

Blended Learning mit Jupyter Notebooks

gefördert durch den Multimedafonds der TU Dresden 2019/20

Dr. Jens Brose

Fakultät Physik

24. November 2022

Programmierkurs: Python in der Physik

Fakultativer Blockkurs

Kurs

Programmierkurs: Python in der Physik 2022

Fakultät Physik | Wintersemester 2021 / 2022

Programmierkurs: Python in der Physik 2022

Verantwortlicher: Jens Brose

Block - Programmierkurs Python für Studierende der Physik

Blended Learning

- im Selbststudium (flipped classroom)
 - Aneignung der Grundlagen der Programmiersprache python
 - Anwendung zur Lösung physikalischer Probleme
 - Integrierte Übungen und Tests
- im PC-Pool
 - bediente Tutorien
 - Kleingruppen-Projektarbeit an konkreten physikalischen Problemen

Zielgruppe

Hauptsächlich Bachelor Studiengang Physik (1. - 5. Semester)

Auswahl aus zwei Angeboten:

- Ohne Programmier-Kenntnisse
 - Grundlagen von python
 - Anwendungen für das Physikalische Grundpraktikum
- Vorkenntnisse in python
 - Wissenschaftliche Datenverarbeitung mit python
 - Anwendungen zur Vorbereitung auf Computational Physics

Einschreibung ist für die Teilnahme erforderlich

Zeitraum: Täglich (zwei Wochen) von 21.3. bis 1.4. 2022

Teilnahme ist im Modul Allgemeine Qualifikationen (AQua) anrechenbar (hier Leistungspunkte).

Kurs beruht auf zentral bereit gestellten jupyter notebooks - keine Software-Installation für Teilnehmer notwendig (jedoch möglich)

Programmierkurs: Python in der Physik

10 Tage im März

Kurs

Programmierkurs: Python in der Physik 2022

Fakultät Physik | Wintersemester 2021 / 2022

Programmierkurs: Python in der Physik 2022

Verantwortlicher: Jens Brose

Block - Programmieren Python für Studierende der Physik

Blended Learning

- im Selbststudium (flipped classroom)
 - Aneignung der Grundlagen der Programmiersprache python
 - Anwendung zur Lösung physikalischer Probleme
 - Integrierte Übungen und Tests
- im PC-Pool
 - bediente Tutorien
 - Kleingruppen-Projektarbeit an konkreten physikalischen Problemen

Zielgruppe

Hauptsächlich Bachelor Studiengang Physik (1. - 5. Semester)

Auswahl aus zwei Angeboten:

- Ohne Programmier-Kenntnisse
 - Grundlagen von python
 - Anwendungen für das Physikalische Grundpraktikum
- Vorkenntnisse in python
 - Wissenschaftliche Datenverarbeitung mit python
 - Anwendungen zur Vorbereitung auf Computational Physics

Einschreibung ist für die Teilnahme erforderlich

Zeitraum: Täglich (zwei Wochen) von 21.3. bis 1.4. 2022

Teilnahme ist im Modul Allgemeine Qualifikationen (AQua) anrechenbar (hier Leistungspunkte).

Kurs beruht auf zentral bereit gestellten jupyter notebooks - keine Software-Installation für Teilnehmer notwendig (jedoch möglich)

Programmierkurs: Python in der Physik

Tägliche Selbst- lerneinheiten

Selbstlerneinheit

jupyter iterables_iterators_while_recursion Last Checkpoint: 13.07.2022 (autosaved) Logout Control Panel

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Wichtige Methoden auf Listen

Gegeben sei die Liste `a = [3, 2, 1]`.

Methode/Funktion	Operation	Beispiel	Ergebnis
<code>.append()</code>	An Listendeende anfügen	<code>a.append(0)</code>	<code>a = [3, 2, 1, 0]</code>
<code>.pop()</code>	Letztes Element entfernen	<code>a.pop()</code>	<code>1</code> ; <code>a = [3, 2]</code>
<code>.index()</code>	Index eines Elementes*	<code>a.index(2)</code> <code>a.index("bla")</code>	<code>1</code> <code>ValueError: 'bla' is not in list</code>
<code>min()</code>	Minimales Element**	<code>min(a)</code>	<code>1</code>
<code>max()</code>	Maximales Element**	<code>max(a)</code>	<code>3</code>
<code>.sort()</code>	Sortieren**	<code>a.sort()</code>	<code>a = [1, 2, 3]</code>
<code>len()</code>	Länge	<code>len(a)</code>	<code>3</code>
<code>[]</code>	Direkter Elementzugriff	<code>a[0]</code> <code>a[2] = 4</code>	<code>3</code> <code>a = [3, 2, 4]</code>

* Indizierung beginnt in Python bei 0

** nur für Listen, welche vergleichbare Daten enthalten.

Weitere Methoden finden sich in der [Dokumentation](#).

Der oben beschriebene direkte Elementzugriff kann auch in "Scheiben" (engl. *slices*) erfolgen. Die Syntax hierfür ist `start:end:step`. Das Element am Index `end` ist **nicht** Teil der Rückgabe. Die Standardwerte sind `0:len():1` und können auch frei gelassen werden.

Programmierkurs: Python in der Physik

Tägliche Selbst- lerneinheiten Basierend auf Jupyter Notebooks

Selbstlerneinheit

jupyter iterables_iterators_while_recursion Last Checkpoint: 13.07.2022 (autosaved) Logout Control Panel

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3

Wichtige Methoden auf Listen

Gegeben sei die Liste `a = [3, 2, 1]`.

Methode/Funktion	Operation	Beispiel	Ergebnis
<code>.append()</code>	An Listende anfügen	<code>a.append(0)</code>	<code>a = [3, 2, 1, 0]</code>
<code>.pop()</code>	Letztes Element entfernen	<code>a.pop()</code>	<code>1</code> ; <code>a = [3, 2]</code>
<code>.index()</code>	Index eines Elementes*	<code>a.index(2)</code> <code>a.index("bla")</code>	<code>1</code> <code>ValueError: 'bla' is not in list</code>
<code>min()</code>	Minimales Element**	<code>min(a)</code>	<code>1</code>
<code>max()</code>	Maximales Element**	<code>max(a)</code>	<code>3</code>
<code>.sort()</code>	Sortieren**	<code>a.sort()</code>	<code>a = [1, 2, 3]</code>
<code>len()</code>	Länge	<code>len(a)</code>	<code>3</code>
<code>[]</code>	Direkter Elementzugriff	<code>a[0]</code> <code>a[2] = 4</code>	<code>3</code> <code>a = [3, 2, 4]</code>

* Indizierung beginnt in Python bei 0
** nur für Listen, welche vergleichbare Daten enthalten.

Weitere Methoden finden sich in der [Dokumentation](#).

Der oben beschriebene direkte Elementzugriff kann auch in "Scheiben" (engl. *slices*) erfolgen. Die Syntax hierfür ist `start:end:step`. Das Element am Index `end` ist **nicht** Teil der Rückgabe. Die Standardwerte sind `0:len():1` und können auch frei gelassen werden.

Programmierkurs: Python in der Physik

Tägliche Selbst-
lerneinheiten
Basierend auf
Jupyter
Notebooks
Im Inverted
Classroom

Selbstlerneinheit

```
In [1]: a = [3, 4, 2, 0, 1, 5]
print(a[:5])      # jedes Element von a bis zur 1 (die 5 gehört nicht mehr dazu)
print(a[1:])      # a ab dem zweiten Element bis zum Ende
print(a[::2])     # jedes zweite Element von a

[3, 4, 2, 0, 1]
[4, 2, 0, 1, 5]
[3, 2, 1]
```

Hinweis: Beim wiederholten Aufruf von Zeilen haben die darin verwendeten Variablen den Wert ihrer letzten Verwendung (auch wenn sie weiter oben stehen als die zuletzt verwendete Zeile).

Sortieren Sie die obige Liste `a` in absteigender Reihenfolge und zeigen Sie die sortierte Liste an (größte Zahl zuerst):

```
In [2]: a.sort(reverse=True)
```

```
In [3]: !run -i tests/iterables_iterators_while_recursion/test1
```

Gut gemacht!

```
In [4]: print(a)
```

```
[5, 4, 3, 2, 1, 0]
```

Programmierkurs: Python in der Physik

Tägliche Selbst-
lerneinheiten

Basierend auf

Jupyter
Notebooks

Im Inverted
Classroom

Je ca. 90 Minuten
Aufwand

Selbstlerneinheit

```
In [1]: a = [3, 4, 2, 0, 1, 5]
print(a[:5])      # jedes Element von a bis zur 1 (die 5 gehört nicht mehr dazu)
print(a[1:])      # a ab dem zweiten Element bis zum Ende
print(a[::2])     # jedes zweite Element von a

[3, 4, 2, 0, 1]
[4, 2, 0, 1, 5]
[3, 2, 1]
```

Hinweis: Beim wiederholten Aufruf von Zellen haben die darin verwendeten Variablen den Wert ihrer letzten Verwendung (auch wenn sie weiter oben stehen als die zuletzt verwendete Zelle).

Sortieren Sie die obige Liste `a` in absteigender Reihenfolge und zeigen Sie die sortierte Liste an (größte Zahl zuerst):

```
In [2]: a.sort(reverse=True)
```

```
In [3]: %run -i tests/iterables_iterators_while_recursion/test1
```

Gut gemacht!

```
In [4]: print(a)
```

```
[5, 4, 3, 2, 1, 0]
```

Programmierkurs: Python in der Physik

Tägliche 90 min
Übung

Notebook

Übungsaufgaben zur Numerischen Nullstellensuche

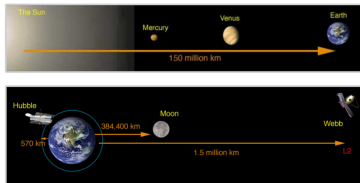
Die Position des James-Webb-Weltraumteleskops im System von Erde und Sonne

Das James-Webb-Weltraumteleskop (als Nachfolger des Hubble-Teleskops) wurde am 25. Dezember 2021 gestartet und erreichte am 24. Januar 2022 eine Umlaufbahn um den etwa 1,5 Millionen Kilometer von der Erde entfernten Lagrange-Punkt L2 im System von Sonne und Erde.



A Solar Orbit

The James Webb Space Telescope will not be in orbit around the Earth, like the Hubble Space Telescope is - it will actually orbit the Sun, 1.5 million kilometers (1 million miles) away from the Earth at what is called the second Lagrange point or L2. What is special about this orbit is that it lets the telescope stay in line with the Earth as it moves around the Sun. This allows the satellite's large sunshield to protect the telescope from the light and heat of the Sun and Earth (and Moon).



Webb will orbit the sun 1.5 million kilometers (1 million miles) away from the Earth at what is called the second Lagrange point or L2. (Note that these graphics are not to scale.)

Programmierkurs: Python in der Physik

Tägliche 90 min
Übung

Kleingruppen mit
max. 12 Personen

Notebook

Übungsaufgaben zur Numerischen Nullstellensuche

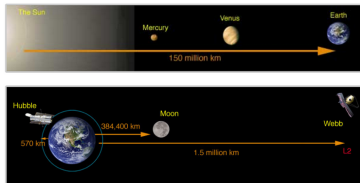
Die Position des James-Webb-Weltraumteleskops im System von Erde und Sonne

Das James-Webb-Weltraumteleskop (als Nachfolger des Hubble-Teleskops) wurde am 25. Dezember 2021 gestartet und erreichte am 24. Januar 2022 eine Umlaufbahn um den etwa 1,5 Millionen Kilometer von der Erde entfernten Lagrange-Punkt L2 im System von Sonne und Erde.



A Solar Orbit

The James Webb Space Telescope will not be in orbit around the Earth, like the Hubble Space Telescope is - it will actually orbit the Sun, 1.5 million kilometers (1 million miles) away from the Earth at what is called the second Lagrange point or L2. What is special about this orbit is that it lets the telescope stay in line with the Earth as it moves around the Sun. This allows the satellite's large sunshield to protect the telescope from the light and heat of the Sun and Earth (and Moon).



Webb will orbit the sun 1.5 million kilometers (1 million miles) away from the Earth at what is called the second Lagrange point or L2. (Note that these graphics are not to scale.)

Programmierkurs: Python in der Physik

Tägliche 90 min
Übung

Kleingruppen mit
max. 12 Personen
mit Tutor im
PC-Pool

Notebook

Übungsaufgaben zur Numerischen Nullstellensuche

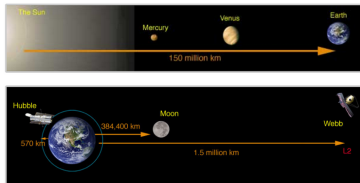
Die Position des James-Webb-Weltraumteleskops im System von Erde und Sonne

Das James-Webb-Weltraumteleskop (als Nachfolger des Hubble-Teleskops) wurde am 25. Dezember 2021 gestartet und erreichte am 24. Januar 2022 eine Umlaufbahn um den etwa 1,5 Millionen Kilometer von der Erde entfernten Lagrange-Punkt L₂ im System von Sonne und Erde.



A Solar Orbit

The James Webb Space Telescope will not be in orbit around the Earth, like the Hubble Space Telescope is - it will actually orbit the Sun, 1.5 million kilometers (1 million miles) away from the Earth at what is called the second Lagrange point or L₂. What is special about this orbit is that it lets the telescope stay in line with the Earth as it moves around the Sun. This allows the satellite's large sunshield to protect the telescope from the light and heat of the Sun and Earth (and Moon).

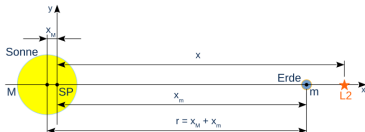


Webb will orbit the sun 1.5 million kilometers (1 million miles) away from the Earth at what is called the second Lagrange point or L₂. (Note that these graphics are not to scale.)

Programmierkurs: Python in der Physik

Notebook

Das allgemeine Dreikörperproblem der Himmelsmechanik ist nicht analytisch lösbar. Wenn der dritte Körper eine – verglichen mit den beiden anderen – vernachlässigbare Masse hat, fanden Leonhard Euler und Joseph-Louis Lagrange fünf analytische Lösungen: In den nach [Lagrange 1.1 bis 1.5 genannten Punkten](#) können dritte Körper (z. B. Satelliten) kräftefrei ruhen. In diesen Punkten kompensieren sich die Gravitationskräfte, die die beiden großen Himmelskörper auf die kleine Masse ausüben und die auf die kleine Masse wirkende Zentrifugalkraft in einem rotierenden Bezugssystem, in dem auch die beiden schweren Himmelskörper ruhen.



Wir betrachten hier das System Sonne - Erde in einem mitrotierenden Koordinatensystem, dessen Ursprung im gemeinsamen Schwerpunkt SP von Sonne und Erde liegt (innerhalb des Sonnenradius). Der äußere Lagrange-Punkt L2 befindet sich auf der Verlängerung der Verbindungslinie zwischen den beiden betrachteten Körpern. In diesem Koordinatensystem sind Sonne, Erde, und L2 ortsfest.

Der mittlere Abstand Sonne - Erde beträgt $r = 1,496 \cdot 10^8$ km, die Sonnenmasse $M = 1,988 \cdot 10^{30}$ kg und die Erdmasse $m = 5,972 \cdot 10^{24}$ kg. Aus dem Hebelgesetz ergeben sich die Abstände

$$x_M = \frac{m}{M+m}r \text{ und } x_m = \frac{M}{M+m}r \quad (1)$$

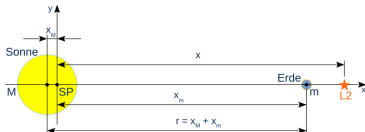
des Sonnen- bzw. Erdmittelpunkts vom Ursprung (gemeinsamer Schwerpunkt SP). Die Umlaufdauer von Sonne und Erde um den gemeinsamen Schwerpunkt SP beträgt $T = 365,26$ Tage (siderisches Jahr).

Basierend auf
Jupyter
Notebooks

Programmierkurs: Python in der Physik

Notebook

Das allgemeine Dreikörperproblem der Himmelsmechanik ist nicht analytisch lösbar. Wenn der dritte Körper eine – verglichen mit den beiden anderen – vernachlässigbare Masse hat, fanden Leonhard Euler und Joseph-Louis Lagrange fünf analytische Lösungen: In den nach [Lagrange 1.1 bis 1.5 genannten Punkten](#) können dritte Körper (z. B. Satelliten) kräftefrei ruhen. In diesen Punkten kompensieren sich die Gravitationskräfte, die die beiden großen Himmelskörper auf die kleine Masse ausüben und die auf die kleine Masse wirkende Zentrifugalkraft in einem rotierenden Bezugssystem, in dem auch die beiden schweren Himmelskörper ruhen.



Wir betrachten hier das System Sonne - Erde in einem mitrotierenden Koordinatensystem, dessen Ursprung im gemeinsamen Schwerpunkt SP von Sonne und Erde liegt (innerhalb des Sonnenradius). Der äußere Lagrange-Punkt L2 befindet sich auf der Verlängerung der Verbindungslinie zwischen den beiden betrachteten Körpern. In diesem Koordinatensystem sind Sonne, Erde, und L2 ortsfest.

Der mittlere Abstand Sonne - Erde beträgt $r = 1,496 \cdot 10^8$ km, die Sonnenmasse $M = 1,988 \cdot 10^{30}$ kg und die Erdmasse $m = 5,972 \cdot 10^{24}$ kg. Aus dem Hebelgesetz ergeben sich die Abstände

$$x_M = \frac{m}{M+m}r \text{ und } x_m = \frac{M}{M+m}r \quad (1)$$

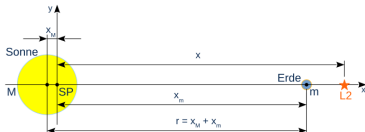
des Sonnen- bzw. Erdmittelpunkts vom Ursprung (gemeinsamer Schwerpunkt SP). Die Umlaufdauer von Sonne und Erde um den gemeinsamen Schwerpunkt SP beträgt $T = 365,26$ Tage (siderisches Jahr).

Basierend auf
Jupyter
Notebooks
Aufgabenstellung

Programmierkurs: Python in der Physik

Notebook

Das allgemeine Dreikörperproblem der Himmelsmechanik ist nicht analytisch lösbar. Wenn der dritte Körper eine – verglichen mit den beiden anderen – vernachlässigbare Masse hat, fanden Leonhard Euler und Joseph-Louis Lagrange fünf analytische Lösungen: In den nach [Lagrange 1.1 bis 1.5 genannten Punkten](#) können dritte Körper (z. B. Satelliten) kräftefrei ruhen. In diesen Punkten kompensieren sich die Gravitationskräfte, die die beiden großen Himmelskörper auf die kleine Masse ausüben und die auf die kleine Masse wirkende Zentrifugalkraft in einem rotierenden Bezugssystem, in dem auch die beiden schweren Himmelskörper ruhen.



Wir betrachten hier das System Sonne - Erde in einem mitrotierenden Koordinatensystem, dessen Ursprung im gemeinsamen Schwerpunkt SP von Sonne und Erde liegt (innerhalb des Sonnenradius). Der äußere Lagrange-Punkt L2 befindet sich auf der Verlängerung der Verbindungslinie zwischen den beiden betrachteten Körpern. In diesem Koordinatensystem sind Sonne, Erde, und L2 ortsfest.

Der mittlere Abstand Sonne - Erde beträgt $r = 1,496 \cdot 10^8$ km, die Sonnenmasse $M = 1,988 \cdot 10^{30}$ kg und die Erdmasse $m = 5,972 \cdot 10^{24}$ kg. Aus dem Hebelgesetz ergeben sich die Abstände

$$x_M = \frac{m}{M+m}r \text{ und } x_m = \frac{M}{M+m}r \quad (1)$$

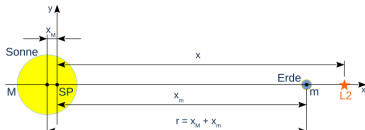
des Sonnen- bzw. Erdmittelpunkts vom Ursprung (gemeinsamer Schwerpunkt SP). Die Umlaufdauer von Sonne und Erde um den gemeinsamen Schwerpunkt SP beträgt $T = 365,26$ Tage (siderisches Jahr).

Basierend auf
Jupyter
Notebooks
Aufgabenstellung
Erläuterungen

Programmierkurs: Python in der Physik

Notebook

Das allgemeine Dreikörperproblem der Himmelsmechanik ist nicht analytisch lösbar. Wenn der dritte Körper eine – verglichen mit den beiden anderen – vernachlässigbare Masse hat, fanden Leonhard Euler und Joseph-Louis Lagrange fünf analytische Lösungen: In den nach [Lagrange 1.1 bis 1.5 genannten Punkten](#) können dritte Körper (z. B. Satelliten) kräftefrei ruhen. In diesen Punkten kompensieren sich die Gravitationskräfte, die die beiden großen Himmelskörper auf die kleine Masse ausüben und die auf die kleine Masse wirkende Zentrifugalkraft in einem rotierenden Bezugssystem, in dem auch die beiden schweren Himmelskörper ruhen.



Wir betrachten hier das System Sonne - Erde in einem mitrotierenden Koordinatensystem, dessen Ursprung im gemeinsamen Schwerpunkt SP von Sonne und Erde liegt (innerhalb des Sonnenradius). Der äußere Lagrange-Punkt L2 befindet sich auf der Verlängerung der Verbindungslinie zwischen den beiden betrachteten Körpern. In diesem Koordinatensystem sind Sonne, Erde, und L2 ortsfest.

Der mittlere Abstand Sonne - Erde beträgt $r = 1,496 \cdot 10^8$ km, die Sonnenmasse $M = 1,988 \cdot 10^{30}$ kg und die Erdmasse $m = 5,972 \cdot 10^{24}$ kg. Aus dem Hebelgesetz ergeben sich die Abstände

$$x_M = \frac{m}{M+m}r \text{ und } x_m = \frac{M}{M+m}r \quad (1)$$

des Sonnen- bzw. Erdmittelpunkts vom Ursprung (gemeinsamer Schwerpunkt SP). Die Umlaufdauer von Sonne und Erde um den gemeinsamen Schwerpunkt SP beträgt $T = 365,26$ Tage (siderisches Jahr).

Basierend auf
Jupyter
Notebooks
Aufgabenstellung
Erläuterungen
Code-Eingaben

Programmierungskurs: Python in der Physik

- OPAL-LTI $\implies$ GitLab $\implies$ Jupyterhub

Programmierkurs: Python in der Physik

- OPAL-LTI $\implies$ GitLab $\implies$ Jupyterhub
- Initial gleiche, individuell bearbeit- und speicherbare Notebooks

Programmierungskurs: Python in der Physik

- OPAL-LTI $\implies$ GitLab $\implies$ Jupyterhub
- Initial gleiche, individuell bearbeit- und speicherbare Notebooks
- Webanwendung unabhängig vom Arbeitsgerät

Programmierkurs: Python in der Physik

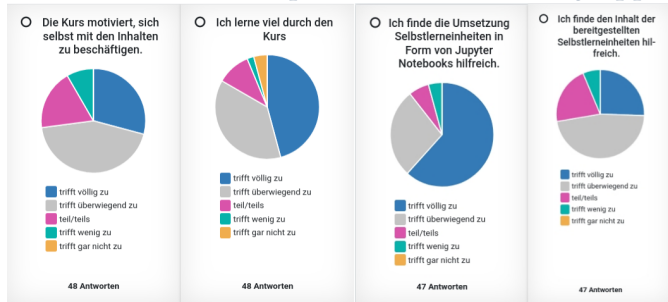
- OPAL-LTI $\implies$ GitLab $\implies$ Jupyterhub
- Initial gleiche, individuell bearbeit- und speicherbare Notebooks
- Webanwendung unabhängig vom Arbeitsgerät
- Inverted Classroom: Zwei unabhängige Kohorten
(Programmier-Anfänger und -Fortgeschrittene)

Programmierkurs: Python in der Physik

- OPAL-LTI $\implies$ GitLab $\implies$ Jupyterhub
- Initial gleiche, individuell bearbeit- und speicherbare Notebooks
- Webanwendung unabhängig vom Arbeitsgerät
- Inverted Classroom: Zwei unabhängige Kohorten
(Programmier-Anfänger und -Fortgeschrittene)
- 68 Teilnehmende von potentiell 100 der Zielgruppe (2022)

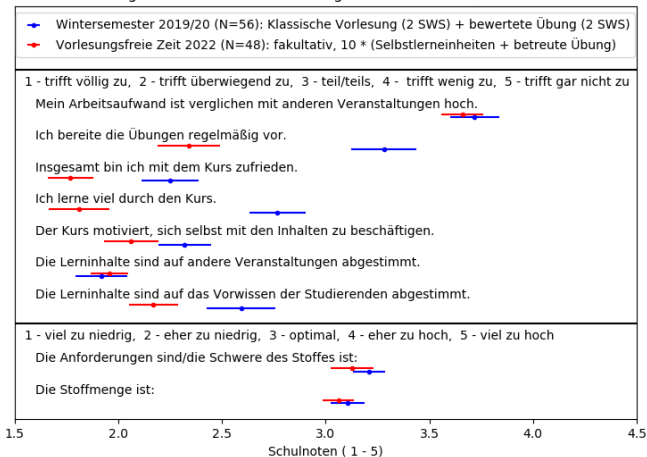
Programmierkurs: Python in der Physik

- OPAL-LTI $\Rightarrow$ GitLab $\Rightarrow$ Jupyterhub
- Initial gleiche, individuell bearbeit- und speicherbare Notebooks
- Webanwendung unabhängig vom Arbeitsgerät
- Inverted Classroom: Zwei unabhängige Kohorten (Programmier-Anfänger und -Fortgeschrittene)
- 68 Teilnehmende von potentiell 100 der Zielgruppe (2022)



Programmierkurs: Python in der Physik

Vergleich der Lehrveranstaltungsevaluation 2019/20 - 2022



Ausblick: Interaktive Jupyternotebooks im Physikstudium

Interaktive Jupyter Notebooks im
Physikstudium



TU Dresden | semesterübergreifend

Interaktive Jupyter Notebooks im Physikstudium

Verantwortliche: [Jens Brose](#)

Eine Bibliothek interaktiver Anwendungen zur Verdeutlichung physikalischer Zusammenhänge

Einbindung in OPAL - Kurse

- zur Demonstration in Vorlesungen / Seminaren / Übungen
- zum Selbststudium für Studierende

gefördert durch den Fonds digitales Lernen und Lehren der TU Dresden 2022/2023

[Weitere Informationen anzeigen](#)

Experimentalphysik 1

Mechanik und Thermodynamik

☐ Gedämpfte Schwingungen

Schwing., aperiodischer Grenz- und Kriechfall



☐ Carnot Prozess

Interaktiver Carnot Kreisprozess



Statistik

Wahrscheinlichkeitsverteilungen

☐ Binomialverteilung



☐ Normalverteilung



Ausblick: Interaktive Jupyternotebooks im Physikstudium

