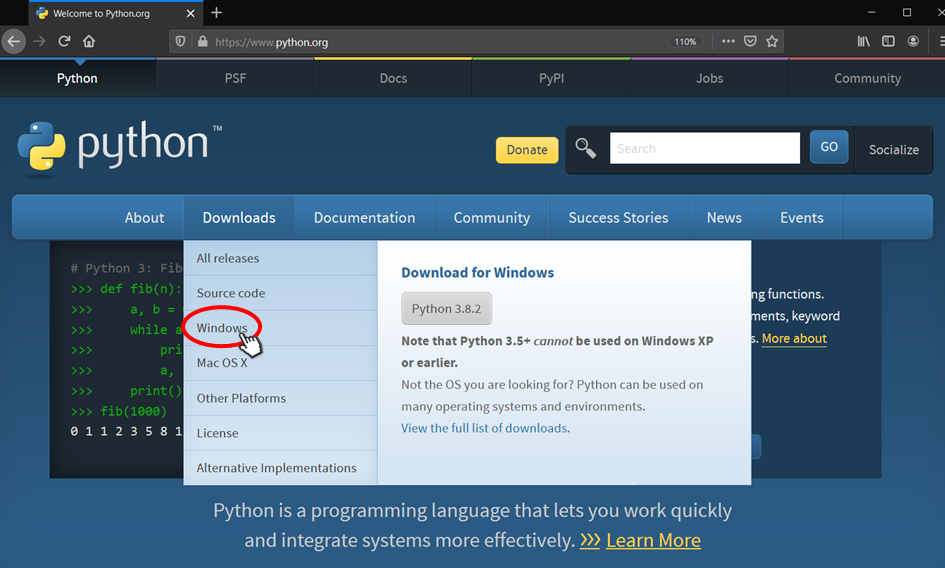
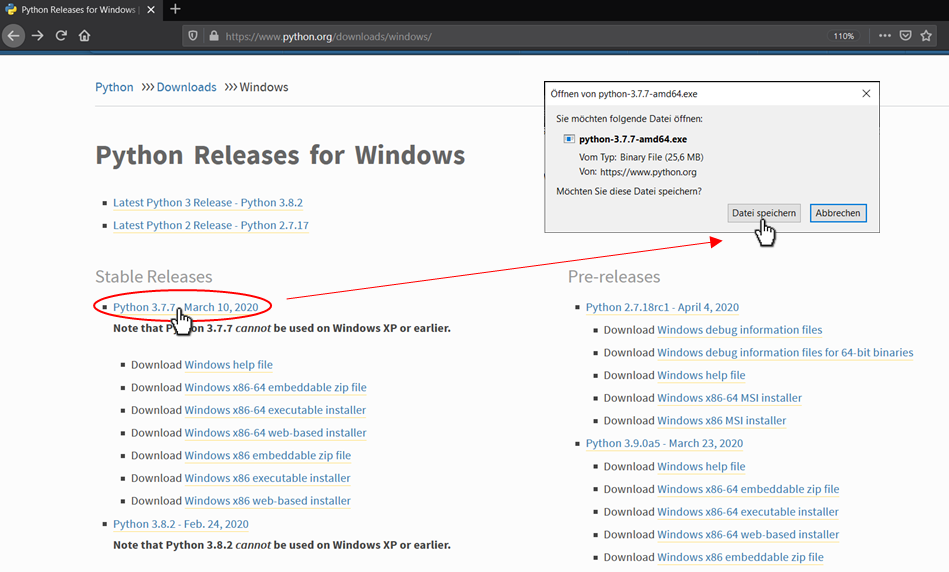
**Anleitung erste Schritte MANGA:**

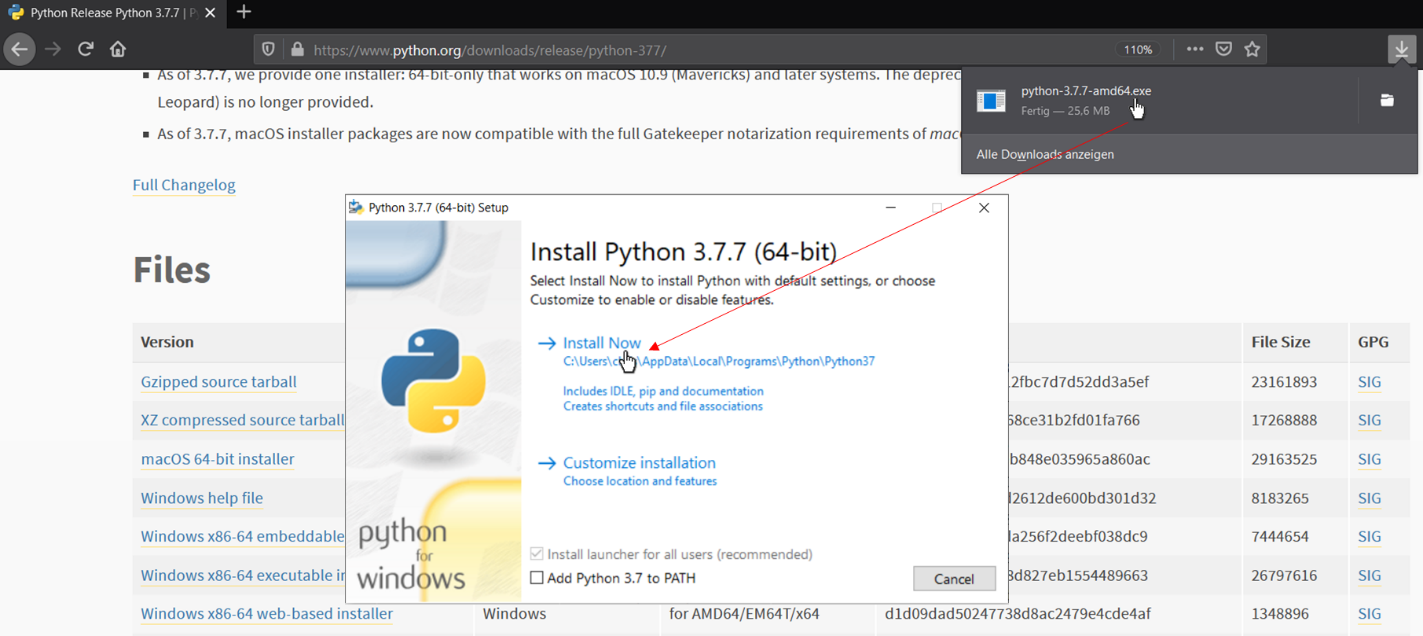
Um MANGA (Mangrove groundwater salinity feedback model) ausführen zu können, müssen Sie zunächst sich ein Compiler für die Programmiersprache Python besorgen. Ein Beispiel wäre pythonTM. Dazu öffnen Sie Ihren Browser und gehen Sie auf die Seite *Python.org*. Im Auswahlmenü unter *Download* finden Sie die aktuelle Release Version für ihr Betriebssystem von Python (in dieser Anleitung wird das Vorgehen unter Windows beschrieben, siehe Abbildung 1).



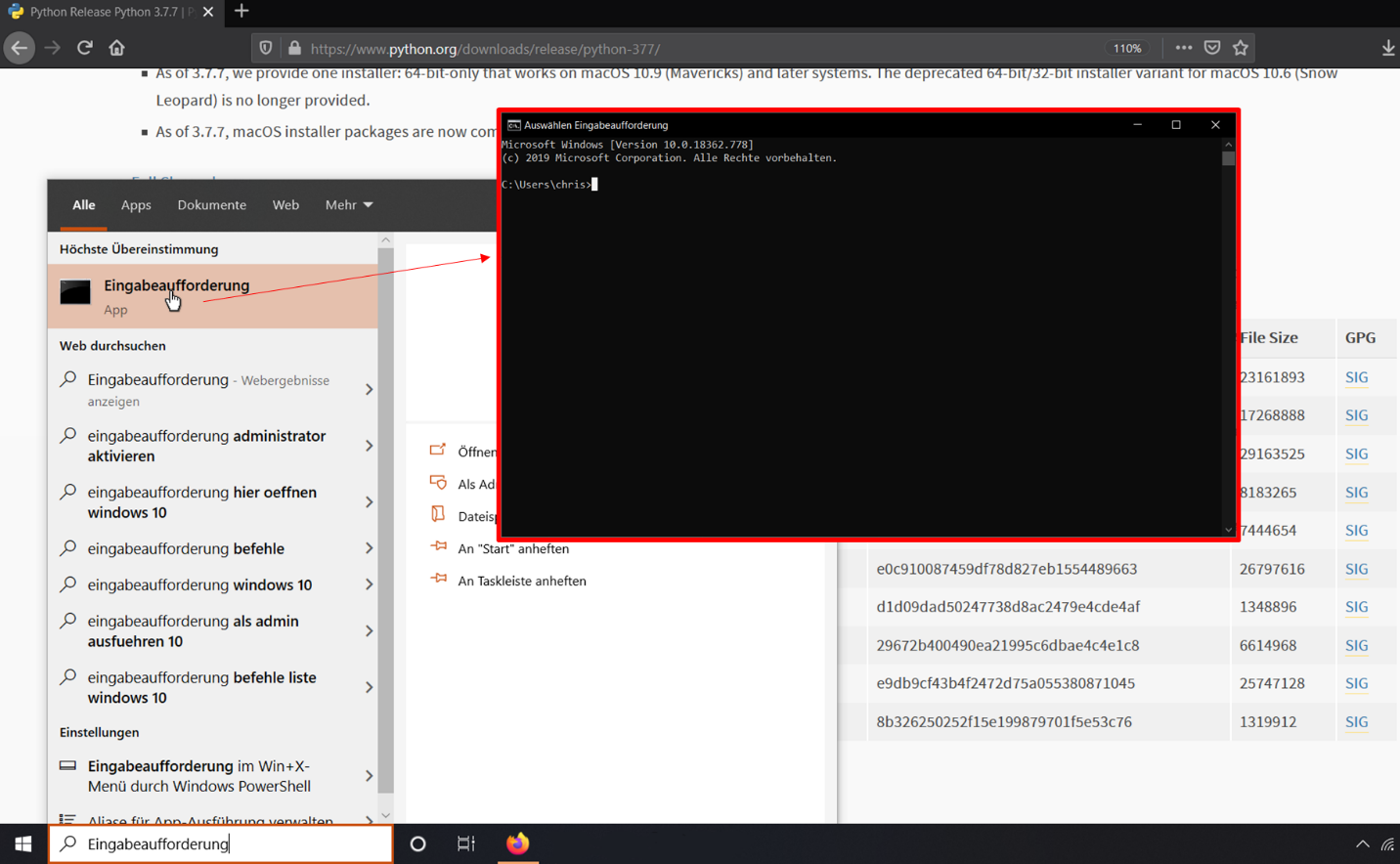
Wählen Sie eine Version aus beispielsweise *Python 3.7.7* und klicken Sie darauf. Daraufhin erscheint eine Beschreibung der ausgewählten Version und die dazugehörigen möglichen Download Files. Hier wählen Sie den *Windows x86-64 executable installer* und laden diesen per Mausklick herunter, bestätigen Sie diesen Vorgang, indem Sie auf *Datei speichern* klicken (siehe Abbildung 2).



Führen Sie die herunter geladene Datei (*python-3.7.7-amd64.exe*) aus, wie eine normale Windows exe und installieren Sie sie auf Ihren Rechner (siehe Abbildung 3). Um MANGA auszuführen müssen noch ein paar Module für dem Python Compiler installiert werden.

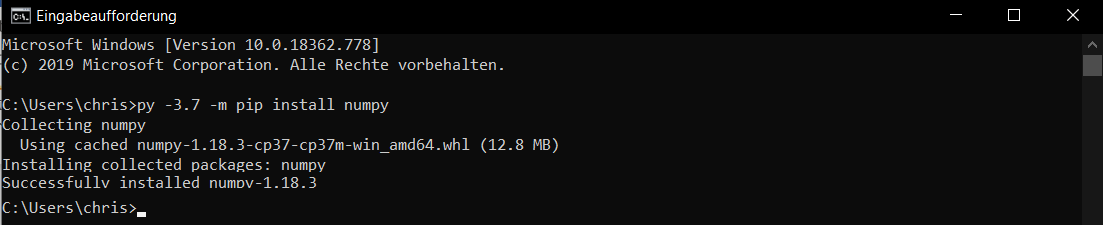


Dazu müssen Sie die *Eingabeaufforderung* öffnen. Diese finden Sie einfach über die Suche, indem Sie der „Eingabeaufforderung“ eingeben und per Mausklick öffnen. Da es sich bei MANGA um ein Zeilenprogramm handelt, spielt sich alles in der Eingabeaufforderung ab (siehe Abbildung 4).



Nun müssen die folgenden Module *numpy*, *vtk*, *lxml* und *matplotlib* installiert werden. Wir beginnen mit den Modulen *numpy*. Geben Sie den aufgezeigten Code in die Eingabeaufforderung, um das Modul zu installieren (siehe Abbildung 5).

* py -3.7 -m pip install numpy [1]

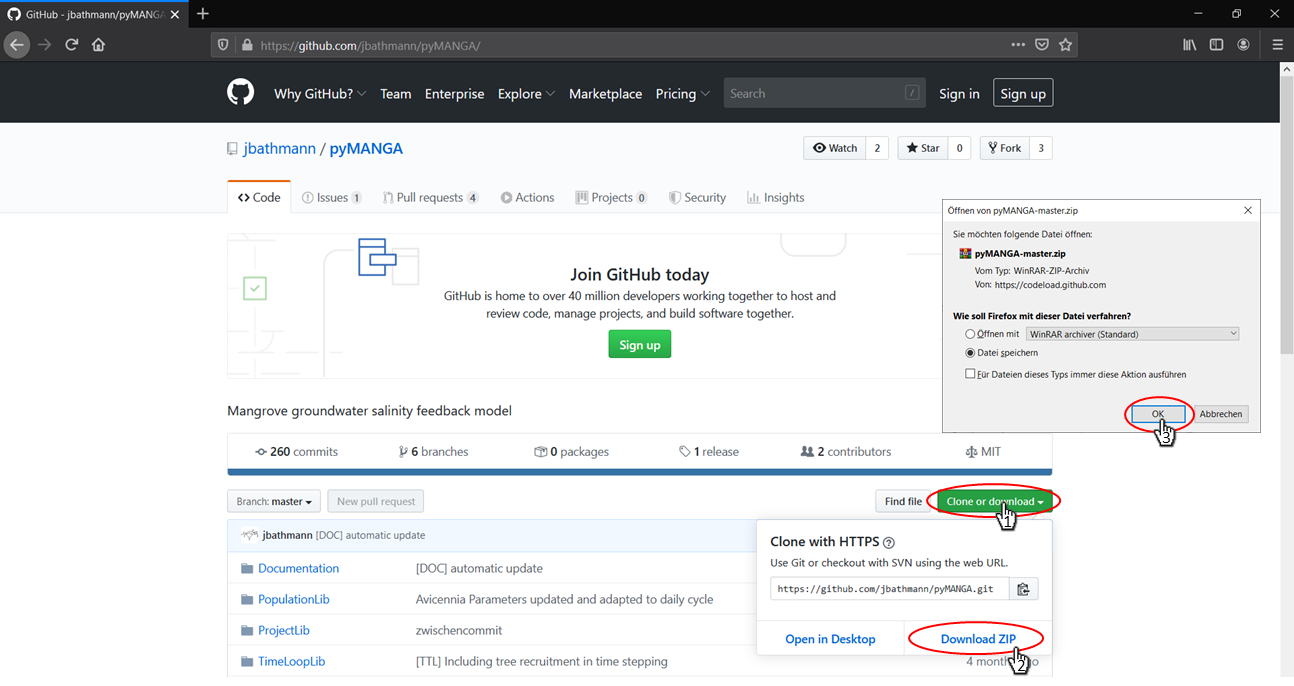


Führen Sie dies analog für die drei anderen Module aus mit dem folgenden Code

* py -3.7 -m pip install vtk [1]
* py -3.7 -m pip install lxml [2]
* py -3.7 -m pip install lxml matplotlib [3]

Hinweis: Sollte die Eingabeaufforderung eine Wahrnung ausgeben, dass *pip* nicht aktuell ist, können Sie mit *upgrade pip* dies aktualisieren. Dies ist aber nicht zwingend erforderlich.

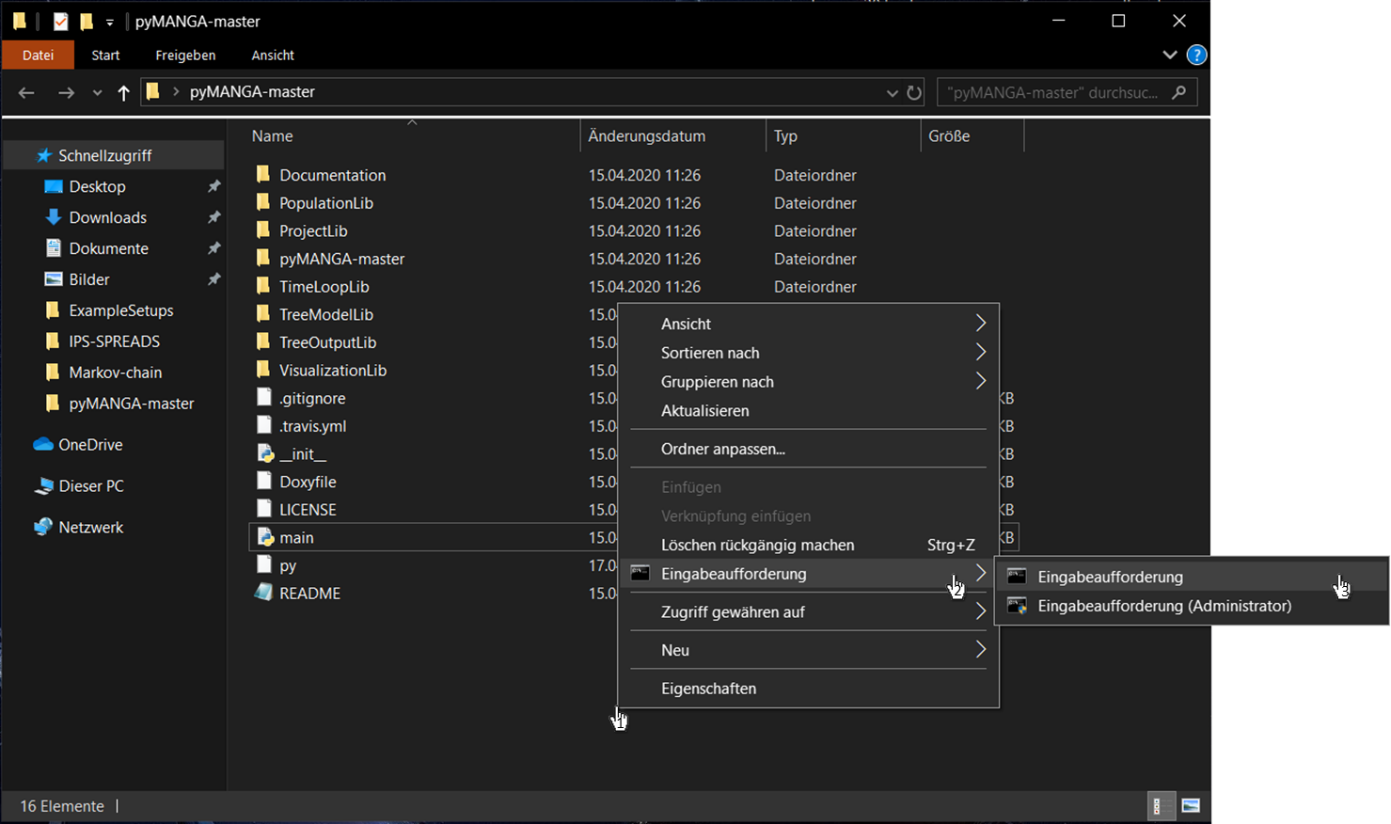
Zur Erklärung was Sie eingegeben haben: *py* bedeuten, dass Sie Python aufrufen. Dabei ist *-3.7* die Version, die Sie nutzen. Mit *-m* wird ein Modul aufgerufen, in diesem Fall *pip*, welches dazu dient andere Module zu installieren. Zum Schluss folgt der Modul Name vom zu installierendem Modul. Nun sind die Vorbereitungen für die Nutzung des Compilers abgeschlossen. Als nächsten Schritt müssen Sie, falls es noch nicht geschehen ist, das Programm MANGA downloaden. Dazu gehen Sie auf die folgende Internetseite [https://github.com/jbathmann/pyMANGA/](https://deref-web-02.de/mail/client/S8bSeluLJF8/dereferrer/?redirectUrl=https%3A%2F%2Fgithub.com%2Fjbathmann%2FpyMANGA%2F) und downloaden das Programm als zip Datei und speichern es auf Ihren Rechner (siehe Abbildung 6).



Anschließend entpacken Sie die Datei (*pyMANGA-master.zip*) auf Ihren Desktop. Sie enthält sämtliche Programmbestandteile von MANGA unteranderem *main.py*, welche die Ausführungsdatei darstellt, die zum Ausführen des Programmes aufgerufen werden muss. Dazu muss Sie nun dem Ordner öffnen und mit Rechtsklick in einem leeren Bereich des Ordners die Eingabeaufforderung öffnen (siehe Abbildung 7) und den folgenden Code eingeben.

* py main.py -h [4]

Hierbei bedeutet wiederum *py* das Python aufgerufen wird, *main.py* stellt die Datei dar, die aufgerufen werden soll, und -h ??



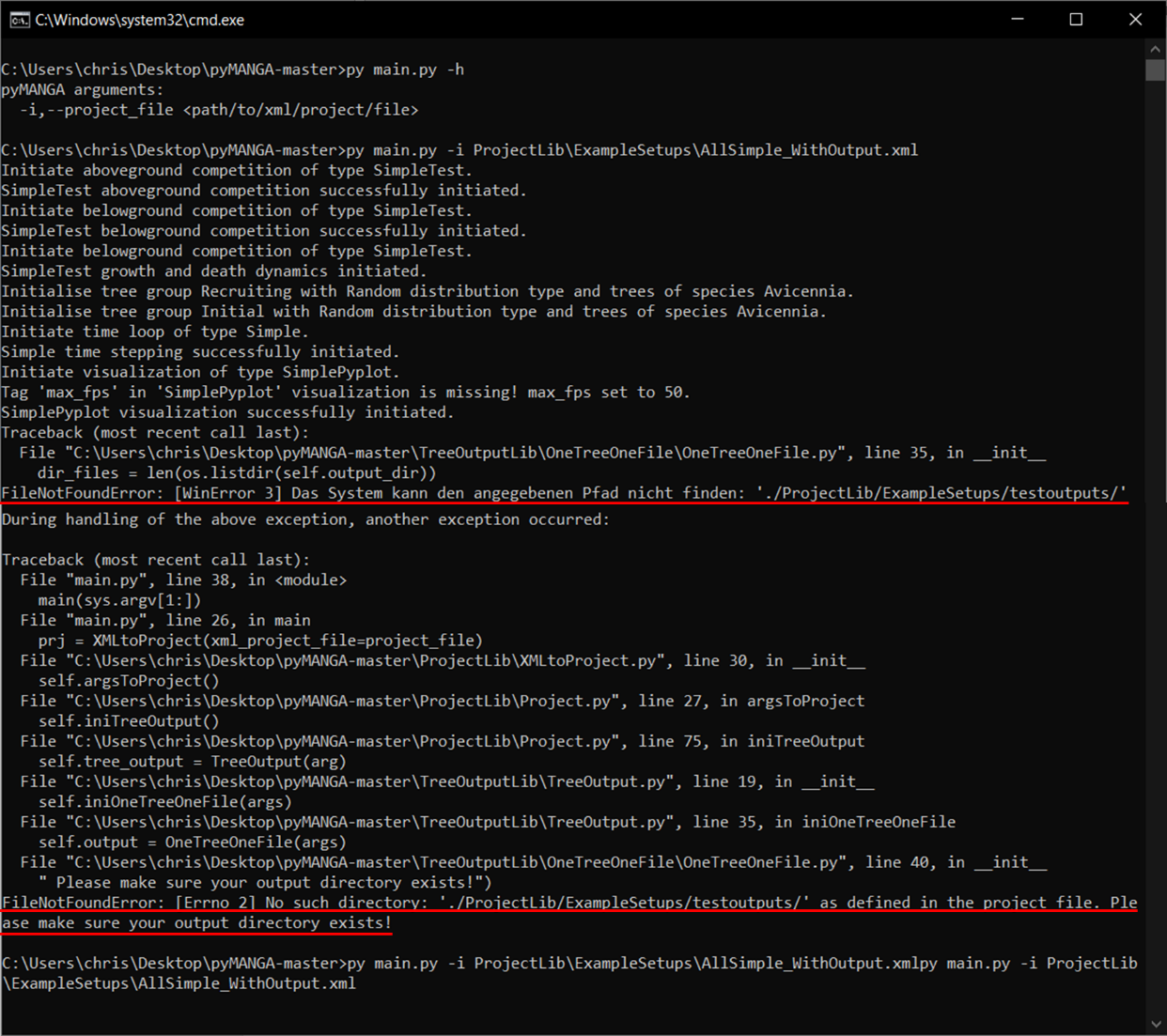
Hinweis: Die Eingabeaufforderung wird im Ordner aufgerufen damit der Ordnerpfad nicht jedes Mal mit eingegeben werden muss. Unter Windows 10 ist dies nur möglich, wenn Sie sich *cmd in Kontextmenü hinzufügen.zip* von der folgenden Internetseite https://www.giga.de/downloads/windows-10/tipps/windows-10-wieder-die-eingabeaufforderung-im-kontextmenue-anzeigen/ herunterladen und wie auf der Seite beschrieben ausführen. Alternativ ist es möglich in Eingabeaufforderung, die Sie in der Windows Suche mit dem Suchbegriff „Eingabeaufforderung“ finden zu nutzen und den vollständigen Dateipfad anzugeben, der in diesem Beispiel wie folgt lautet *C:\Users\chris\Desktop\pyMANGA-master*. Um Ihren Dateipfad herauszufinden machen Sie einen Rechtsklick auf den Ordner *pyMANGA-master* und gehen Sie auf *Eigenschaften*. Hier finden sich die Angaben zum Ort des Ordners an den Sie noch mit eine \ den Namen des Ordners anhängen müssen.



Hier ist zu sehen, dass die *main.py* Datei ausgeführt wurde und auf weitere Eingaben wartet. Damit ist der Start von MANGA geglückt und Sie können ein paar erste Verwendungsbeispiele testen. Dazu können Sie folgenden Code eingeben (siehe Abbildung 8).

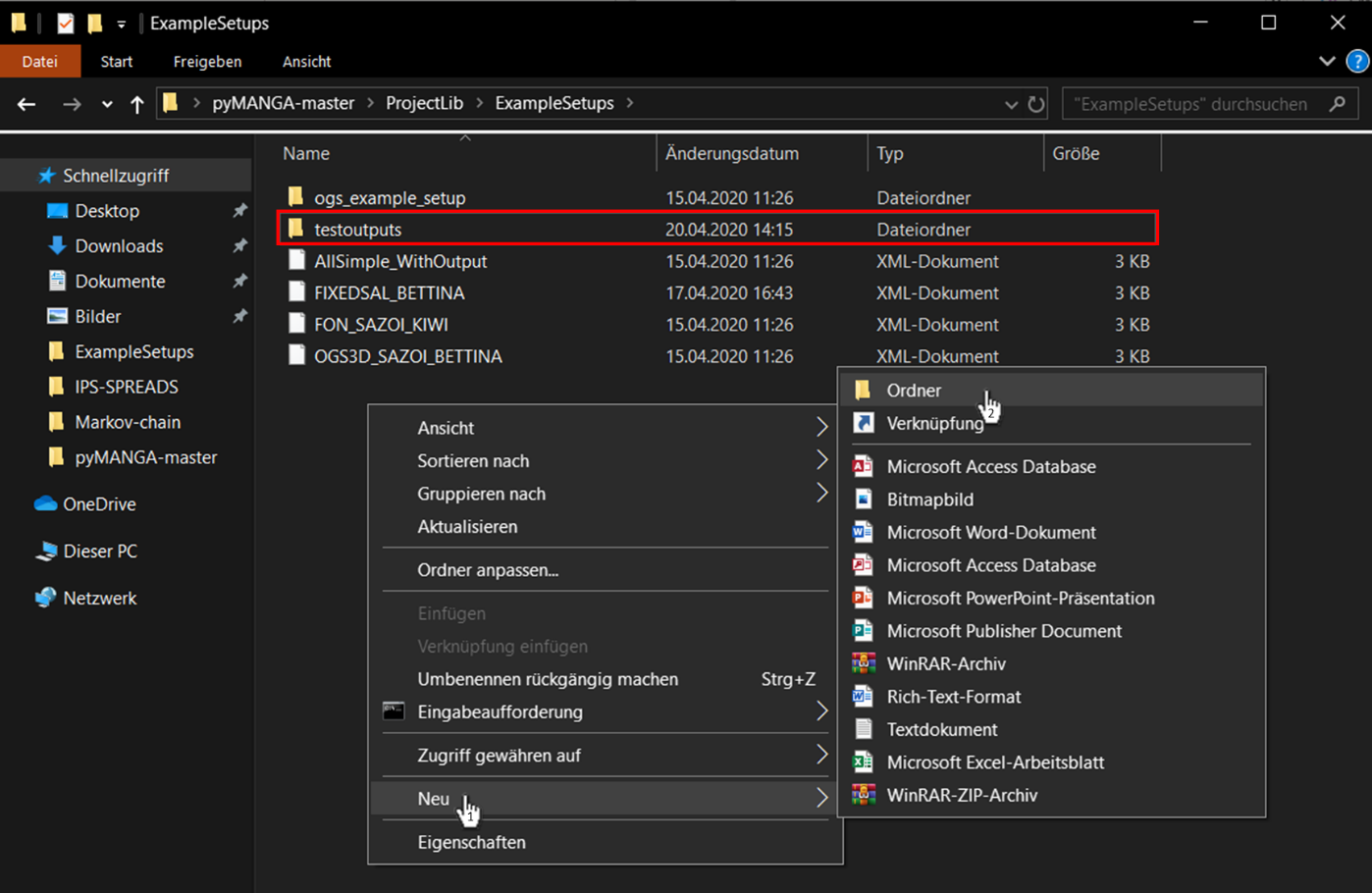
* py main.py -i ProjectLib\ExampleSetups\AllSimple\_WithOutput.xml [5]

*-i* beschreibt dabei den Index bzw. den Pfad der Datei, in der der Input definiert ist, der für dieses Beispiel verwendet werden soll.

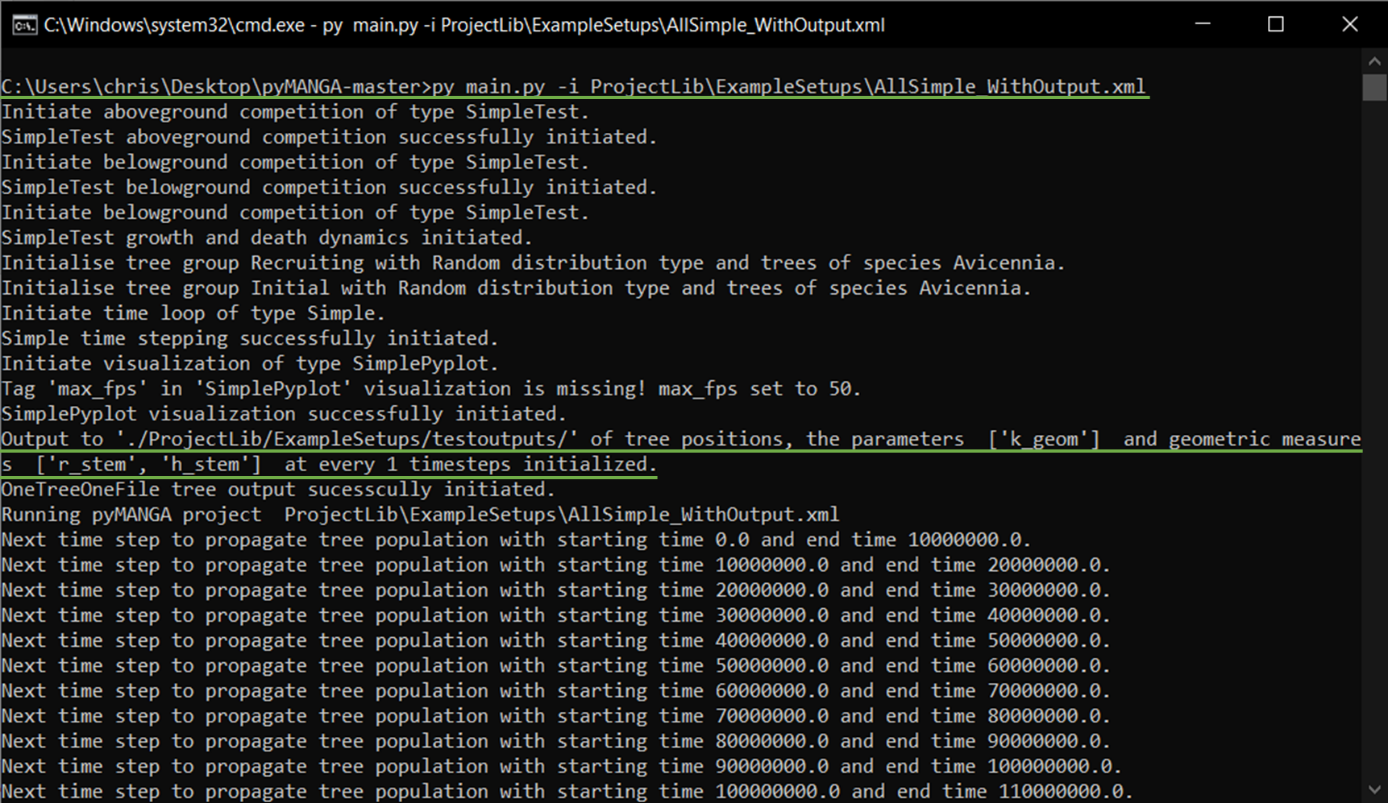


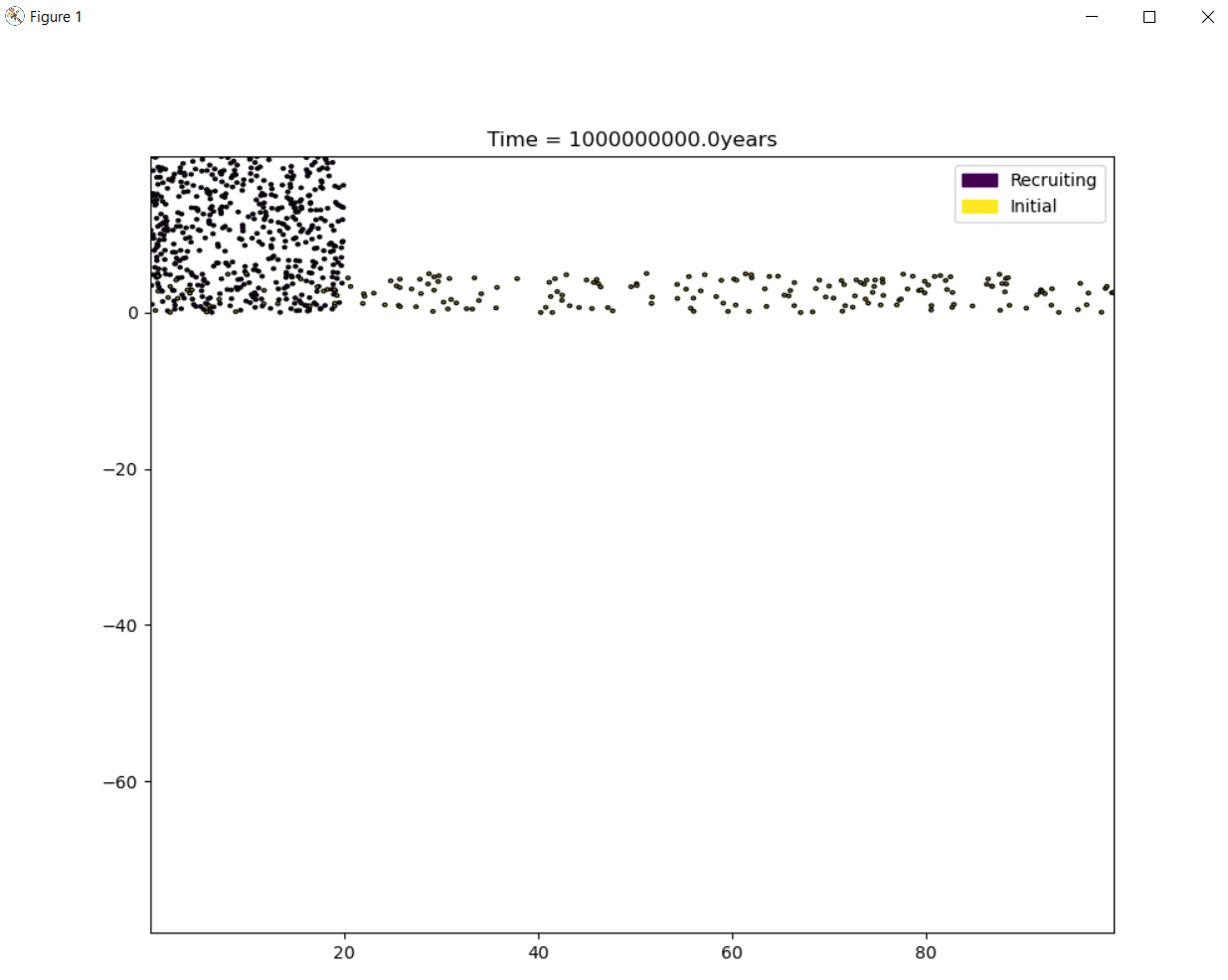
Nach Ausführung des Codes wird eine Fehlermeldung ausgegeben, welche beschreibt, dass ein Ordner Namens *testoutputs* nicht existiert, welcher aber vom Programm benötigt wird, um die erzeugten Daten der Simulation abzuspeichern. Diese Information findet sich in der Datei *AllSimple\_WithOutput.xml*, welche den Input für unser Beispiel definiert. Um diese einsehen zu können, müssen Sie die Datei mit Hilfe des Editors öffnen. Dazu folgen Sie dem in Abbildung 8 angegeben Dateipfad im Ordner *pyMANGA-master* und machen einen Rechtsklick auf die genannte Datei, gehen auf *Öffnen* *mit* und suchen den Editor raus (siehe Abbildung 9).

In der Datei finden sich die rot-markierten Zeilen, welche angeben was vom Programm ausgegeben werden soll, z.B. die Baumhöhe (h\_stem) und wo hin, nämlich in den nicht existenten Ordner *testoutputs*, welcher als Ausgabeort für die Simulationsergebnisse definiert wurde. Demzufolge müssen Sie nun diesen Ordner erstellen. Dazu machen Sie einen Rechtsklick in den Unterordner *C:\Users\chris\Desktop\pyMANGA-master\ProjectLib\ExampleSetups*, klicken auf *Neuen Ordner erstellen* und nennen ihn *testoutputs* (siehe Abbildung 10).



Im Anschluss führen Sie den Code 5 erneut in der Eingabeaufforderung aus. Nun sollte das Programm die erste Simulation starten (siehe Abbildung 11). Es gibt insgesamt input Parameter, welche im MANGA eingestellt werden können. Eine Beschreibung dieser Parameter findet sich auf der folgenden Internetseite auf Englisch <https://jbathmann.github.io/pyMANGA/project_dox__MangaProject__MangaProject.html>. Des Weiteren ist eine Tabelle im Anhang zu finden, welche diese Parameter erläutert. Die Ergebnisse der Simulation werden einmal visuell in einem separaten Fenster dargestellt (siehe Abbildung 12) und in Form von csv Dateien im neu angelegten Ordner *testoutputs*. Damit haben Sie erfolgreich das erste Beispiel durchgeführt.

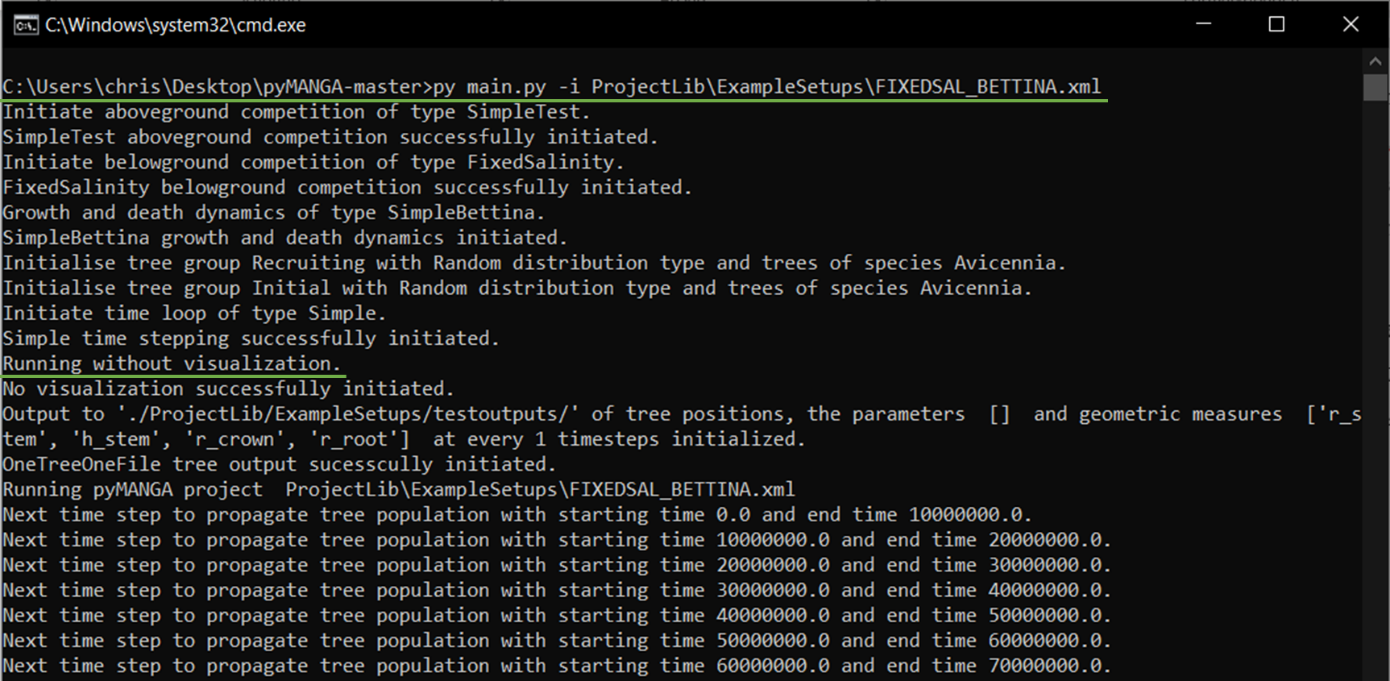




Analog dazu können Sie mit den folgenden Codes zwei weitere Beispiel ausprobieren, in dem andere Input-Varianten definiert sind. Dazu müssen Sie aber zunächst den Ordner *testoutputs* leeren bzw. einen anderen Ordner in den Input-Dateien mit Hilfe des Editors definieren, da das Programm die alten Output-Daten nicht überschreiben kann. Anschließend geben Sie einfach wieder den Code in die Eingabeaufforderung ein.

* py main.py -i ProjectLib\ExampleSetups\FIXEDSAL\_BETTINA.xml [6]
* py main.py -i ProjectLib\ExampleSetups\FON\_SAZOI\_KIWI.xml [7]

Aufgrund von anderen Input-Varianten (in FIXEDSAL\_BETTINA.xml und FON\_SAZOI\_KIWI.xml) wird unteranderem im Beispiel zu Code 6 keine visuelle Darstellung ausgegeben (vergleiche Abbildung 13).

Im Beispiel zu Code 6 gibt es wieder eine visuelle Darstellung. Des Weiteren wurden wiederum andere Parameter verändern. Verschaffen Sie sich mit Hilfe der Internetseite bzw. der Tabelle im Anhang einen Überblick über die Einstellungsvarianten der Input-Parameter, die in den Beispielen verwendet worden und vergleichen Sie sie.