

# Vorläufige Version, To be improved...

## 1.1 Installation

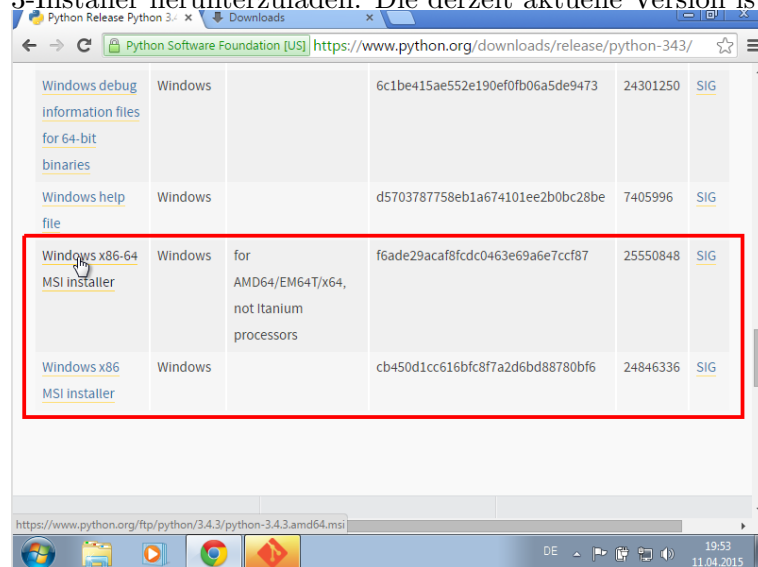
Diese Anleitung ist für Windows ausgelegt.

### 1.1.1 Download

Python kann online unter <https://www.python.org/downloads/> heruntergeladen werden.



**Hinweis:** Im CoderDojo verwenden wir Python 3, deswegen ist es wichtig den Python 3-Installer herunterzuladen. Die derzeit aktuelle Version ist 3.4.3.



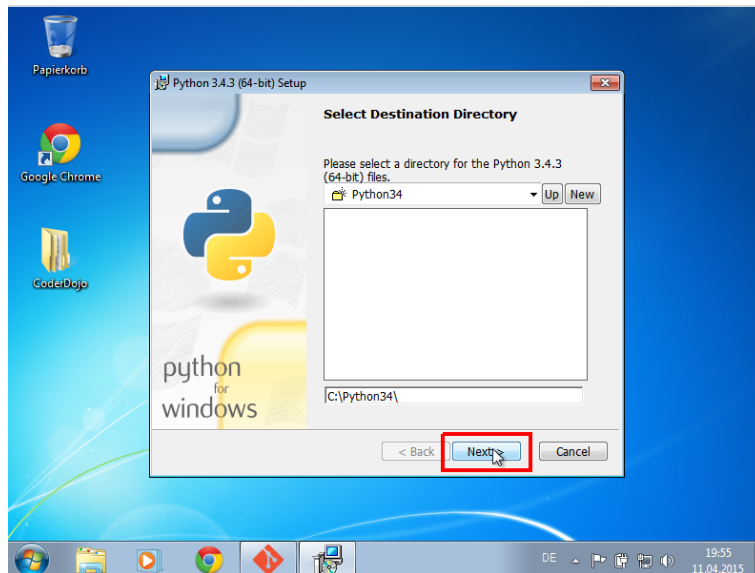
Falls der Download nicht automatisch startet, muss die korrekte Version von Hand ausgewählt werden. In dem Bild die 64-bit Windows Version, dies ist allerdings abhängig

von der installierten Windowsversion.

### 1.1.2 Installation

Die heruntergeladene Installationsdatei durch Doppelklick starten. In den erscheinenden Fenstern auf "Weiter" und/oder "Ok" klicken um die Installation durchzuführen. Die Voreingestellten Werte (z.B. Installationsverzeichnis) sollten okay sein, können aber natürlich nach belieben geändert werden.

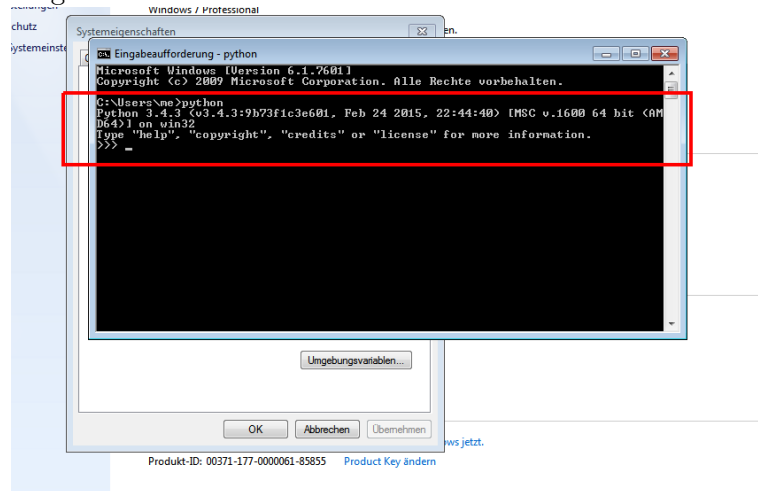
**Hinweis:** Die Installation erfordert Administratorrechte.



Die Installation sollte ohne Probleme durchlaufen. Danach ist Python erfolgreich auf dem System installiert.

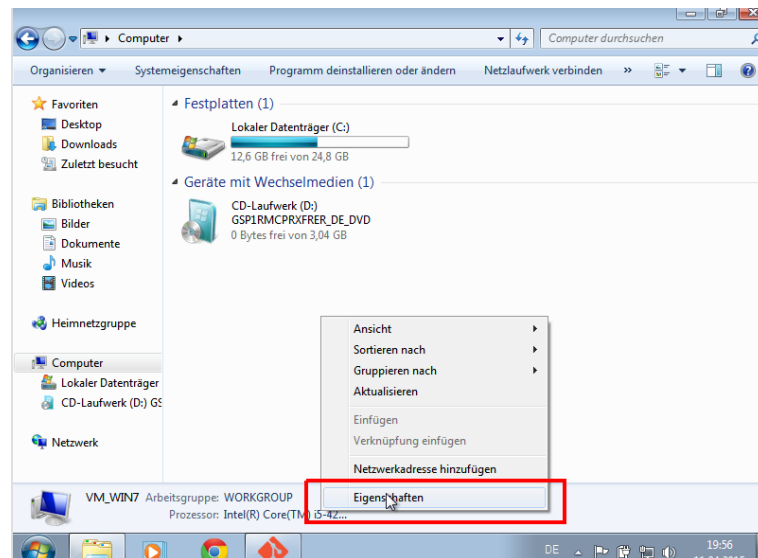
## 1.2 Anpassung der Pfad-Umgebungsvariablen

Um Python einfacher auf dem Rechner nutzen zu können ist es empfehlenswert den Installationsordner zur Pfad-Umgebungsvariablen hinzuzufügen. Programme deren Installationsordner in der Pfad-Umgebungsvariablen stehen können direkt über ihren Namen aufgerufen werden.

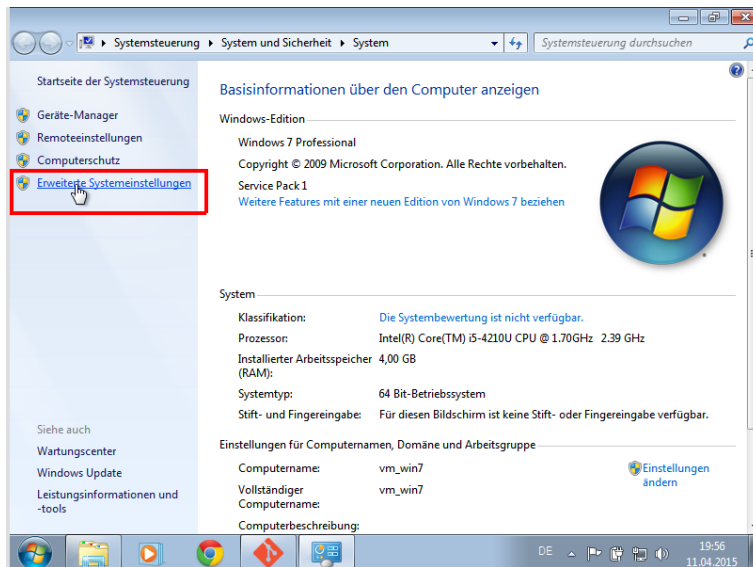


Python kann dann direkt von der Kommandozeile/Konsole verwendet werden.

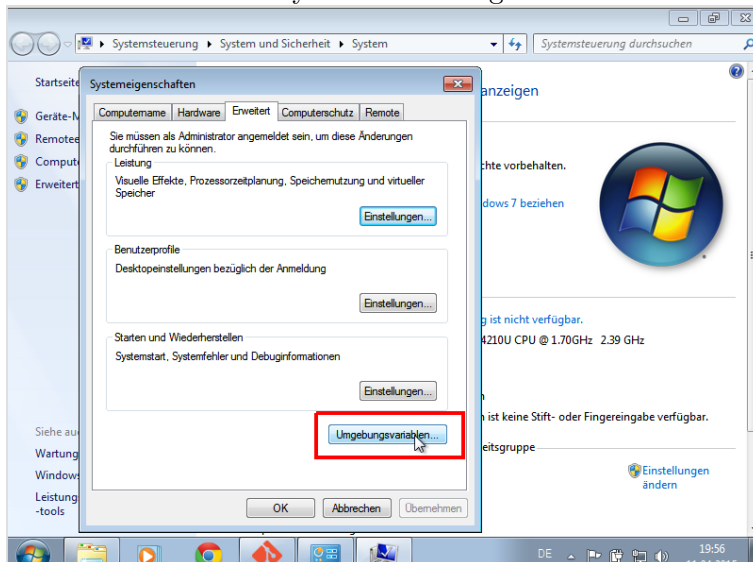
### 1.2.1 Finden der Umgebungsvariablen



Diese sind mittels Rechtsklick auf "Computer/Arbeitsplatz" ->"Eigenschaften" ...



... unter "Erweiterte Systemeinstellungen" ...

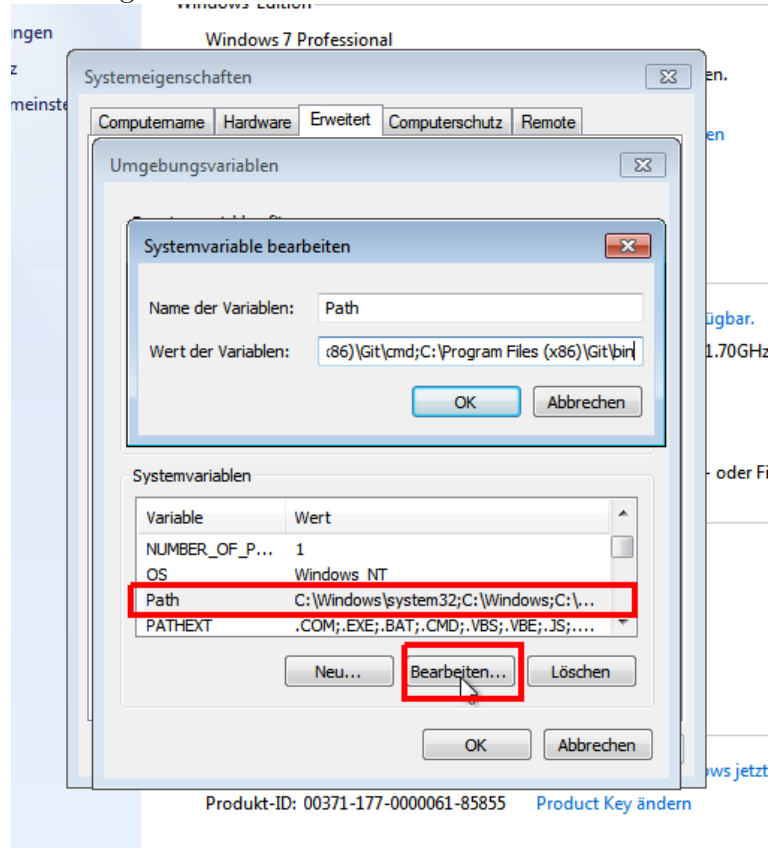


... zu finden.

### 1.2.2 Anpassen der Pfad-Umgebungsvariablen

Nun muss aus der Liste der Umgebungsvariablen die "Pfad" / "Path" ausgewählt werden. Durch Klicken auf "Bearbeiten..." öffnet sich ein Dialog zum Bearbeiten der Pfad-Umgebungsvariablen.

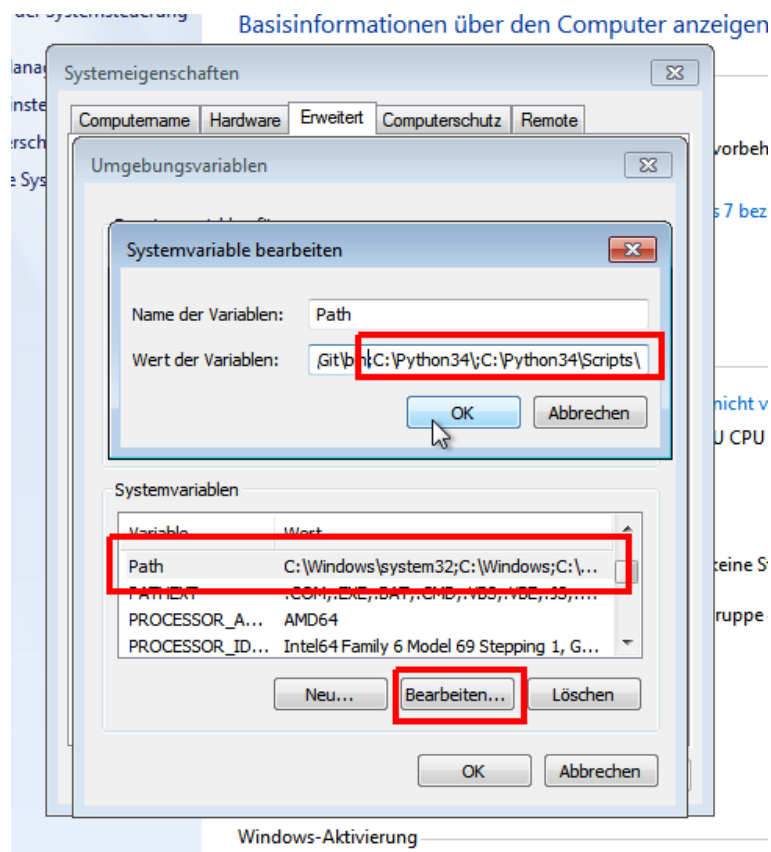
**Hinweis:** Das Ändern der Pfad-Umgebungsvariablen erfordert Administratorrechte. Außerdem ist Vorsicht geboten beim Bearbeiten der Pfad-Variablen. Sollte diese aus Versehen gelöscht oder fehlerhaft bearbeitet werden, kann dies Auswirkungen auf andere Programme haben kann!



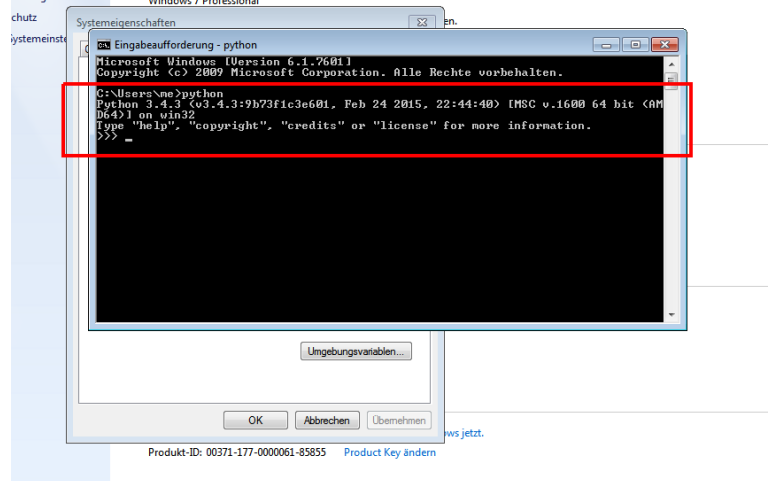
Die Pfad-Umgebungsvariable ist eine Liste von Verzeichnispfaden, die mit ';' getrennt geschrieben sind.

Nun muss am Ende mit ';' das Python-Installationsverzeichnis eingefügt werden. Das Standardverzeichnis für Python 3.4 ist "C:\Python34\". Der Pfad muss allerdings gegebenenfalls angepasst werden!

Optional kann auch das "Scripts"-Verzeichnis im Python-Ordner zur Pfad-Variablen hinzugefügt werden. Dies hilft, falls später später bspw. mittels "pip" weitere Python-Module installiert werden sollen.



Nach Bestätigung über "Ok" und Schließen der Fenster sollte die Pfadvariablen korrekt gesetzt sein. Dies kann überprüft werden indem man die Windowskommandozeile öffnet und "python" eintippt und mit der Enter-Taste bestätigt. Die Kommandozeile kann man durch Suchen von "cmd" im Windows-Suchfeld finden.

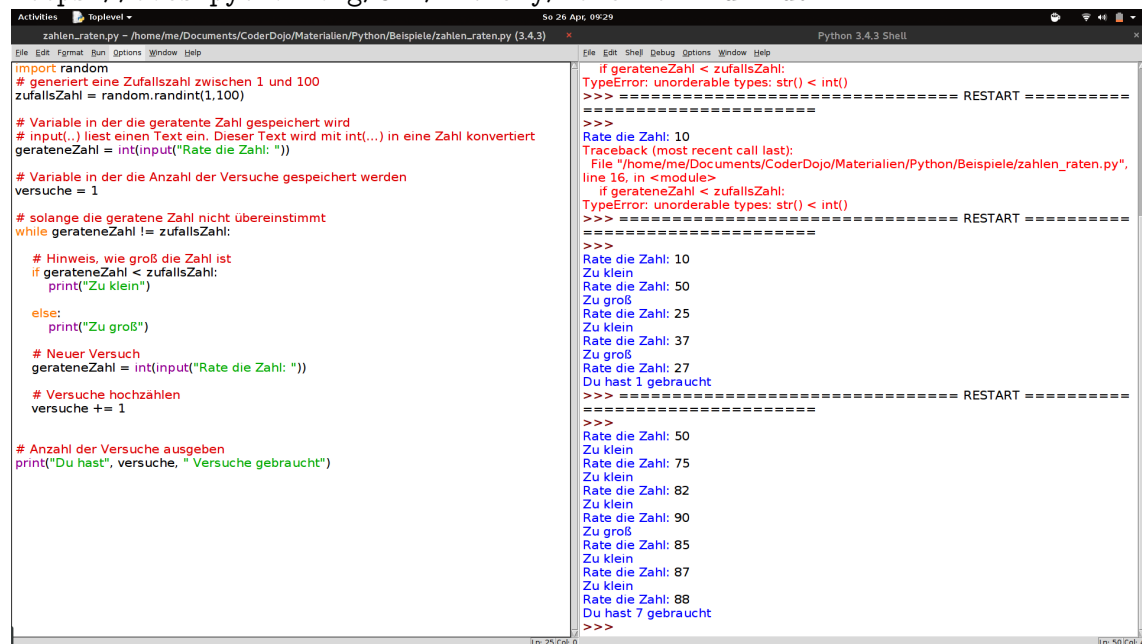


## 1.3 Editor

Die Installation von Python sollte jetzt komplett sein. Nun ist nur noch die Frage zu klären, wie und mit welchem Programm Python programmiert werden kann.

### 1.3.1 IDLE

Standartmäßig kommt mit Python die IDLE (Integrated Development Environment), die benutzt werden kann. Diese ist allerdings nicht allzu komfortabel zu benutzen, besitzt allerdings alle Wesentlichen Funktionen. Die englische Dokumentation ist hier: <https://docs.python.org/3.4/library/idle.html> zu finden.



The screenshot shows the Python IDLE interface. The left pane displays a Python script named `zahlen_raten.py` with the following code:

```
import random
# generiert eine Zufallszahl zwischen 1 und 100
zufallsZahl = random.randint(1,100)

# Variable in der die geratene Zahl gespeichert wird
# input(.) liest einen Text ein. Dieser Text wird mit int(...) in eine Zahl konvertiert
gerateneZahl = int(input("Rate die Zahl: "))

# Variable in der die Anzahl der Versuche gespeichert werden
versuche = 1

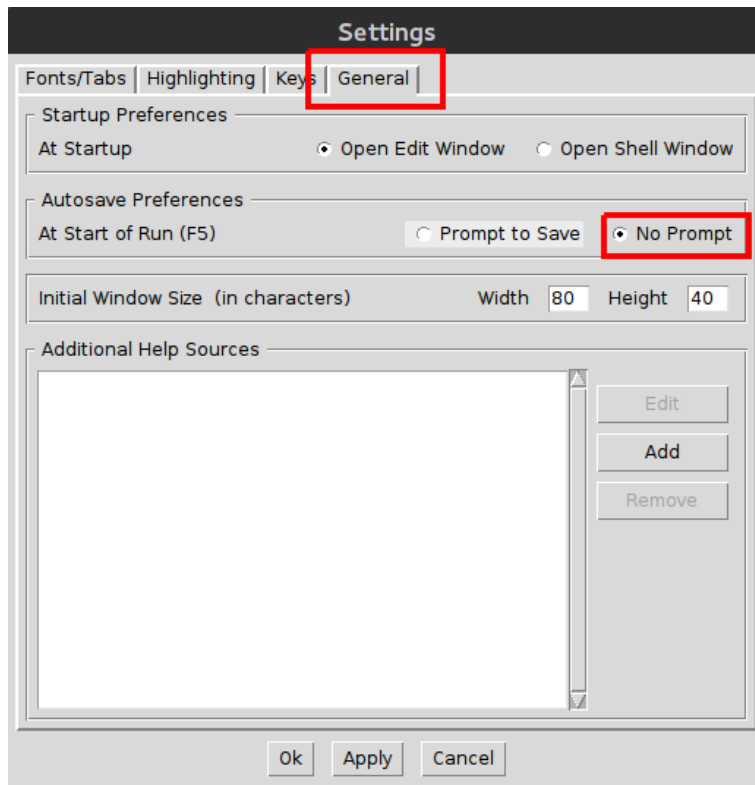
# solange die geratene Zahl nicht übereinstimmt
while gerateneZahl != zufallsZahl:
    # Hinweis, wie groß die Zahl ist
    if gerateneZahl < zufallsZahl:
        print("Zu klein")
    else:
        print("Zu groß")
    # Neuer Versuch
    gerateneZahl = int(input("Rate die Zahl: "))
    # Versuche hochzählen
    versuche += 1

# Anzahl der Versuche ausgeben
print("Du hast", versuche, "Versuche gebraucht")
```

The right pane shows the Python Shell output, which includes a `TypeError: unorderable types: str() < int()` error and a subsequent `RESTART` prompt. The user has entered several numbers (10, 50, 75, 82, 90, 85, 87, 88) and the program has responded with "Zu klein" or "Zu groß" until it finally prints "Du hast 7 gebraucht".

Unter "File ->New" kann eine neue Datei erstellt werden. Unter "Run ->Run Module (F5)" kann die Datei ausgeführt werden. Dies öffnet ein "Python Shell Fenster". Es empfiehlt, solange der Bildschirm groß genug ist, beide Fenster nebeneinander zu haben.





Um die Datei auszuführen, ist es gut sich die Tastenkombination "F5" zum Ausführen zu merken und zu benutzen. Dafür ist es außerdem hilfreich, wenn man in den Einstellungen unter "Options -> Configure IDLE" im "General"-Tab die Einstellung bei "Autosave Preferences" auf "No Prompt" setzt, damit die Datei automatisch beim Ausführen gespeichert wird.