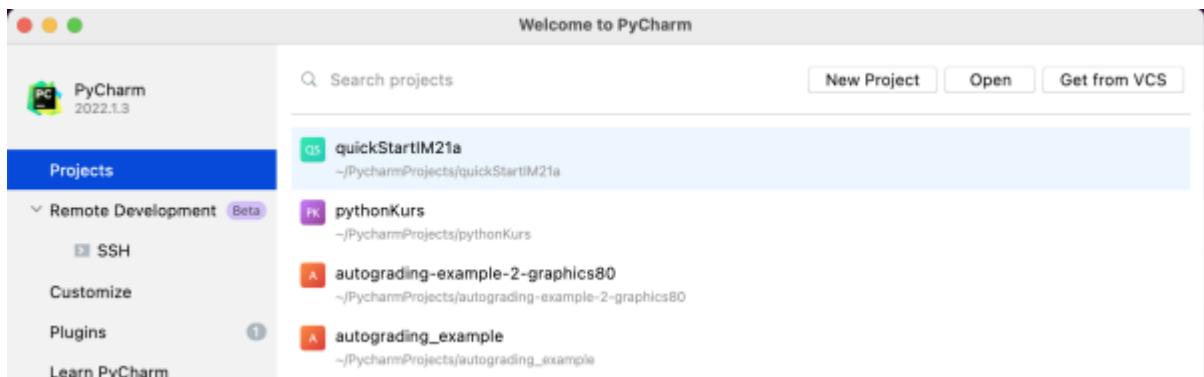
# LU03c - PyCharm - Projekt erstellen

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

Entweder via File > New > Project

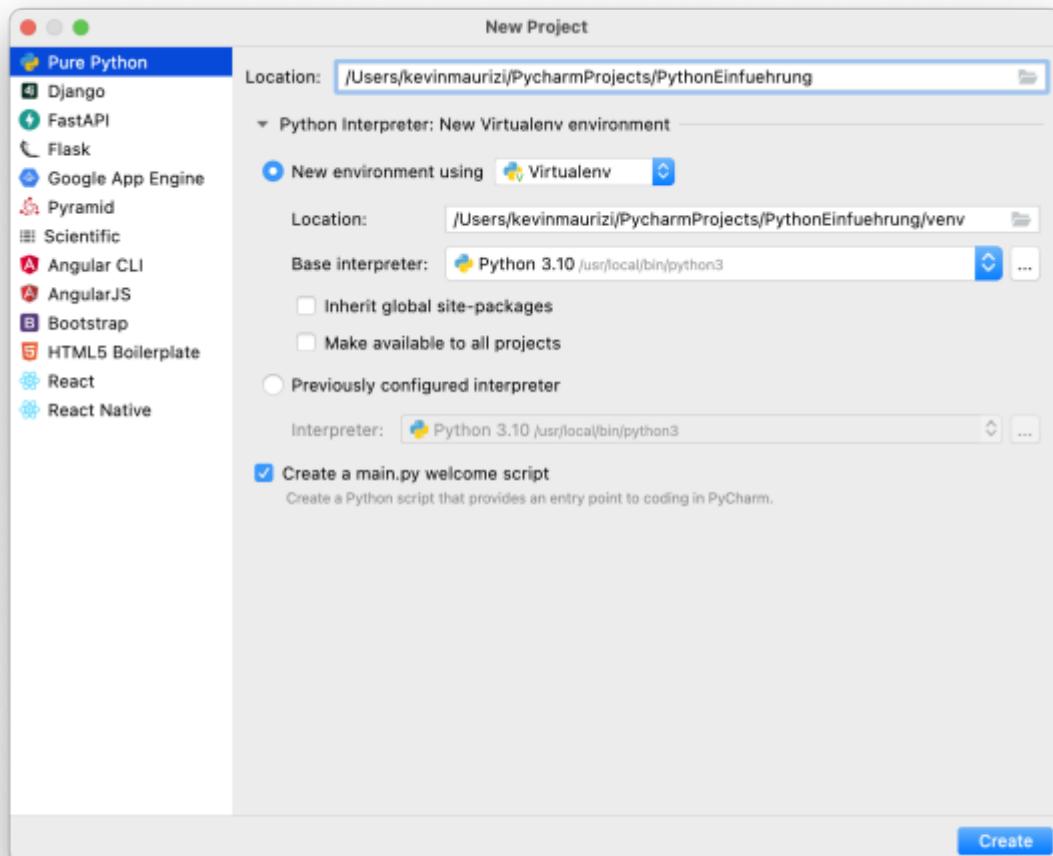


oder über den Projekte-Dialog:



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\PROJEKTNAMEN\venv. Falls dies nicht so ist, korrigieren Sie den Pfad.

Klicken Sie anschliessen Create

---

PyCharm erstellt Ihnen nun Ihr ersten Python-Projekt.

```

PythonEinführung main.py
Project PythonEinführung - PyCharmProject/PythonEinführung
  + venv
    + bin
    + lib
    + __pycache__
    + __pycache__.pyc
  main.py
  + External Libraries
  + Scratches and Caches

This is a sample Python script.

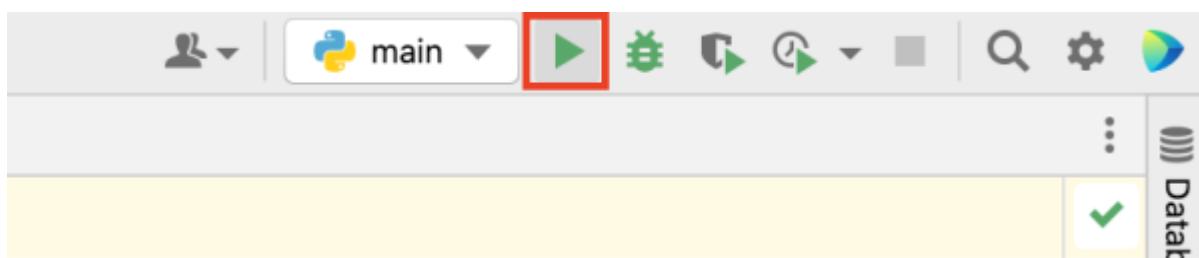
1 | # Press ⌘R to execute it or replace it with your code.
2 |
3 | # Press Double ⇧ to search everywhere for classes, files, tool windows, options, and settings.
4 |
5 |
6 |
7 | def print_hi(name):
8 |     # Use a breakpoint in the code line below to debug your script.
9 |     print(f'Hi, {name}') # Press ⌘F8 to toggle the breakpoint.

10|
11|
12| # Press the green button in the gutter to run the script.
13| if __name__ == '__main__':
14|     print_hi('PyCharm')

15|
16| # See PyCharm help at https://www.jetbrains.com/help/pycharm/
17|

```

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



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

```

Run main
  /Users/kevinsaurizi/PycharmProjects/PythonEinführung/venv/bin/python /Users/kevinsaurizi/PycharmProjects/PythonEinführung/main.py
  Hi, PyCharm
  Process finished with exit code 0

```

## 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

pass	Der Befehl pass macht nichts, ist aber notwendig für die Vorlage.
if __name__ == '__main__':	Diese Codezeile vereinfacht es uns, das Programm auszuführen
main()	Wir rufen die oben definierte Funktion main 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 Programmblöck. 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>

Last update: **2025/11/17 08:33**

