

Jupyter Notebook Tutorials im Browser

Inhalt

1. Was sind Jupyter Notebooks?	1
2. Aufruf der Jupyter Notebooks	1
3. Nutzung der Jupyter Notebooks	2
4. Weiterführende Informationen	4

1. Was sind Jupyter Notebooks?

Jupyter Notebooks ermöglichen das Verfassen reproduzierbarer Analysen, anschaulicher Beispiele und interaktiver Übungen, indem Text mit ausführbarem Code kombiniert dargestellt wird. Die im Webbrowser aufrufbaren Jupyter Notebook Tutorials der Deutschen Nationalbibliothek zu exemplarischen Schnittstellenabfragen und Datenanalysen können auch ohne Programmierkenntnisse ausgeführt werden.

2. Aufruf der Jupyter Notebooks

Die Jupyter Notebook Tutorials der Deutschen Nationalbibliothek können über das [GitHub Repository](https://notebooks.gesis.org/binder/) geräteunabhängig in der webbasierten Entwicklungsumgebung <https://notebooks.gesis.org/binder/> aufgerufen werden.

Für das Arbeiten sollte beachtet werden, dass das Speicherlimit für nicht registrierte Nutzer bei 4 GB liegt und nach 10 Minuten Inaktivität ein Timeout stattfindet. Weitere Informationen zu Timeout und Speicherlimits werden unter <https://notebooks.gesis.org/faq/> aufgeführt.

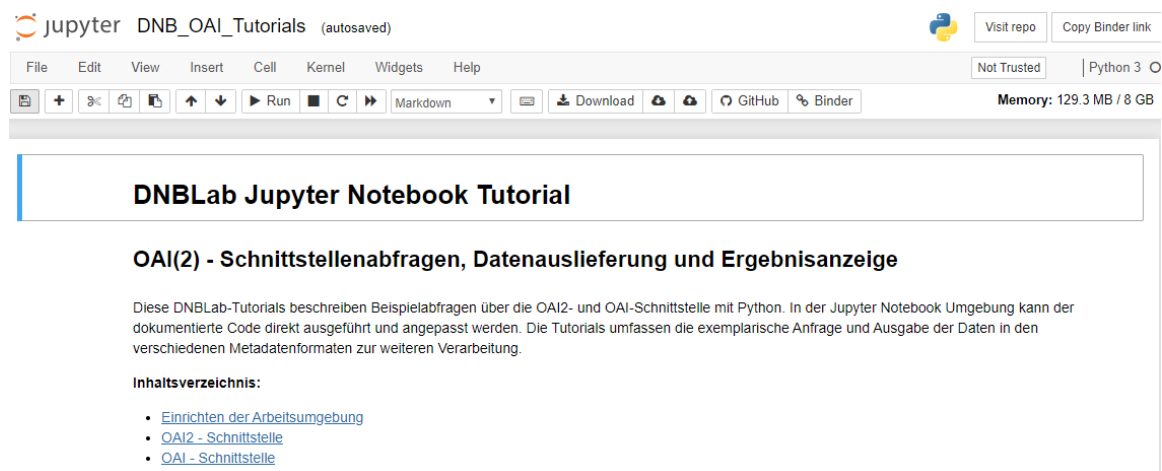
Schritt 1: Öffnen des Verzeichnisses mit allen Jupyter Notebook Tutorials über <https://notebooks.gesis.org/binder/v2/gh/deutsche-nationalbibliothek/dnblab/HEAD>



The screenshot shows the Jupyter Notebook interface. At the top, there is a header with the Jupyter logo and buttons for 'Visit repo', 'Copy Binder link', and 'Quit'. Below this, there are tabs for 'Files', 'Running', and 'Clusters'. A message says 'Select items to perform actions on them.' with buttons for 'Upload', 'New', and a refresh icon. The main area displays a list of files in a table:

	Name	Last Modified	File size
<input type="checkbox"/>	/		
<input type="checkbox"/>	DNB_OAI_Tutorials.ipynb	vor 3 Minuten	1.42 MB
<input type="checkbox"/>	DNB_SRU_Tutorial.ipynb	vor 3 Minuten	9.21 kB
<input type="checkbox"/>	DNB_Titelanalyse_MARC_Tutorial_Teil1.ipynb	vor 3 Minuten	16.2 kB
<input type="checkbox"/>	DNB_Titelanalyse_MARC_Tutorial_Teil2.ipynb	vor 3 Minuten	13.7 kB

Schritt 2: Auswahl eines Jupyter Notebook Tutorials mit Dateiendung **ipynb**.



Jupyter DNB_OAI_Tutorials (autosaved) Visit repo Copy Binder link

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 Memory: 129.3 MB / 8 GB




DNBLab Jupyter Notebook Tutorial

OAI(2) - Schnittstellenabfragen, Datenauslieferung und Ergebnisanzeige

Diese DNBLab-Tutorials beschreiben Beispielabfragen über die OAI2- und OAI-Schnittstelle mit Python. In der Jupyter Notebook Umgebung kann der dokumentierte Code direkt ausgeführt und angepasst werden. Die Tutorials umfassen die exemplarische Anfrage und Ausgabe der Daten in den verschiedenen Metadatenformaten zur weiteren Verarbeitung.

Inhaltsverzeichnis:

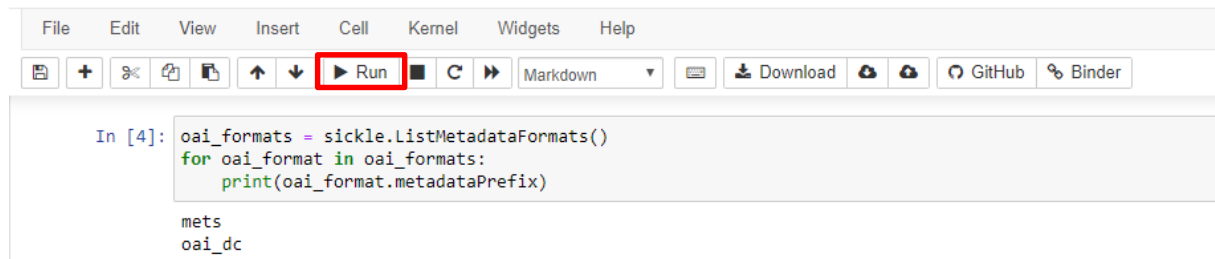
- [Einrichten der Arbeitsumgebung](#)
- [OAI2 - Schnittstelle](#)
- [OAI - Schnittstelle](#)

Tipp: Mittels Klick auf den Button  Download können Notebooks in den standardmäßig hinterlegten Downloadordner des Browsers heruntergeladen werden. Zum Schnellspeichern im Browserspeicher klicken Sie auf  . Zum Aufruf eines gespeicherten Notebooks verwenden Sie den Button  .


3. Nutzung der Jupyter Notebooks

Alle Jupyter Notebook Tutorials sind ähnlich strukturiert. Nach einer kurzen Beschreibung folgt der vorher beschriebene Code in der Programmiersprache Python und kann direkt ausgeführt werden.

Schritt 1: Die Codeteile werden über Mauscursor im Codebereich und gleichzeitigem Klick des Buttons RUN ausgeführt. Der Code muss in der vorgegebenen Reihenfolge ausgeführt werden, da die Codeteile aufeinander aufbauen.



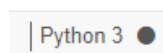
File Edit View Insert Cell Kernel Widgets Help

 Run

```
In [4]: oai_formats = sickle.ListMetadataFormats()
for oai_format in oai_formats:
    print(oai_format.metadataPrefix)

mets
oai_dc
```

Schritt 2: Das Ausführen kann einen Moment dauern. Im oberen rechten Bereich befindet sich neben „Python“ ein leerer Kreis, der den Zustand des Notebooks wiedergibt. Ist dieser Kreis grau gefüllt, so wird gerade Code ausgeführt. Während der Ausführung wird außerdem vor dem Codefeld **In [*]:** angezeigt, welches nach erfolgreicher Fertigstellung mit einer Zahl (Ausführungsschritt) gefüllt wird.



Python 3 ●

Schritt 3: Die Ergebnisse werden direkt unterhalb des jeweiligen Codefelds angezeigt.

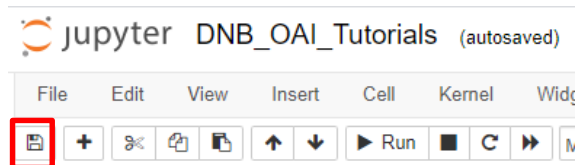
```
In [9]: oai_sets = sickle.ListSets()
for oai_set in oai_sets:
    print('setSpec value for selective harvesting: ' + oai_set.setSpec)
    print('Name of the set (setName): ' + oai_set.setName + '\n')

setSpec value for selective harvesting: dnb
Name of the set (setName): Deutsche Nationalbibliografie (DNB)

setSpec value for selective harvesting: dnb:sg0
Name of the set (setName): Deutsche Nationalbibliografie (DNB) - Allgemeines, Informationswissenschaft

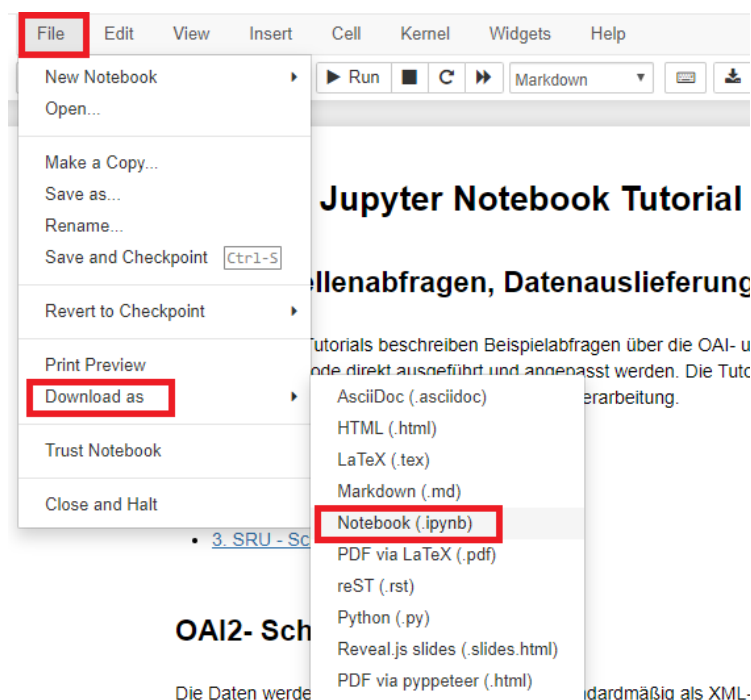
setSpec value for selective harvesting: dnb:sg000
Name of the set (setName): Deutsche Nationalbibliografie (DNB) - Allgemeines, Wissenschaft
```

Schritt 4: Änderungen in dem geöffneten Jupyter Notebook Tutorial können in der aktuellen Sitzung zwischengespeichert werden.



Schritt 5: Zur Sicherung der eigenen Anpassungen empfiehlt sich der Download des individuell angepassten Jupyter Notebooks. Die Notebooks werden im standardmäßig hinterlegten Downloadordner des Browsers abgelegt.

Tipp: Im Lesesaal finden Sie die Downloaddateien auf dem Desktop im Ordner „Daten Temporär“. Diese können beispielsweise per Mailanhang an die eigene Mail-Adresse versendet werden. Alternativ empfiehlt sich die Nutzung eines eigenen GitHub Repositoriums.



4. Weiterführende Informationen

Auf der DNBLab-Seite (<https://www.dnb.de/dnblab>) finden Sie weitere Informationen zu:

- frei verfügbaren Datensets
- Dokumentationen der verschiedenen Schnittstellenfunktionen
- Beschreibungen der angebotenen Metadatenformate
- weiteren Tutorials zu Datenanalysen und Abrufen von Schnittstellen