

Jupyter Notebooks für Lehrveranstaltungen



Gottfried Mandlbürger

gottfried.mandlbuerger@geo.tuwien.ac.at

TU Wien, Institut für Geodäsie und Geoinformation

Forschungsgruppe Photogrammetrie (E120.7)

mit Beiträgen von: Anna Iglseider, Elias Wimmer, Florian Jäger



Tools für digitale Lehre

▪ Elektronische Lernplattformen

- Udemy, Learningbank, SAP Litmos, eloomi,
- eCampus Steiermark: TU Graz, Joanneum....
- eCampus/OpenOlat: Innsbruck: UIBK, Med IBK... (Apache)
- Moodle Learning Management System (GNU GPL)
- Ilias Open Source Learning Management System (GNU GPL)

▪ Videokonferenztools

- Adobe Connect
- Zoom
- GoToMeeting
- Google Hangouts / Google Meet
- MS Teams,
- BigBlueButton (GNU LGPL)
- Jitsi / Jisti Meet (Apache)

TU Wien: TUWEL+Zoom

Hintergrund @ GEO-TUW

- **Geodäsie und Geoinformation**
 - Bachelor-Studiengang: 35+ Teilnehmer:innen
 - Masterprogramm: 20+ Teilnehmer:innen
- **Umweltingenieurwesen (UIW)**
 - Bachelor-Studiengang: **150+** Teilnehmer:innen (laufend seit 2019)
 - Masterstudiengang: **10+** (im Aufbau seit 2021)
- **VU: Einführung in die Programmierung für Geodäsie und Umweltingenieurwesen**
 - **Pflichtveranstaltung** in der Anfangsphase des Bachelorstudiums (GEO/UIW: 1./3. Sem.)
 - **Gemeinsamer Kurs für GEO und UIW** 180+ Teilnehmer:innen
 - Programmiersprache **Python**
 - **leicht** zu lernen
 - **Skripting** und voll **funktionsfähige** Programmiersprache
 - **Zielsetzung:**
 - **Theorie:** Grundlegende Programmierkonzepte (Datentypen, Kontrollstrukturen, IO, Plotten...)
 - **Praktische Beispiele** für GEO und UIW

Ziele - technische Lösungen - weiterführende Fragen

- Ziel
 - Vollständig digitalisierte Umsetzung eines Programmiervorlesung mit Übungen (VU) mit **180+ Teilnehmer:innen**
 - Ermöglichung von “echtem Hybridbetrieb” → **Hörsaal und (!) online**
- (Quelloffene) technische Umsetzung
 - E-Learningplattform → **Moodle/ TUWEL**
 - Zentrale interaktive Programmierumgebung → **JupyterHub/Notebooks**
 - Videokonferenztool (nicht quelloffen!) → **Zoom**
- Weiterführende Fragen
 - Auch für fachorientierte, nicht-programmier LVAs vorteilhaft?
 - Welche Einsatzmöglichkeiten?

TUW-weite Infrastruktur für digitale Lehre

JupyterHub

Jupyter-Notebooks

Automatische Bewertung von Code-Abgaben: nbgrader

TU.it



TUWEL/Moodle

Zentrale Schnittstelle für Studenten

Zugang zu JupyterHub - Rückmeldung der Ergebnisse

Online-Selbsttests (Quizz) und Online-Tests

Dig Teaching



Didaktisches Konzept

Pre-recorded Videos (TUbe) – Self-paced learning

Selbstüberprüfung: Theorie (TUWEL) / Praxis (Jupyter)

Hybride Vorlesungseinheiten mit invertiertem Klassenzimmer

GEO

Didaktisches Konzept I

- **Theorievermittlung im eigenen Tempo (self-paced learning)**
 - Einteilung des Lernstoffs in **Kapitel** (1 Kapitel = 1 Vorlesungswoche)
 - Gliederung der einzelnen Kapitel in **5-7 Unterabschnitte**
 - 1 **Jupyter Notebook** pro **Unterabschnitt** (**Theorie und Programmierbeispiele**)
 - 1 **voraufgezeichnetes Video** pro **Unterabschnitt** (**Länge ~ 10 min**)
 - **Hosting** von Videos auf einer zentralen **TU Wien-Plattform** (**TUbe**)
 - Bereitstellung von **Links zu Videos** im E-Learning-Kurs (**Moodle**)
- **Selbstüberprüfung: Anreiz über Bonuspunktsystem (15% extra)**
 - **Theorie**: In **Moodle** implementierte **Quizfragen** (1 Quiz/Woche=1 Quiz/Kapitel)
 - **Single-** und **Multiple-Choice**-Tests
 - **Lückentext** und **Drag&Drop**-Aufgaben
 - **Praxis**: Übungen, als **Aufgaben** im **JupyterHub** implementiert (3-4 Bsp. pro Woche)
 - **Automatische Überprüfung** (nbgrader)
 - **Sichtbare** und **nicht sichtbare** Überprüfungszellen

Didaktisches Konzept II und Benotung

▪ Nachbesprechung statt Vorlesung

- Hybrid-Modus möglich:
 - Hörsaal (vorzugsweise!!!)
 - per Telefonkonferenz (während der Pandemie erforderlich)
- Zusammenfassende Darstellung/Wiederholung der Theorie
- Referenzlösung der Aufgabe (Live Coding)
- Inverted classroom Konzept (Studierende schlagen Lösungen vor)
- Diskussion mit den Studierenden (Probleme, Codierungsdetails, Programmierstile, ...)

▪ Benotung

- Praktische Aufgaben – Assignments (60%)
 - 3 Assignments (a) (Autom. Korrektur, mehrheitlich verborgene Überprüfungszellen)
 - Automatische Benotungs- und Feedbacksystem (nbrader)
- Theorie (40%):
 - Moodle-Online-Test (SC/MC, kprim, drag&drop, Lückentext, Reihenfolge)
- Self assessment (15% extra)
 - Theorie und Praxis

Technisches Konzept

- **Moodle/TUWEL als zentrale Basis**
 - Kommunikation über **Foren**
 - **Links zu voraufgezeichneten Videos**
 - **Zugang zu JupyterHub**
 - Automatische Authentifizierung über TUW-Anmeldeinformationen
 - Zugang zu Jupyter Notebooks (=Skriptum)
 - **Quiz zur Selbsteinschätzung**
- **JupyterHub mit integriertem JupyterLab**
 - **Individueller Bereich** für Studierende am Server (d.h. eigene Home-Directories)
 - **Vorkonfigurierte Ordnerstruktur** mit dem gesamten Lernmaterial (Notebooks)
 - **Zugang zu Assignments**
 - **Zugang zu** (automatisch und/oder manuell generiertem) **Feedback**
- **(Halb)automatische Übertragung von JupyterHub-Auswertungen Moodle**

Automatische Benotung der Assignments

Jupyter Notebook → Übungsblatt

Digitale Lehre in Geodäsie und Geoinformation mit quelloffenen Werkzeugen (120.999)

Übungsblatt zu Kapitel 4: Funktionen

Aufgabe 1

Schreiben Sie eine Funktion `avg`, die als Parameter eine Zahlenliste übernimmt und den Mittelwert der Zahlenliste zurückliefert.

Angabe

```
#####  
# Aufgabe 1:  
#####  
  
## BEGIN SOLUTION  
def avg(liste):  
    summe=0  
    for wert in liste:  
        summe+=wert  
    return summe/len(liste)  
## END SOLUTION
```

Referenzlösung

```
#####  
# Sichtbare Überprüfung 1 zu Aufgabe 1  
#####  
import types
```

```
assert isinstance(avg, types.FunctionType)  
print("Bravo: Funktion richtig definiert")
```

Bravo: Funktion richtig definiert

Sichtbare Überprüfungszelle

```
#####  
# Sichtbare Überprüfung 2 zu Aufgabe 1  
#####
```

```
assert type(avg([1,2,3])) == float  
print('Richtig! Die Funktion avfg liefert einen float Wert zurück')
```

Richtig! Die Funktion avfg liefert einen float Wert zurück

Sichtbare Überprüfungszelle

```
#####  
# Verborgene Überprüfung zu Aufgabe 1  
#####
```

```
## BEGIN HIDDEN TESTS  
l, ref = list(range(10)), 4.5  
assert avg(l) == ref  
print('Richtig: Der Mittelwert der Zahlen von 0-9 beträgt {:.1}.format(ref))  
## END HIDDEN TESTS
```

Unsichtbare Überprüfungszelle

Richtig: Der Mittelwert der Zahlen von 0-9 beträgt 4.5.

Virtueller Rundgang

<https://edu.tuwien.ac.at/course/view.php?id=133>

- Close course index
- Ankündigungen
- Diskussionsforum
- ▼ **Einstieg**
 - Fragebogen
 - Einstufungstest
- ▼ **JupyterHub**
 - JupyterHub
- ▼ **Weitere Unterlagen**
 - Cheatsheet
- ▼ **Kapitel 1 - Grundlagen**
 - Self Assessment - Kapitel 1
- ▼ **Kapitel 2 - Container**
 - Self Assessment - Kapitel 2
- ▼ **Kapitel 3 - Kontrollstrukturen**
 - Self Assessment - Kapitel 3
- ▼ **Kapitel 4 - Funktionen**
 - Self Assessment - Kapitel 4
- ▼ **Abschluss**
 - Abschlusstest
 - Feedback zum Kurs
- ▼ **Weiterführende Kurse**

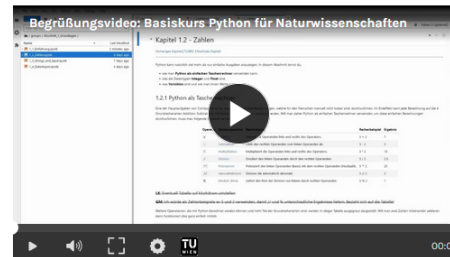
Basiskurs Python für Naturwissenschaften

Course Settings Participants Grades Reports More

▼ Collapse all

- FORUM Ankündigungen
- FORUM Diskussionsforum

▼ Einstieg



Herzlich Willkommen im **Basiskurs Python für Naturwissenschaften!** In diesem Kurs kannst du selbstständig die **Programmiersprache Python** lernen. Für die Programmierbeispiele verwenden wir **JupyterHub**, eine Umgebung mit der du direkt in deinem Browser programmieren kannst. Zum Start schau dir am Besten unser **Begrüßungsvideo** an. In diesem werden alle relevanten Schritte und Funktionen erklärt, damit du zu allen Inhalten navigieren kannst. Sollte dann noch etwas unklar sein, schreibe uns einfach im **Diskussionsforum!** Viel Erfolg!

Zusammenfassung

- **Volldigitalisierung** in der Lehre
- Vollständig digitalisierte **Vorlesung und Übung** (VU)
- Zentrale Jupyter-Notebook-Serverumgebung (**JupyterHub**)
- **Integration** von Jupyter-Notebooks in **Moodle** bzw. TUWEL
- **Automatische Bewertung/Beurteilung** praktischer Code-Aufgaben mit Feedback.
- **Selbstgesteuerte Lerngeschwindigkeit** über kurze Videos
- **Flipped Classroom** Konzept während der wöchentlichen Vorlesungen
- **Self Assessments** in Theorie und Praxis
- **Interaktive Jupyter Notebooks** (widgets) auch einsetzbar zur Unterstützung von **fachlich-geodätischen Inhalten**