

Hydroinformatik II - SoSe 2024

HyBHW-S2-01-V02: Werkzeuge

Prof. Dr.-Ing. habil. Olaf Kolditz

¹Helmholtz Centre for Environmental Research – UFZ, Leipzig

²Technische Universität Dresden – TUD, Dresden

³Center for Advanced Water Research – CAWR

⁴TUBAF-UFZ Center for Environmental Geosciences – C-EGS, Freiberg / Leipzig

Dresden, 14.06.2024

Zeitplan: Hydroinformatik II - SoSe 2024

Datum	No	Thema	Typ
14.06.2024	2-01	Einführung in die Lehrveranstaltung - Teil 2	L
14.06.2024	2-02	Werkzeuge Tools	L
14.06.2024	2-03	Grundlagen: Kontinuumsmechanik	L
14.06.2024	2-04	Grundlagen: Hydromechanik	L
21.06.2024	2-05	Grundlagen: Partielle Partialgleichungen	L
21.06.2024	2-06	Grundlagen: Näherungsverfahren	L
28.06.2024	2-07	Tools: Compiler, Python, Jupyter (V02)	E
28.06.2024	2-08	Übung: Jupyter Diffusionsprozess	E
05.07.2024*	2-10	Numerik: Finite-Differenzen-Methode (explizit)	L
05.07.2024*	2-11	Numerik: Finite-Differenzen-Methode (implizit)	L
12.07.2024	2-12	Grundlagen: Gerinnehydraulik	L
12.07.2024	2-13	Übung: Gerinnehydraulik	E
19.07.2024	2-14	Ausblick: Grundwassermodellierung	E
19.07.2024	2-15	Klausur/Beleg: Besprechung zur Vorbereitung	L

1 Wiederholung (HyBHW-S1-01-V1/2: Intro/Hydroinformatik)

2 OPAL (Freigabe für registrierte Nutzer)

3 Tools: Compiler-Installation (MinGW) + Übung (HA01)

4 Tools: Python (HA02)

5 Ihre Fragen

The screenshot shows a web browser window with the URL <https://bildungsportal.sachsen.de/opal/auth/RepositoryEntry/2725>. The page header includes the OPAL logo and navigation tabs: Startseite, Lehren & Lernen, Kursangebote, and Hydroinformatik. The main content area displays the course details for 'Hydroinformatik' at TU Dresden, Wintersemester 2020/2021, taught by Olaf Kolditz. The course description states: 'Die Einsatzmöglichkeiten rechnergestützter Software bei der Bearbeitung wissenschaftlicher Fragen im Bereich der Hydrowissenschaften werden vorgestellt. Dies beinhaltet sowohl die Anwendung allgemein verfügbarer als auch die Entwicklung eigener Softwarekomponenten. Die vorgestellten Methoden werden durch geeignete Beispiele erläutert und geübt. Die Studierenden sind in der Lage quantitative Problemstellungen rechnergestützt zu bearbeiten. Dazu gehören auch die Auswahl, der Einsatz und die (Weiter-)Entwicklung von Software bzw. Softwarekomponenten.' Below the description is a 'Linkliste' section with links to 'Lehrewebseite' and 'Videovorlesung'.

Hydroinformatik I - Helmholtz X OPAL - Online-Plattform für A...

Suche

Startseite Lehren & Lernen Kursangebote Hydroinformatik

Hydroinformatik

Hydroinformatik

Linkliste

TU Dresden | Wintersemester 2020 / 2021
Hydroinformatik
Verantwortliche/r: Olaf Kolditz
Hydroinformatik

Die Einsatzmöglichkeiten rechnergestützter Software bei der Bearbeitung wissenschaftlicher Fragen im Bereich der Hydrowissenschaften werden vorgestellt. Dies beinhaltet sowohl die Anwendung allgemein verfügbarer als auch die Entwicklung eigener Softwarekomponenten. Die vorgestellten Methoden werden durch geeignete Beispiele erläutert und geübt. Die Studierenden sind in der Lage quantitative Problemstellungen rechnergestützt zu bearbeiten. Dazu gehören auch die Auswahl, der Einsatz und die (Weiter-)Entwicklung von Software bzw. Softwarekomponenten.

Weitere Informationen anzeigen

Linkliste

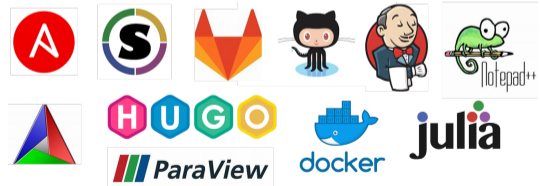
- Lehrewebseite
- Videovorlesung

Rückblick letzte Veranstaltung: Tools

- 1 Software
- 2 Compiler / Interpreter
- 3 Dokumentation (Repositories) / Visualisierung



- Editor: Notepad++, ...
- Compiler: C++, MinGW, (Qt), ...
- Skripte: Python (Jupyter), ...
- Visualisierung: ParaView, ...



Werkzeuge

- Editor: Notepad++
- Compiler: MinGW (C++)
- Python: Simulation und (einfache) Grafik
- Jupyter-Notebook: Workflows
- Visualisierung: ParaView
- Repository: Git

- Editor: Notepad++
- Compiler: MinGW (C++)
- Python: Simulation und (einfache) Grafik
- Jupyter-Notebook: Workflows
- Visualisierung: ParaView
- Repository: Git



MinGW64



Übung

Compiler Installation

- EX01

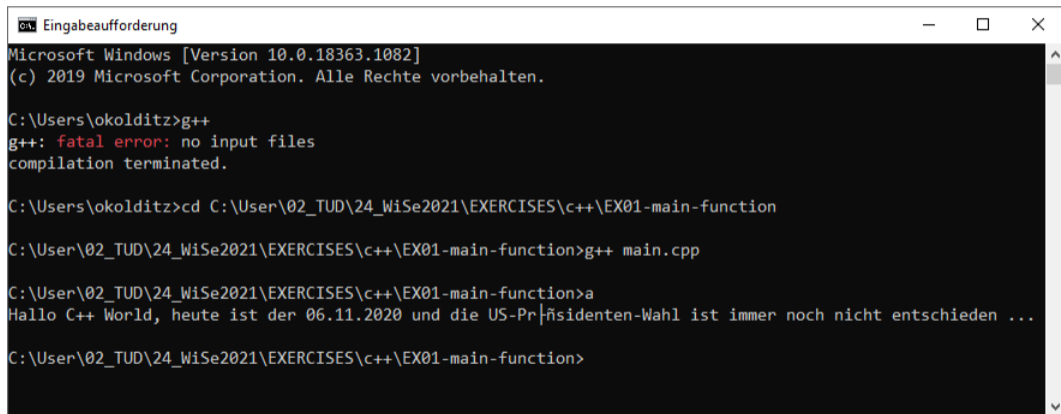
- Anleitung für die Installation (siehe Lehre-Webseite), automatischer Download: https://www.ufz.de/export/data/2/244688_mingw-installation-2020.pdf
 - Voting (als Chat (privat)):
 - "+": hat funktioniert
 - "+-": hat noch nicht ganz funktioniert, weiß aber im Prinzip, wie es geht
 - "-": hat nicht funktioniert, weiß nicht wie es funktioniert
 - >> Übung: EX01 (von der Webseite runter laden, später machen wir dies über GitHub)
-
- DOS Kommandos / BATch Befehle: https://www.script-example.com/themen/cmd_Batch_Befehle.php

```
1 #include <iostream>
2 using namespace std;
3 int main()
4 {
5     std::cout << "Hallo C++ World, heute ist der 06.11.2020 und die US-
6     Praesidenten-Wahl ist immer noch nicht entschieden ..." << std::endl;
7     return 0;
8 }
```

Listing: C++ example of a main function

C++ Compiler Installation

Übung EX01: Step-by-step



```
Microsoft Windows [Version 10.0.18363.1082]
(c) 2019 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\okolditz>g++
g++: fatal error: no input files
compilation terminated.

C:\Users\okolditz>cd C:\User\02_TUD\24_WiSe2021\EXERCISES\c++\EX01-main-function

C:\User\02_TUD\24_WiSe2021\EXERCISES\c++\EX01-main-function>g++ main.cpp

C:\User\02_TUD\24_WiSe2021\EXERCISES\c++\EX01-main-function>a
Hallo C++ World, heute ist der 06.11.2020 und die US-Präsidenten-Wahl ist immer noch nicht entschieden ...

C:\User\02_TUD\24_WiSe2021\EXERCISES\c++\EX01-main-function>
```

C++ Compiler Installation

Übung EX01: BATch file

```
1  REM comment out line like this
2  REM ECHO OFF
3  g++
4  cd C:\User\02_TUD\24_WiSe2021\EXERCISES\c++\EX01-main-function
5  g++ main.cpp
6  a.exe
7  dir
```

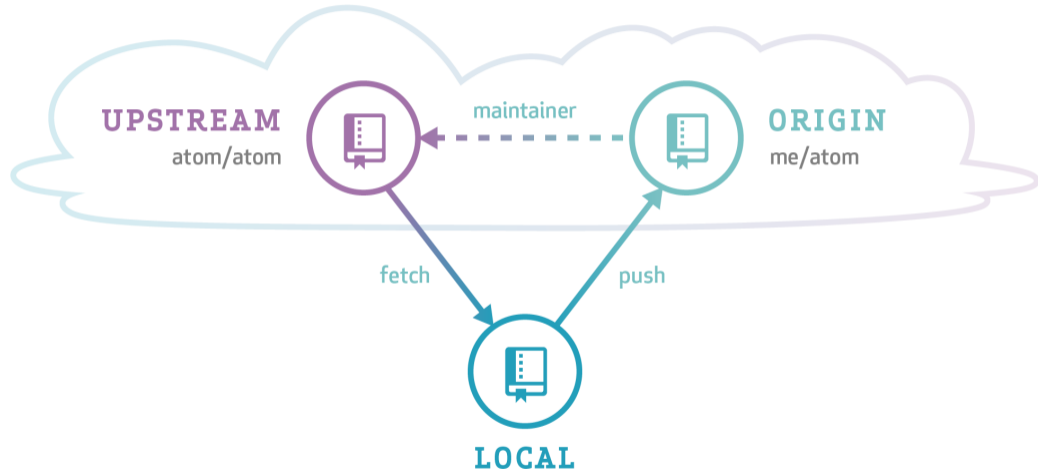
length: 139 lines Ln: 1 Col: 31 Sel: 0|0 Windows (CR LF) UTF-8 INS

GitHub

- "GitHub ist ein netzbasierter Dienst zur Versionsverwaltung für Software-Entwicklungsprojekte ..."
- Webseite: <https://github.com/>
- Vorteil: Webbasiert (und damit Plattform-unabhängig)
- ... wir nutzen GitHub zum archivieren unserer Übungen (>> Demo)
- Webseite:
<https://github.com/OlafKolditz>
<https://github.com/OlafKolditz/HYDROINFORMATIK-II>

GitHub





OlafKolditz / HYDROINFORMATIK-I

Unwatch 1 Star 0 Fork 0

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

main 1 branch 0 tags

Go to file Add file Code

OlafKolditz Add files via upload d7fc244 6 minutes ago 1 commits

EX01-main-function.cpp Add files via upload 6 minutes ago

About

A teaching repository for the Hydroinformatics I Course at TU Dresden

Übung GitHub

- EX08a Container

siehe: Git installieren (DOCX)

Zusammenfassung:

- Git muss installiert sein <https://git-scm.com/book/en/v2/Getting-Started-Installing-Git>
- Repository clonen
- Änderungen vornehmen (edit)
- Änderungen mitteilen (git commit)
- Änderungen hochladen (git push)
- Änderungen werden geprüft bestätigt oder abgewiesen (code review)

Quellen: <https://www.ionos.de/digitalguide/websites/webentwicklung/git-tutorial/>

Übung GitHub

- EX01

```
1 echo "# HYDROINFORMATIK-II" >> README.md
2 git init
3 git add README.md
4 git commit -m "first commit"
5 git branch -M main
6 git remote add origin
7   https://github.com/OlafKolditz/HYDROINFORMATIK-II.git
8 git push -u origin main
```

Listing: Creating Git repository on the command line


```
1 git remote add origin  
2   https://github.com/OlafKolditz/HYDROINFORMATIK-II.git  
3 git branch -M main  
4 git push -u origin main
```

Listing: Creating new from an existing Git repository on the command line

Python

- "Python is a programming language that lets you work more quickly and integrate your systems more effectively."
- Webseite: <https://www.python.org>
- Vorteil: funktioniert auf allen Rechnern (>> Demo)





The screenshot shows the Python.org website with a dark blue header and a lighter blue navigation bar. The main content area is dark blue with yellow and white text and graphics. The navigation bar includes links for About, Downloads, Documentation, Community, Success Stories, News, and Events. The main content area features a large yellow button for downloading Python 3.8.3 for Windows, with links for other operating systems, pre-releases, and Docker images. A graphic of two parachutes carrying boxes is on the right.

Python

PSF

Docs

PyPI

Jobs

Community

python™

Donate

Search

GO

Socialize

About

Downloads

Documentation

Community

Success Stories

News

Events

Download the latest version for Windows

Download Python 3.8.3

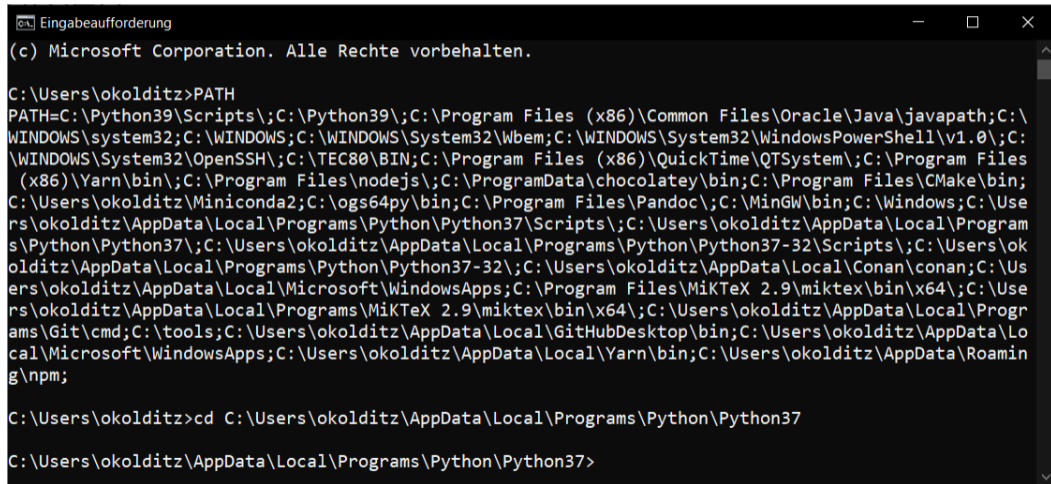
Looking for Python with a different OS? Python for [Windows](#), [Linux/UNIX](#), [Mac OS X](#), [Other](#)

Want to help test development versions of Python? [Pre-releases](#), [Docker images](#)

Looking for Python 2.7? See below for specific releases



PATH



```
Eingabeaufforderung
(c) Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\okolditz>PATH
PATH=C:\Python39\Scripts\;C:\Python39\;C:\Program Files (x86)\Common Files\Oracle\Java\javapath;C:\
WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem;C:\WINDOWS\System32\WindowsPowerShell\v1.0\;C:
\WINDOWS\System32\OpenSSH\;C:\TEC80\BIN;C:\Program Files (x86)\QuickTime\QTSystem\;C:\Program Files
(x86)\Yarn\bin\;C:\Program Files\nodejs\;C:\ProgramData\chocolatey\bin;C:\Program Files\CMake\bin;
C:\Users\okolditz\Miniconda2;C:\ogs64py\bin;C:\Program Files\Pandoc\;C:\MinGW\bin;C:\Windows;C:\Use
rs\okolditz\AppData\Local\Programs\Python\Python37\Scripts\;C:\Users\okolditz\AppData\Local\Program
s\Python\Python37\;C:\Users\okolditz\AppData\Local\Programs\Python\Python37-32\Scripts\;C:\Users\ok
olditz\AppData\Local\Programs\Python\Python37-32\;C:\Users\okolditz\AppData\Local\Conan\conan;C:\Us
ers\okolditz\AppData\Local\Microsoft\WindowsApps;C:\Program Files\MiKTeX 2.9\miktex\bin\x64\;C:\Use
rs\okolditz\AppData\Local\Programs\MiKTeX 2.9\miktex\bin\x64\;C:\Users\okolditz\AppData\Local\Progr
ams\Git\cmd;C:\tools;C:\Users\okolditz\AppData\Local\GitHubDesktop\bin;C:\Users\okolditz\AppData\Lo
cal\Microsoft\WindowsApps;C:\Users\okolditz\AppData\Local\Yarn\bin;C:\Users\okolditz\AppData\Roamin
g\npm;

C:\Users\okolditz>cd C:\Users\okolditz\AppData\Local\Programs\Python\Python37

C:\Users\okolditz\AppData\Local\Programs\Python\Python37>
```

Python: Installation

The screenshot shows a Windows File Explorer window for the user 'okolditz'. The address bar indicates the path: 'Dieser PC > Windows (C:) > Benutzer > okolditz'. The left sidebar shows the navigation pane with 'Dieser PC' selected. The main area displays a list of folders and files with columns for Name, Änderungsdatum, Typ, and Größe. The folders listed include .android, .cache, .cisco, .conan, .conda, .config, .fontconfig, .idlerc, .ipython, .jupyter, .matplotlib, .openjfx, .ssh, 3D-Objekte, and AppData. The status bar at the bottom indicates '37 Elemente'.

Name	Änderungsdatum	Typ	Größe
.android	20.10.2020 20:07	Dateiordner	
.cache	08.11.2020 00:26	Dateiordner	
.cisco	18.02.2019 15:21	Dateiordner	
.conan	15.01.2020 13:34	Dateiordner	
.conda	07.11.2020 01:29	Dateiordner	
.config	28.09.2018 13:02	Dateiordner	
.fontconfig	31.03.2022 21:38	Dateiordner	
.idlerc	03.10.2018 18:07	Dateiordner	
.ipython	15.09.2019 19:48	Dateiordner	
.jupyter	07.05.2021 10:34	Dateiordner	
.matplotlib	08.11.2020 00:27	Dateiordner	
.openjfx	13.05.2020 17:13	Dateiordner	
.ssh	29.04.2022 16:42	Dateiordner	
3D-Objekte	15.05.2021 21:12	Dateiordner	
AppData	15.05.2021 20:49	Dateiordner	

Python: Module installieren

matplotlib

- 1 `python -m pip install -U pip`
- 2 `python -m pip install -U matplotlib`

Listing: Installieren von der Konsole

```
Eingabeaufforderung
Microsoft Windows [Version 10.0.18363.1082]
(c) 2019 Microsoft Corporation. Alle Rechte vorbehalten.

C:\Users\okolditz>python -m pip install -U pip
DEPRECATION: Python 2.7 reached the end of its life on January 1st, 2020. Please upgrade your Python as Python 2.7 is no longer maintained. pip 21.0 will drop support for Python 2.7 in January 2021. More details about Python 2 support in pip can be found at https://pip.pypa.io/en/latest/development/release-process/#python-2-support pip 21.0 will remove support for this functionality
Requirement already up-to-date: pip in c:\python27\lib\site-packages (20.2.4)

C:\Users\okolditz>python -m pip install -U matplotlib
DEPRECATION: Python 2.7 reached the end of its life on January 1st, 2020. Please upgrade your Python as Python 2.7 is no longer maintained. pip 21.0 will drop support for Python 2.7 in January 2021. More details about Python 2 support in pip can be found at https://pip.pypa.io/en/latest/development/release-process/#python-2-support pip 21.0 will remove support for this functionality
Collecting matplotlib
  Downloading matplotlib-2.2.5-cp27-cp27m-win_amd64.whl (8.7 MB)
    |#####| 8.7 MB 3.3 MB/s
Requirement already satisfied, skipping upgrade: cyycler>=0.10 in c:\python27\lib\site-packages (from matplotlib) (0.10.0)
Requirement already satisfied, skipping upgrade: six>=1.10 in c:\python27\lib\site-packages (from matplotlib) (1.12.0)
Requirement already satisfied, skipping upgrade: pytz in c:\python27\lib\site-packages (from matplotlib) (2019.3)
Requirement already satisfied, skipping upgrade: numpy>=1.7.1 in c:\python27\lib\site-packages (from matplotlib) (1.17.0)
```

matplotlib

Plot types Examples Tutorials Reference User guide Develop Release notes

Search the docs ...

- plot(x, y)
- scatter(x, y)
- bar(x, height) / barh(y, width)
- stem(x, y)
- step(x, y)
- fill_between(x, y1, y2)
- imshow(Z)
- pcolormesh(X, Y, Z)
- contour(X, Y, Z)
- contourf(X, Y, Z)
- barbs(X, Y, U, V)
- quiver(X, Y, U, V)
- streamplot(X, Y, U, V)
- hist(x)
- boxplot(X)
- errorbar(x, y, yerr, xerr)
- pie(labels)


Plot types

Overview of many common plotting commands in Matplotlib.


Note that we have stripped all labels, but they are present by default. See the [gallery](#) for many more examples and the [tutorials](#) page for longer examples.

Basic


Basic plot types, usually y versus x.




plot(x, y)



scatter(x, y)



bar(x, height) / barh(y, width)

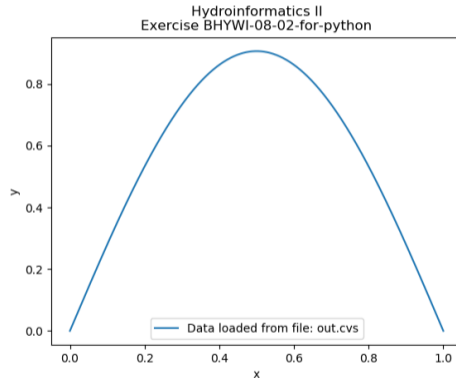
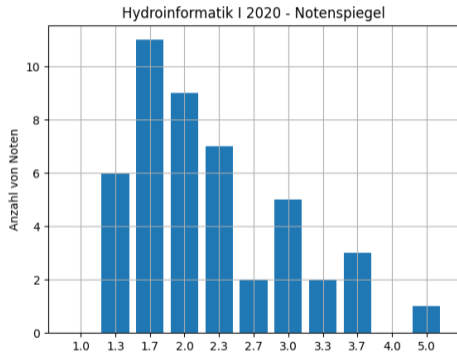


stem(x, y)

On this page

- Basic
- Plots of arrays and fields
- Statistics plots
- Unstructured coordinates

Python: Plotting (matplotlib)



Python: Plotting bar charts

Übung: Hydroinformatik-I-Noten.py

```
1 from matplotlib.ticker import FuncFormatter
2 import matplotlib.pyplot as plt
3 import numpy as np
4
5 year = np.arange(11)
6 publications = [1,1,7,4,8,7,6,7,3,2,3]
7 fig, ax = plt.subplots()
8 ax.set_title('Hydroinformatik I 2019 - Notenspiegel')
9 ax.set_ylabel('Anzahl von Noten')
10 plt.bar(year, publications)
11 plt.xticks(year, ('1.0', '1.3', '1.7', '2.0', '2.3', '2.7', '3.0', '3.3', '3.7',
12                  '4.0', '5.0'))
13 plt.grid(True)
14 plt.show()
```

Listing: Python example for bar charts: Hydroinformatik-I-Noten.py

Diffusionsgleichung: Analytische Lösung Tutorial: Abschnitt 2.2

- Repo: BHYWI-08-02-E-Script
- C++ Variante
- C++ / Python Variante
- Python Variante

MinGW + Python: Function plotter (analytical solution) #1

Übung: BHYWI-08-02-E-Script: Skript

```
1 echo Compilation
2 g++ main.cpp
3 echo Execution
4 a.exe
5 echo Ploting
6 data_from_file.py
7 echo End
```

Listing: Script file for entire workflow

MinGW + Python: Function plotter (analytical solution) #2

Übung: BHYWI-08-02-E-Script: C++

```
1 #include <cmath>
2 #include <fstream>
3 #define PI 3.14159265358979323846
4 int main(int argc, char *argv[])
5 {
6     //1-Definitionen
7     int numPoints = 1000;
8     double x,y,alpha=1.,t=0.01;
9     std::ofstream out_file;
10    out_file.open("out.csv");
11    //2-Berechnung
12    for (int i = 0; i < numPoints+1; ++i)
13    {
14        x = double(i)/double(numPoints);
15        y=sin(PI*x) * exp(-alpha*PI*PI*t);
16        out_file << x << "," << y << std::endl;
17    }
18    //3-Ausgabe
19 }
```

Listing: C++ program for analytical solution

MinGW + Python: Function plotter (analytical solution) #3

Übung: BHYWI-08-02-E-Script: File reading and data plotting

```
1 import matplotlib.pyplot as plt
2 import csv
3
4 x = []
5 y = []
6 with open('out.csv', 'r') as csvfile:
7     plots = csv.reader(csvfile, delimiter=',')
8     for row in plots:
9         x.append(float(row[0]))
10        y.append(float(row[1]))
11
12 plt.plot(x,y, label='Data loaded from file: out.csv')
13 plt.xlabel('x')
14 plt.ylabel('y')
15 plt.title('Hydroinformatics II\nExercise BHYWI-08-02-for-python')
16 plt.legend()
17 plt.savefig("diffusion-equation.png.png")
18 plt.show()
```

Listing: File reading and data plotting

```
1 import math
2 import matplotlib.pyplot as plt
3 PI = 3.14159265358979323846
4 numPoints = 10
5 alpha = 1.0
6 t = [0.1,0.5,1.0,2.0]
7 x = []
8 y = []
9 for n in t:
10     for i in range(0,numPoints+1):
11         x.append(float(i)/float(numPoints))
12         y.append(math.sin(PI*x[i]) * math.exp(-alpha*n*n))
13     plt.plot(x,y,color='red',marker="o")
14     x = []
15     y = []
16 plt.xlabel('x')
17 plt.ylabel('u')
18 plt.savefig("diffusion-equation.png")
19 plt.show()
```

Listing: Analytical solution

??

Jupyter

- Jupyter Notebook
- Jupyter Lab
- Browser-basiert

- "The Jupyter Notebook · The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, ..."
- Webseite: <https://jupyter.org/>
- Vorteil: funktioniert auf allen Rechnern
- ... ein Teil unserer (neuen) Übungen machen wir mit Jupyter Notebooks (>> Demo)



Jupyter: Example

HyBHW-1-01-02

localhost:8888/notebooks/HyBHW-1-01-02-jymb

jupyter HyBHW-1-01-02 (autosaved)

File Edit View Insert Cell Kernel Help

Trusted Python 2

TECHNISCHE UNIVERSITÄT DRESDEN

Professur für Angewandte Umweltsystemanalyse an der TU Dresden
Prof. Dr.-Ing. habil. Olof Klotze
Hydroinformatik (HyBHW-1-01)
[Lehre-Webseite](#)

Exercise 2 - Figures

```
In [2]: from matplotlib import FigureFormatter
import matplotlib.pyplot as plt
import numpy as np

year = np.arange(11)
publications = [1,1,7,4,8,7,6,7,3,2,3]

fig, ax = plt.subplots()

ax.set_title('Hydroinformatik I 2019 - Notenspiegel')
ax.set_ylabel('Anzahl von Noten')

plt.bar(year, publications)
plt.xticks(year, ('1.0', '1.3', '1.7', '2.0', '2.3', '2.7', '3.0', '3.3', '3.7', '4.0', '5.0'))
plt.grid(True)
plt.show()
```

Hydroinformatik I 2019 - Notenspiegel

Anzahl von Noten	Anzahl von Jahren
1.0	1
1.3	1
1.7	7
2.0	4
2.3	8
2.7	7
3.0	6
3.3	7
3.7	3
4.0	2
5.0	3

jupyterlab

localhost:8888/lab

File Edit View Run Kernel Tabs Settings Help

d2022-jupyter-lab-task-g1-k

Python 3

Benchmarks: HM processes

Benchmark 1: Sneddon (Opening profile)

Running OGS Simulation

OpenGeoSys
jupyterlab

Benchmark 2: Propagating straight fracture (pressure and crack length)

```
[7]: plt.plot(np.arange(5)/4, x_c, np.arange(length)/w, "black", label="Closed form solution")
plt.plot(np.arange(x_0/0.75)/x_c, x_c, np.arange(length_0/75)/[1])/w, "color", fillstyle="none", label="Lab
```

Python 3 | Idle

Saving completed Mode: Command Ln 1, Col 1 d2022-jupyter-lab-task-g1-jupyter