jupytercon

Creating Accessible Jupyter Notebooks: UC Berkeley's JupyterLab-a11y-checker

UC Berkeley

Allison Czapracki
Balaji Alwar
Chanbin Park
Ryan Lovett



#JupyterCon

Speakers

Allison Czapracki, Manager of the Learning Environments and Tools (LET) team in Research, Teaching and Learning at UC Berkeley



Balaji Alwar, Service Lead for DataHub and a member of the LET team



Chanbin Park, UC Berkeley, Senior studying Computer Science; Intern, LET



Ryan Lovett, Director of Computing, Statistical Computing Facility, and Jupyter Distinguished Contributor



UC Berkeley by the Numbers

#1

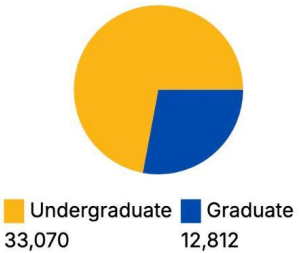
Public university worldwide: U.S. News & World Report

~400

Degree programs at UC Berkeley

26%

First generation first-year students



Total student enrollment, Fall 2024

19.4 to 1

Student-to-faculty ratio

21%

Transfer students

DataHub

- UC Berkeley's JupyterHub is known as DataHub
- DataHub piloted for the first time as part of Data8 course at UC Berkeley in 2017
- Joint initiative between Research, Teaching and Learning (RTL) and Computing, Data Science and Society (CDSS)
- Core part of 40+ courses every semester, serving approximately 14,000 students
- Many Data Science and Statistics courses use DataHub regularly for their lab, homework, and assignments

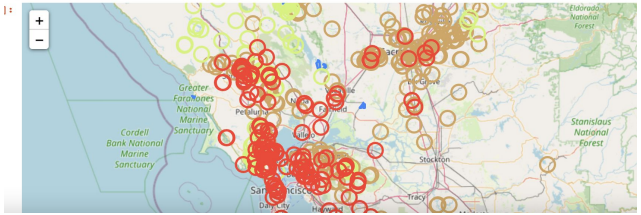


DataHub: Diverse Use Cases

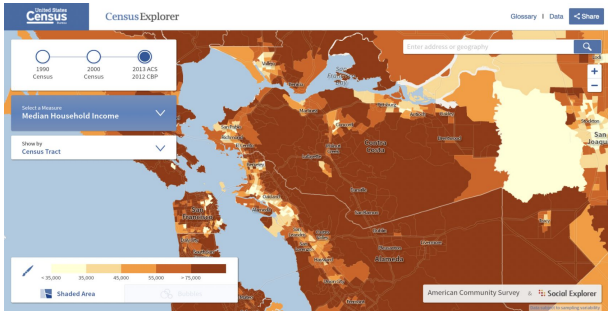
species	HEX	Color
Quercus agrifolia	#FD372E	
Quercus durata	#D4F273	
Quercus lobata	#D5A45E	

```
mapc = folium.Map([37.359276, -122.179626], zoom_start=5)

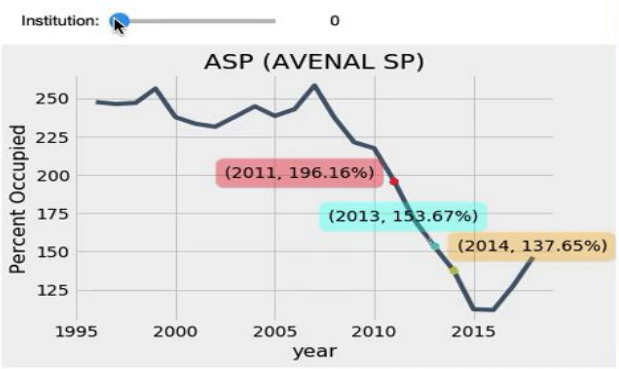
points = folium.features.GeoJson(reserves)
mapc.add_child(points)
for r in records_df.iterrows():
    lat = r[1]['decimalLatitude']
    long = r[1]['decimalLongitude']
    folium.CircleMarker((lat, long), color=color_dict[r[1]['species']]).add_to(mapc)
mapc
```



Notebook based pedagogy in a Environmental Science and Policy Management Course



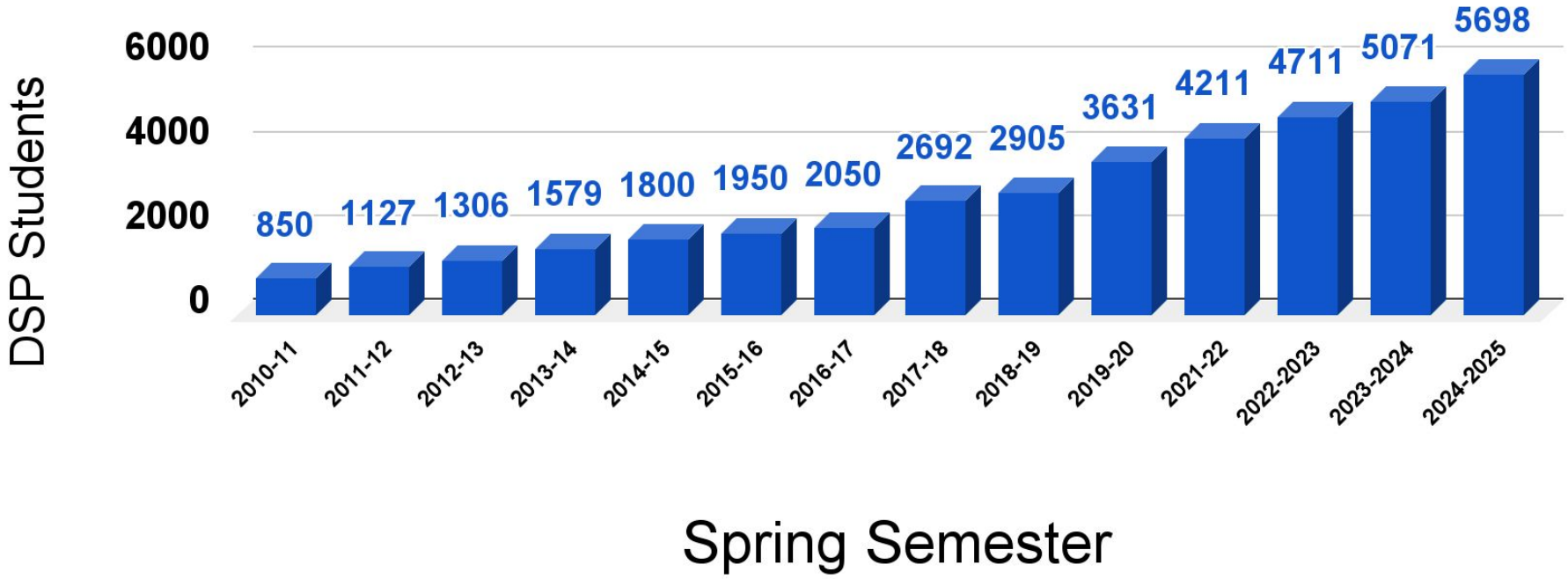
Notebook based pedagogy in a Civil Engineering course



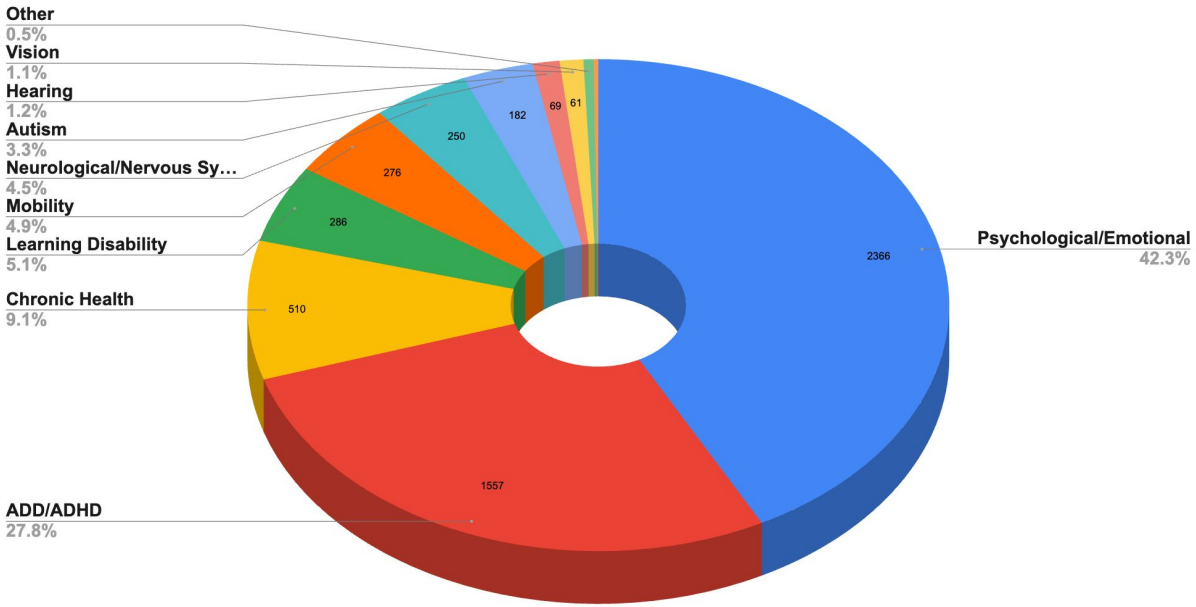
Notebook based pedagogy in a Ethnic Studies Course

Letter of Accommodations (LOAs)

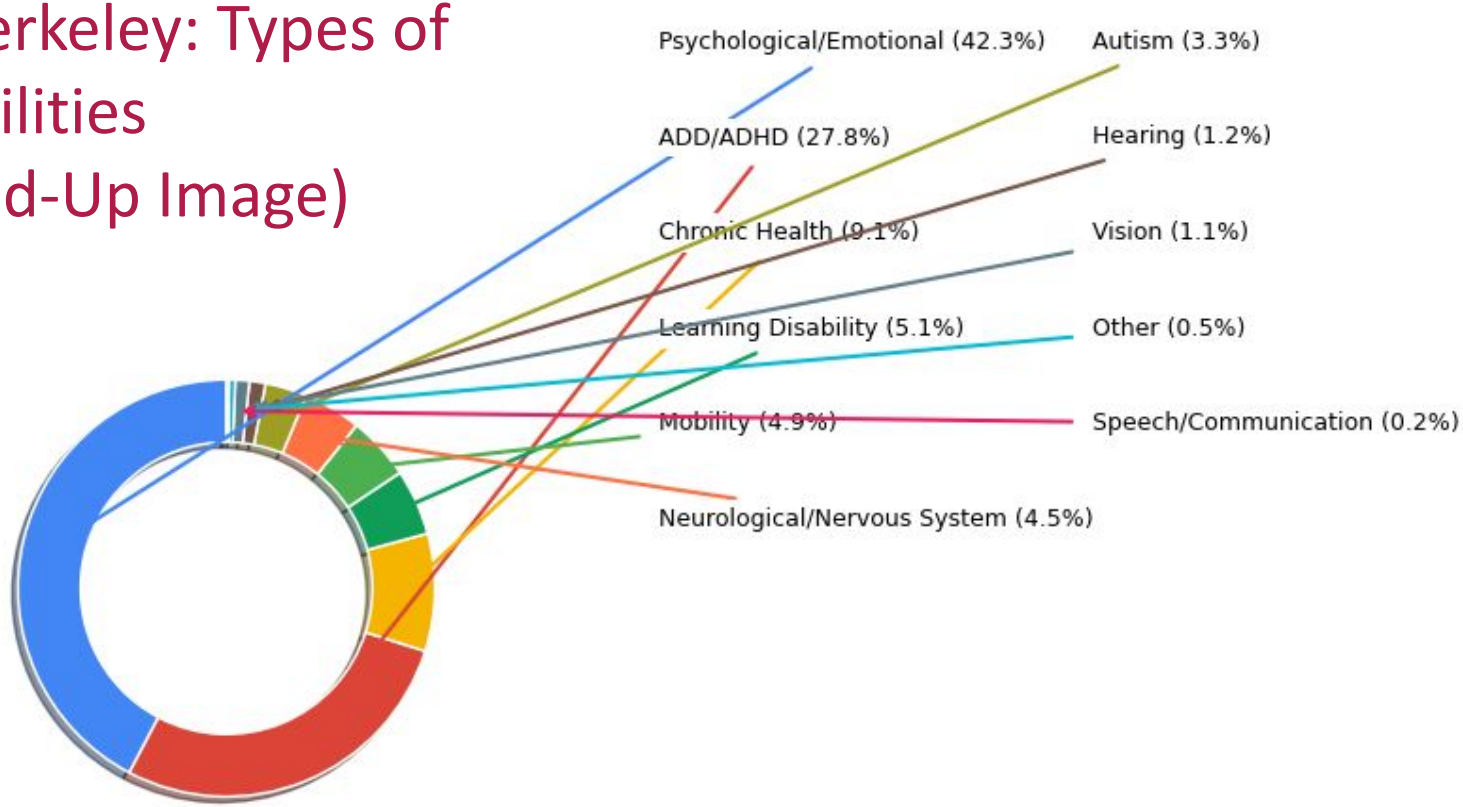
UC Berkeley



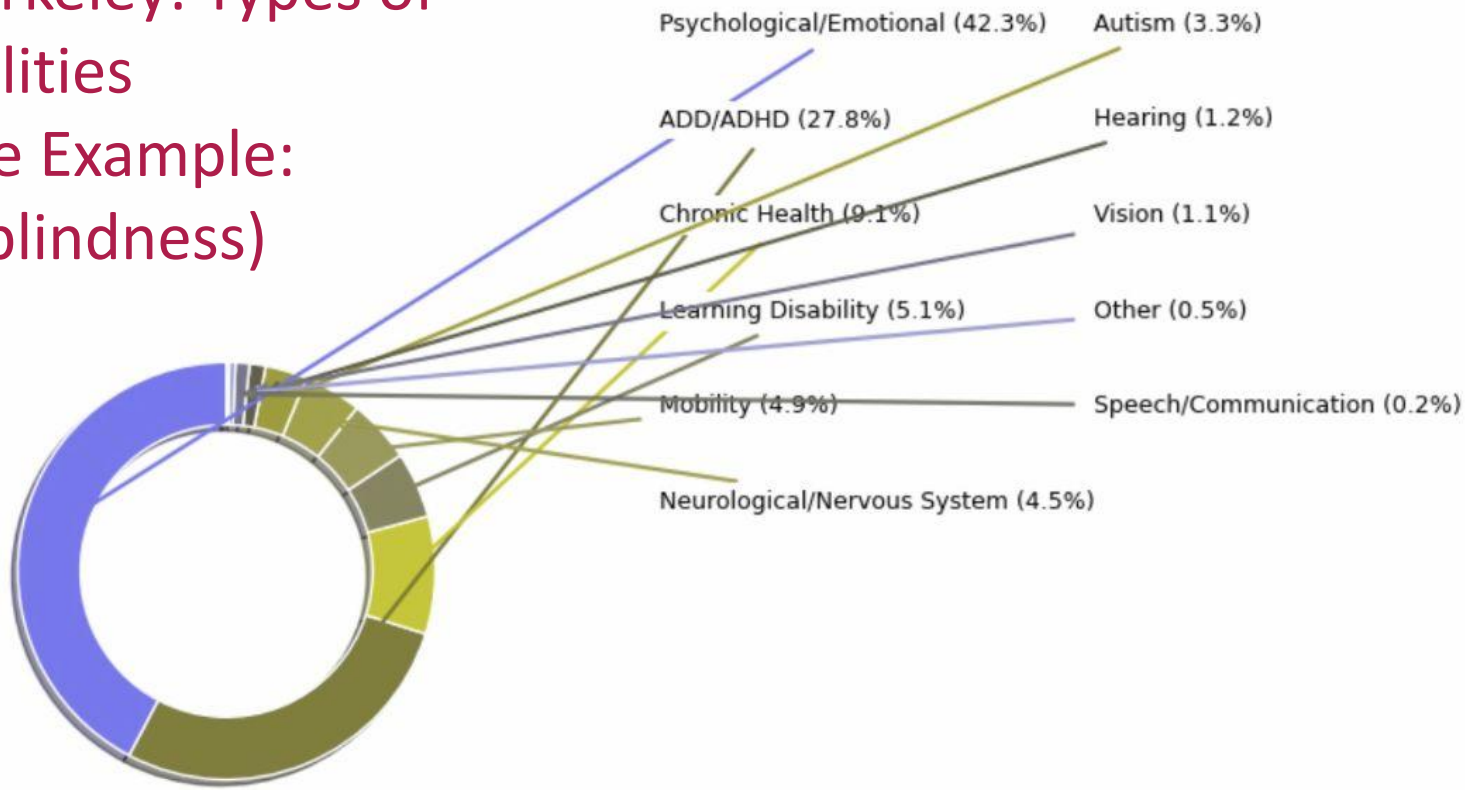
Disabilities Among UC Berkeley Students (Fall 2025)



UC Berkeley: Types of Disabilities (Mixed-Up Image)



UC Berkeley: Types of Disabilities (Image Example: Colorblindness)



WCAG Guidelines

- Stands for “Web Content Accessibility Guidelines”
- Set of guidelines published by World Wide Web Consortium (W3C) to make web content more accessible, primarily for people with disabilities
- WCAG 2.0 and 2.1 are commonly referenced standards for web accessibility



WCAG POUR Principles

Principle	Explanation	Example
Perceivable	Web content should be presented in a way that it can be perceived by users, including those with visual and auditory impairments.	Each image includes alternative text
Operable	Users should be able to navigate and interact with web content using various input methods.	Each notebook supports keyboard-only navigation
Understandable	Web content should be presented in a clear and understandable manner with focus on readability and comprehension.	Each notebook uses headings, subheadings, and bullet points to organize content into digestible chunks
Robust	Web content must be robust enough to be interpreted reliably by a wide range of technologies, including assistive technologies.	Each notebook is compatible with screen readers for reliable navigation

UC Berkeley's WCAG Requirements & Title II

- Accessibility of the Jupyter tools has been a key focus area at UC Berkeley
- All academic courses within the University of California system are required to align with WCAG 2.1 AA guidelines
- Based on the Department of Justice guidelines, all academic content (including Jupyter notebooks) is expected to be fully accessible by April 2026



Common Accessibility Issues in Jupyter Notebooks

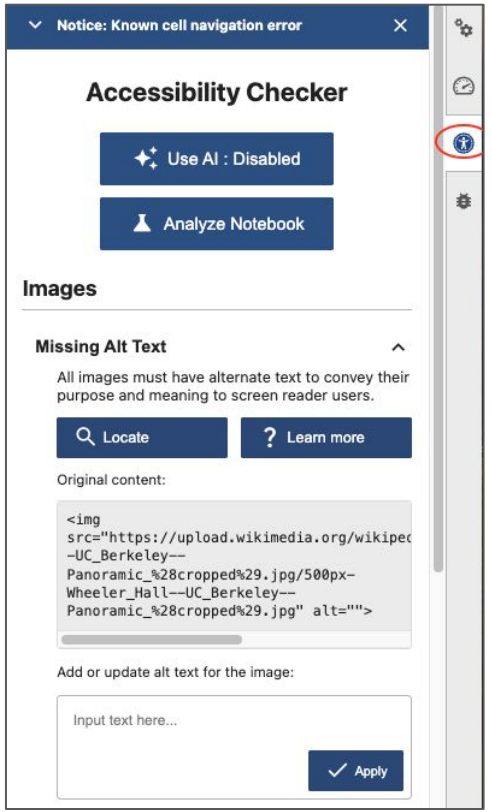
- Potluri et. al. analyzed 100,000+ Jupyter notebooks and reported common accessibility issues
- Accessibility issues reported were specific to the authoring practices of the notebook creators and were outside the scope of work of the core Jupyter team
- Our extension is designed to help notebook authors release accessible notebooks!



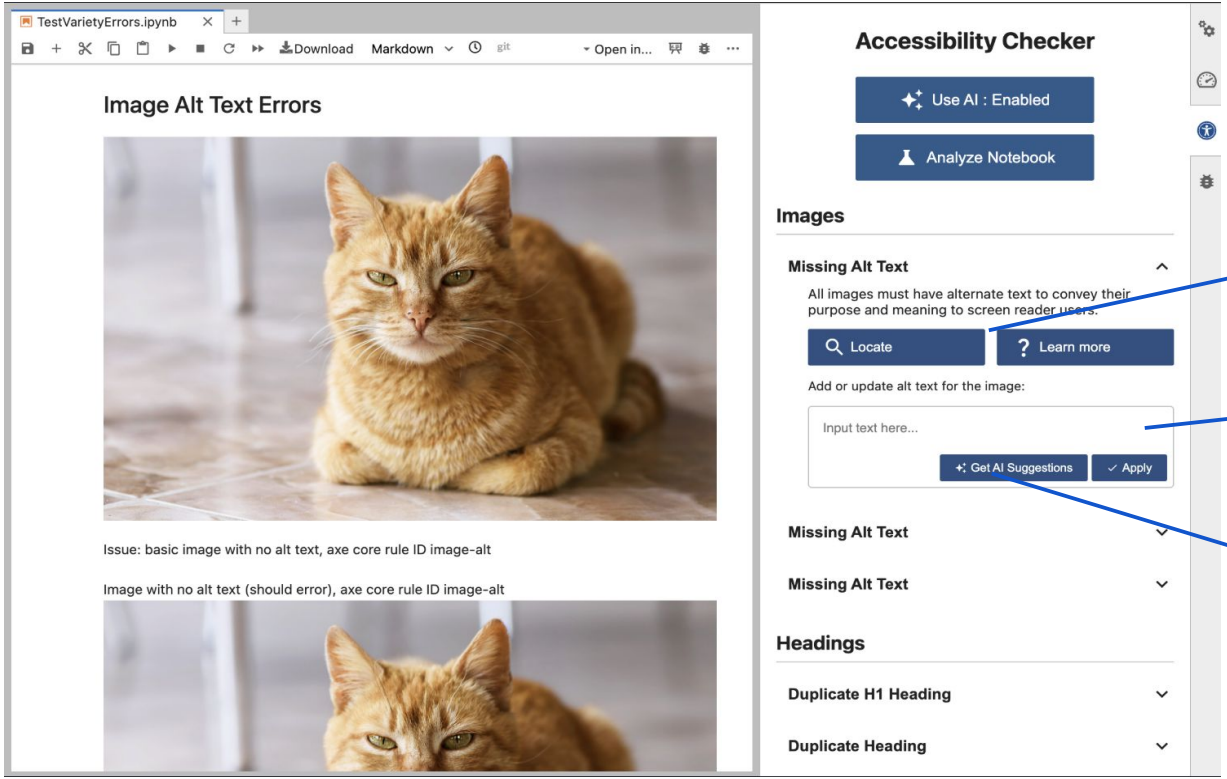
Study: Notably Inaccessible — Data Driven Understanding of Data Science Notebook (In)Accessibility

JupyterLab-a11y-checker Extension

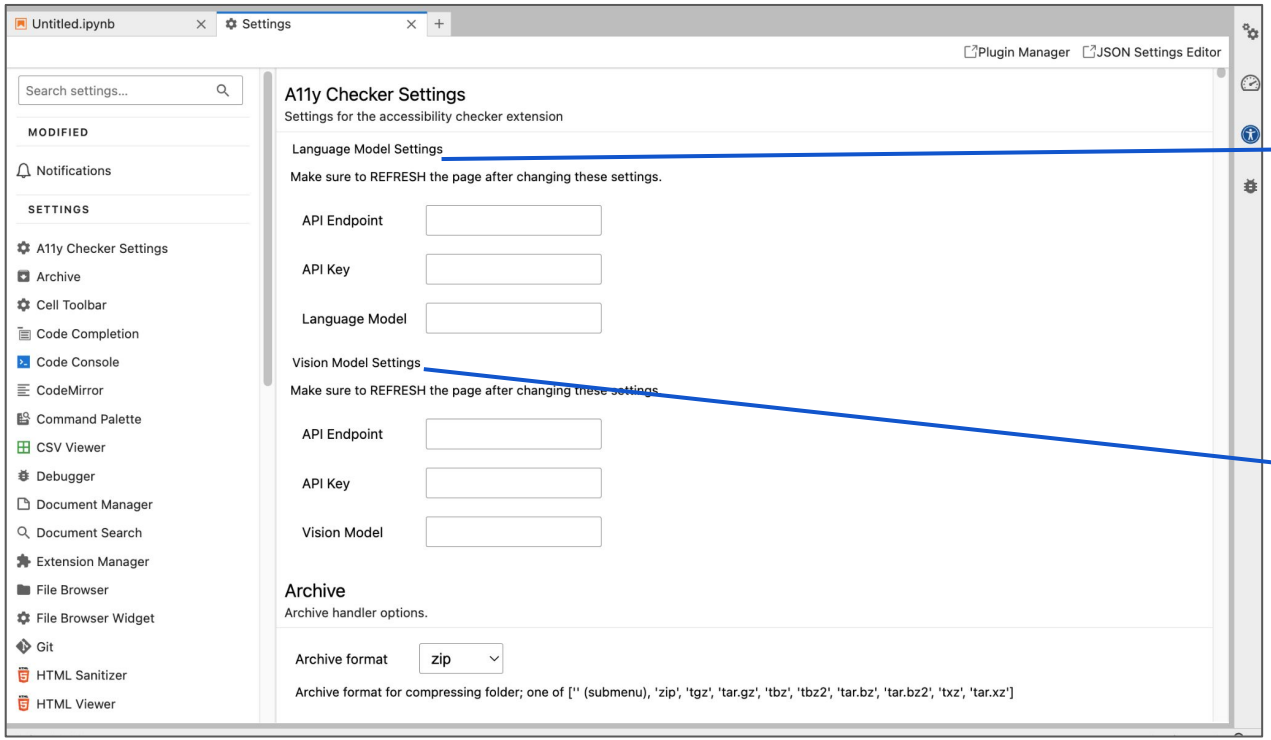
- UC Berkeley student developer team built a11y-checker extension to help notebook authors identify and fix common a11y issues during notebook development
- Extension helps notebook authors do the following:
  - Quickly identify common a11y issues
  - Review actionable insights about the accessibility issues
  - Fix a11y issues in minimal clicks



JupyterLab-a11y-checker Features



Usage of AI models



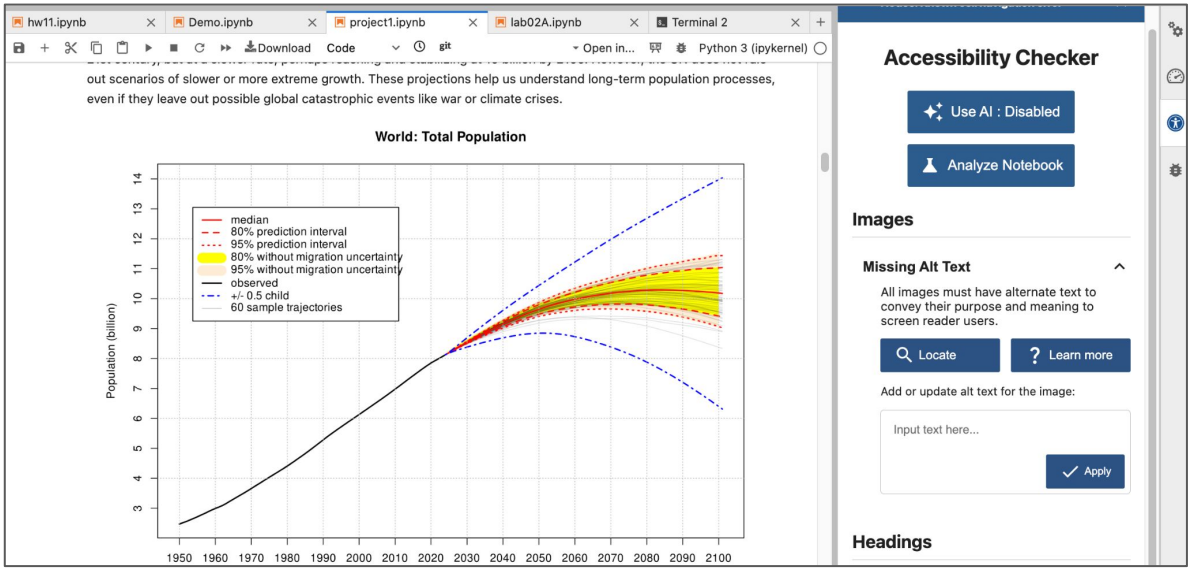
Input language model information to generate table captions (summary)

Input vision model information to generate alternative text for images

WCAG Guidelines Currently Implemented

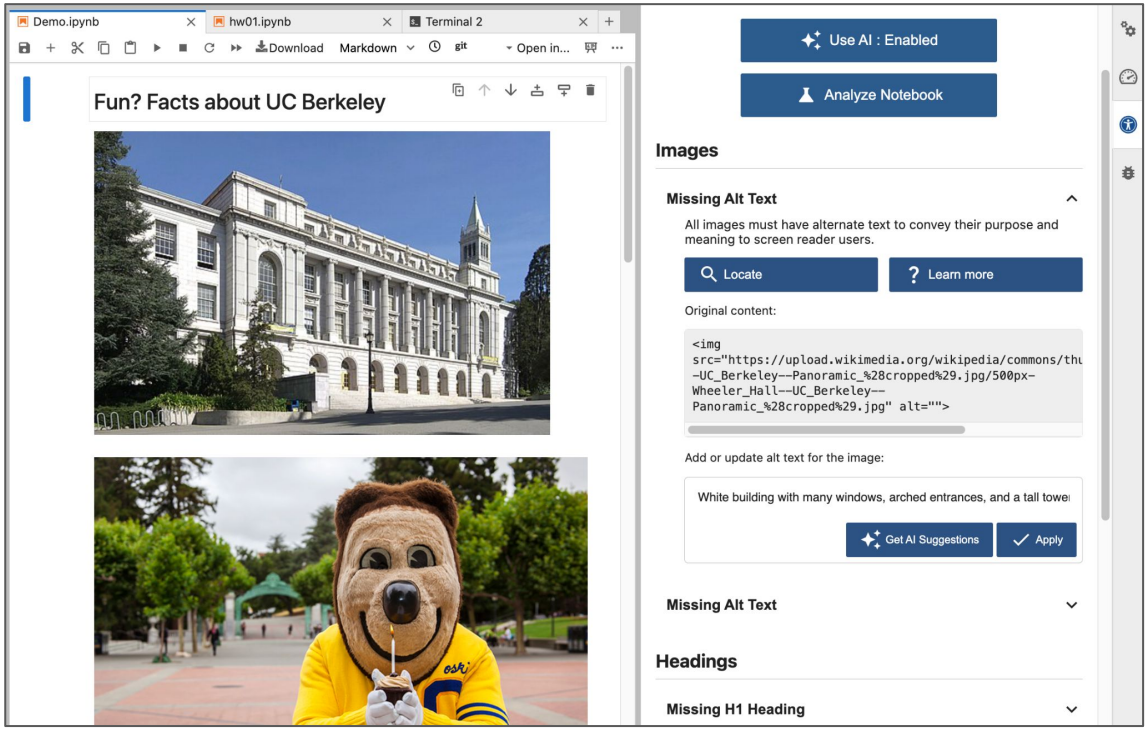
Guideline	Principle	What it Means for Notebooks
1.1.1 Non-text Content	Perceivable	Images and non-text elements need text alternatives (alt text).
1.3.1 Info and Relationships	Perceivable	Structure content logically (headings, lists). Information conveyed by visuals also available programmatically.
1.4.3 Contrast (Minimum)	Perceivable	Contrast ratio of 4.5:1 for text/images of text; 3:1 for large text.
2.4.2 Page Title	Operable	Notebooks need a clear, descriptive title.
2.4.4 Link Purpose	Understandable	Purpose of a link in a notebook must be generally clear from its text or its determined context.
2.4.6 Headings and Labels	Operable	Use clear, descriptive headings and labels with logical hierarchy.

Missing Alt-text Issue (User generated alt-text)



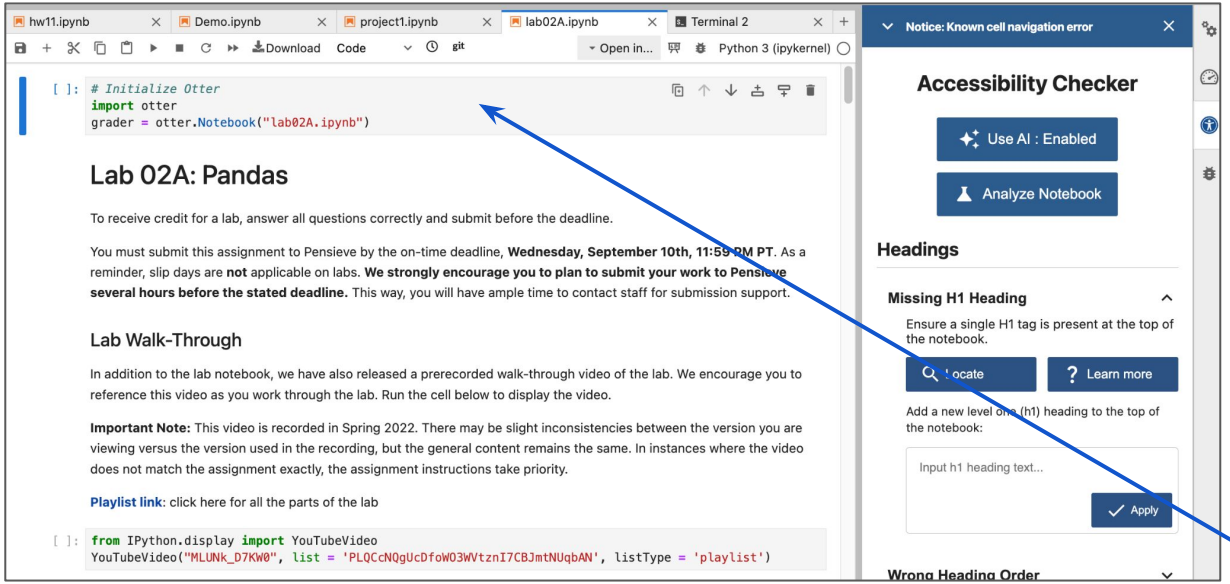
- Extension identifies images in markdown cell that don't have alt text
- Helps notebook authors insert alt-text to the markdown cell

Missing Alt-text Issue (AI generated alt-text)



- Generates an AI-powered description for the image
- Authors can review, edit, and then insert the suggested alt text directly into the markdown cell

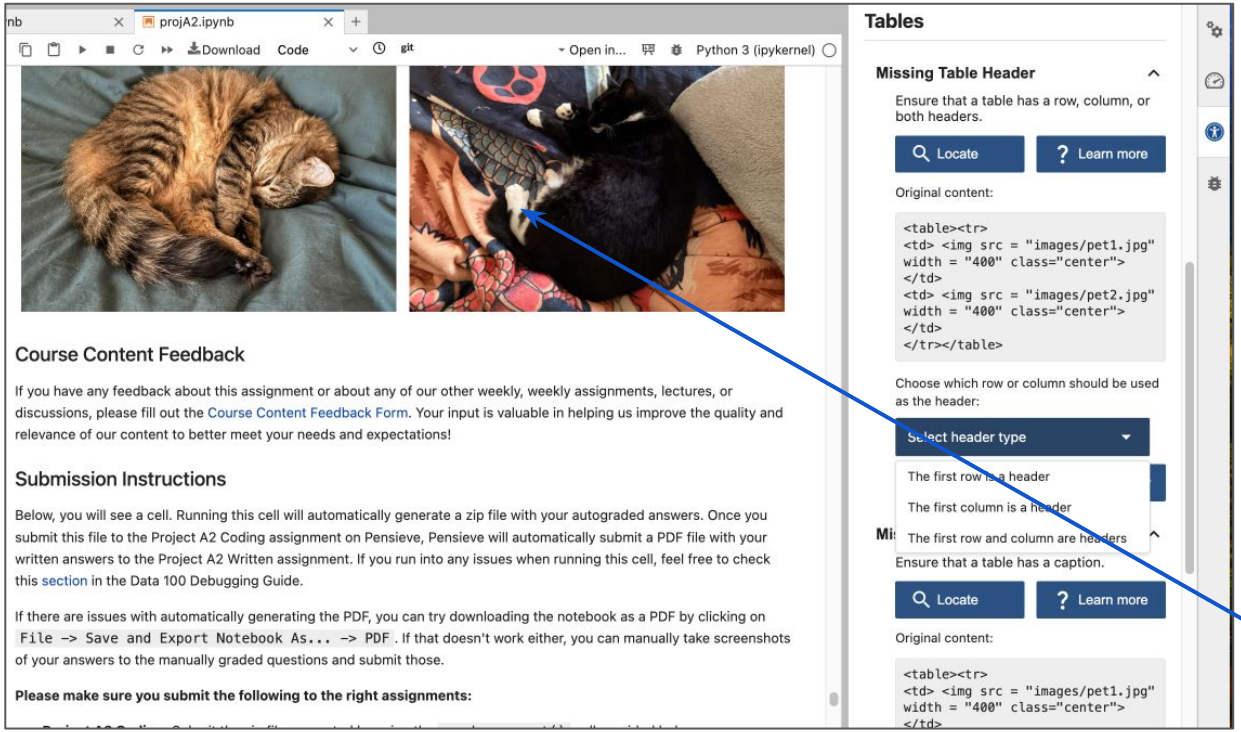
Missing H1 Heading Issue



- Extension detects notebook with a missing H1 heading
- Enables notebook authors to enter a notebook title

Doesn't describe the notebook title

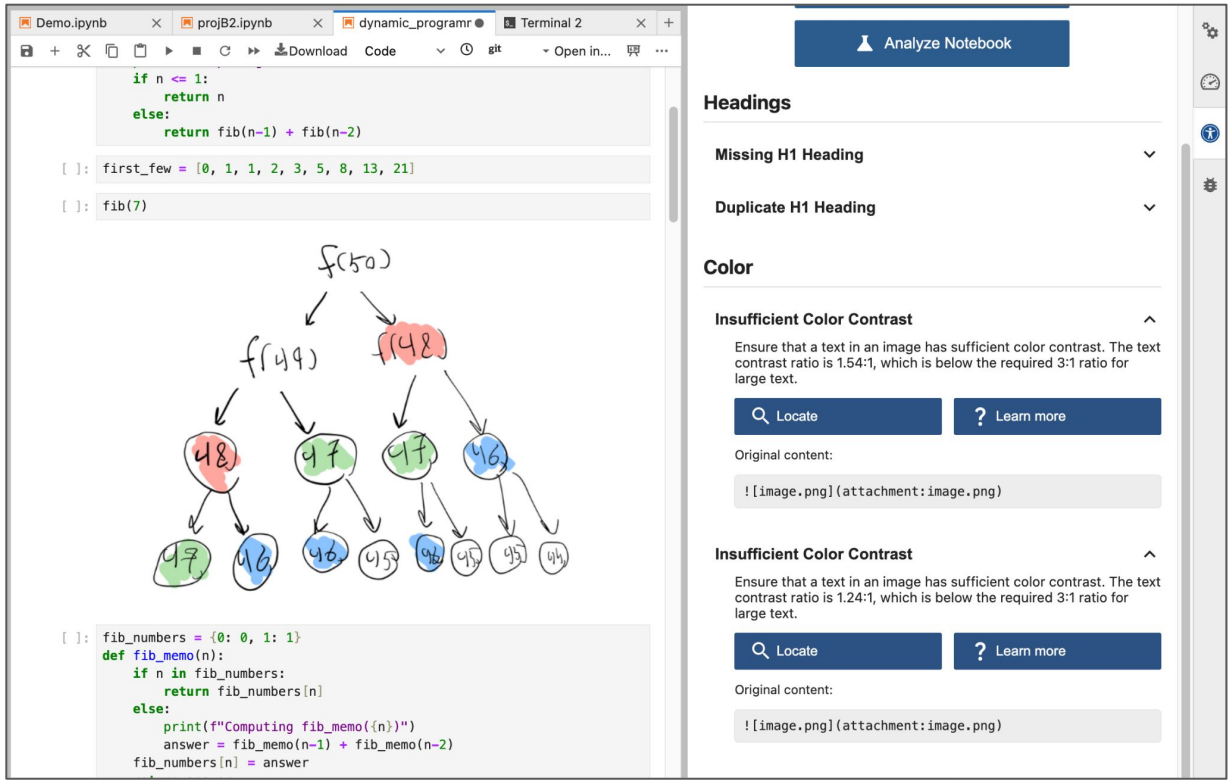
Missing Table Header Issue



- Extension identifies a table without headers
- Notebook authors will be prompted to define the appropriate rows and columns to designate as headers

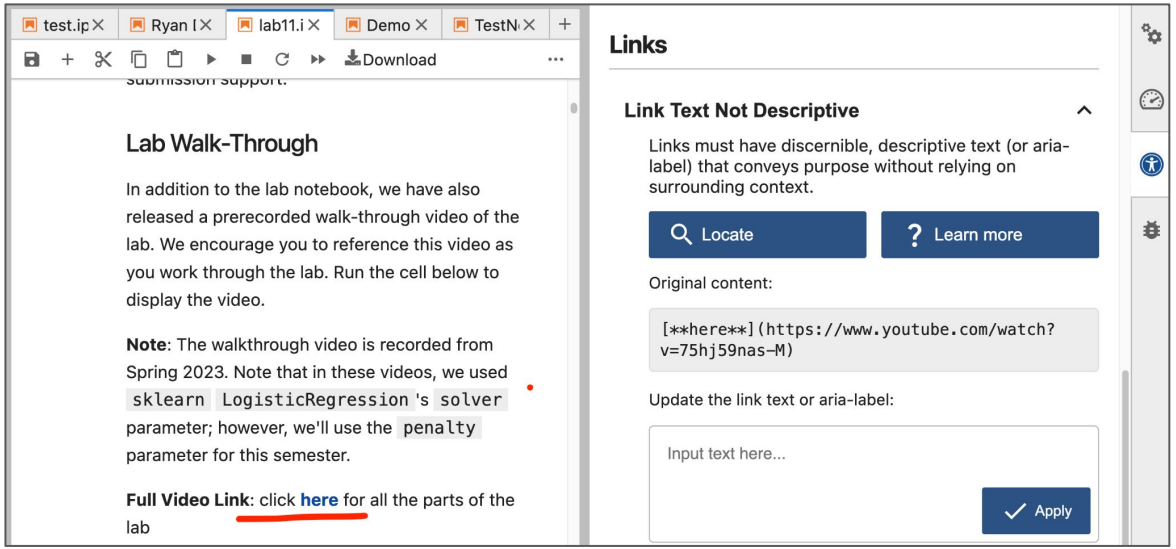
Images are added in the form of a table without headers

Insufficient Color Contrast Issue



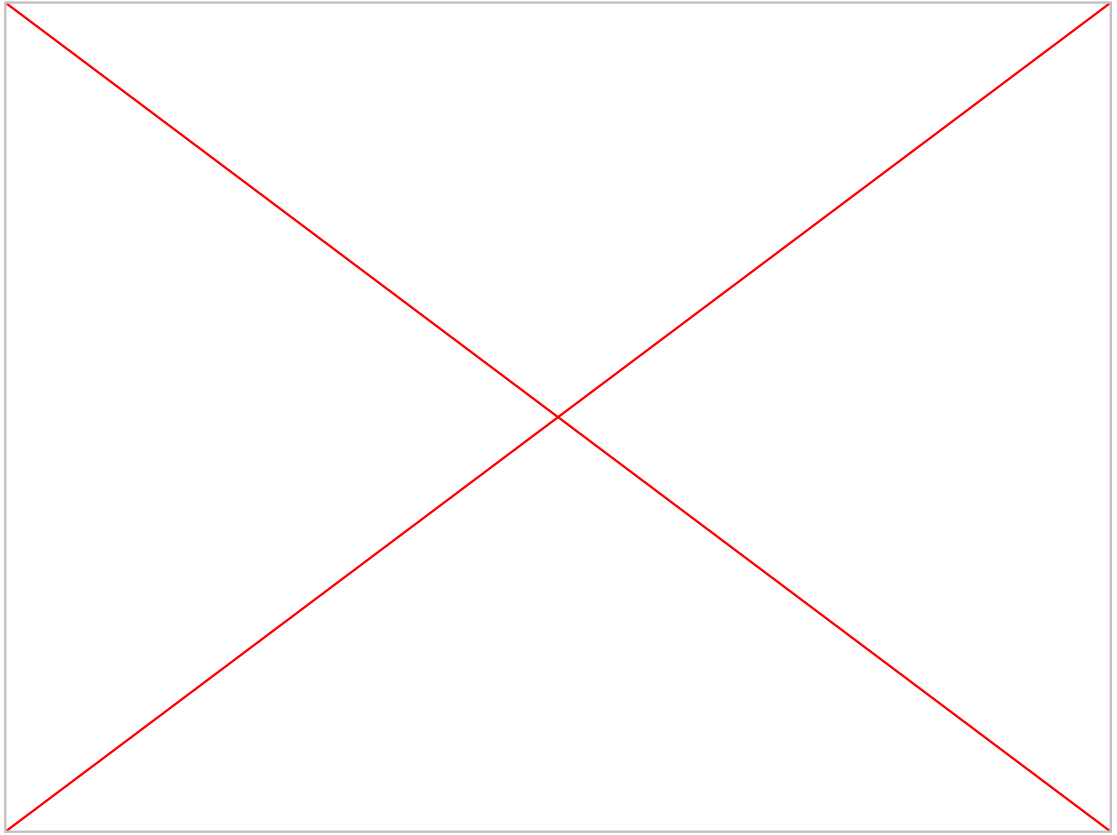
- Extension highlights images in notebooks having text with insufficient color contrast
- It displays the exact color contrast ratio

Link Without Purpose



- Extension analyzes links in notebooks to see if their purpose is clear from the link text alone
- Highlights links with ambiguous link text and enables authors to add descriptive link text

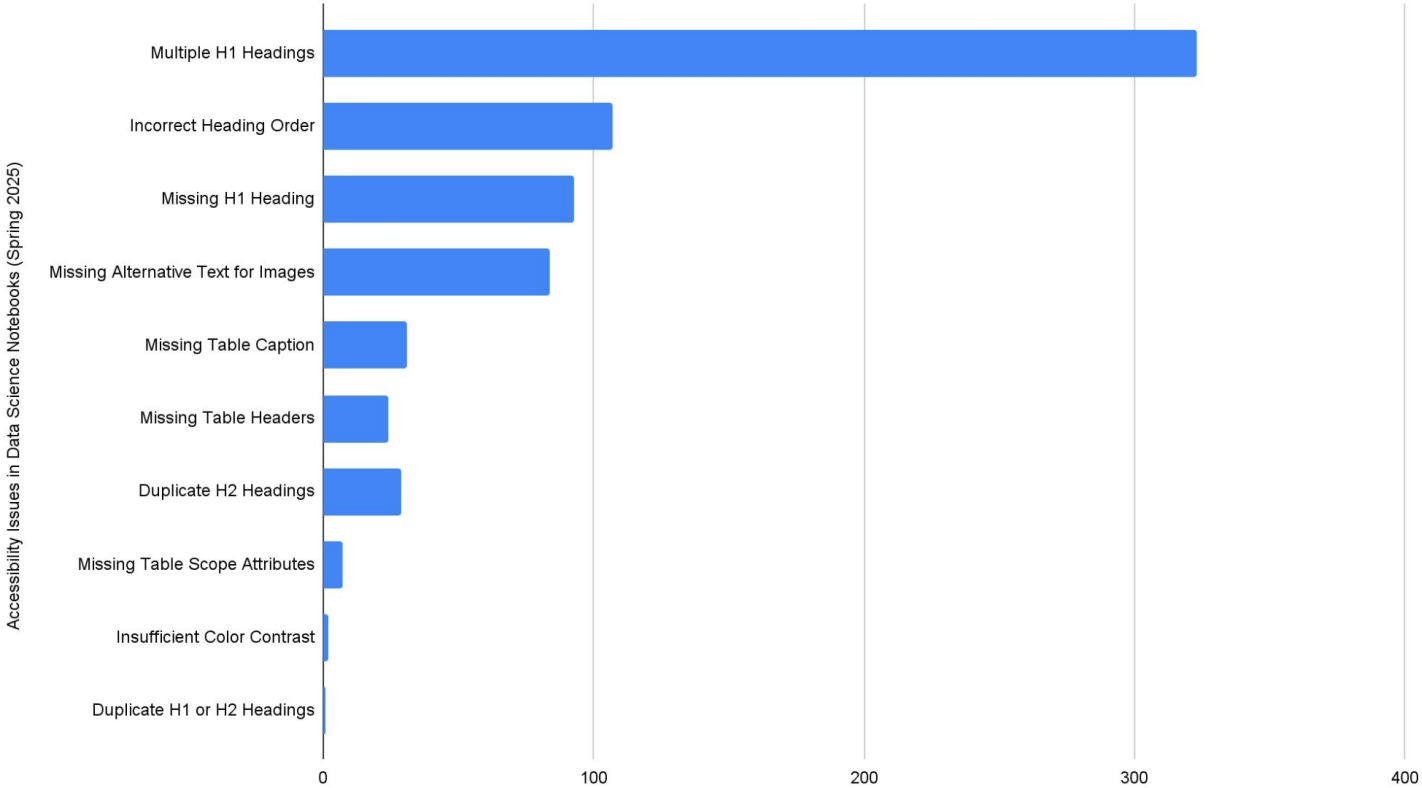
Demo Video



Current State of JupyterLab-a11y-checker

- JupyterLab-a11y-checker is currently being piloted by Data Science courses at UC Berkeley
- Metrics: 9.4k+ downloads since April 2025!
- You can download and install the extension via PyPI

A11y Issues in Data Science Notebooks (Spring 2025)



jupyterlab-a11y-checker 0.2.5

```
pip install jupyterlab-a11y-checker
```

Latest version

Released: Oct 24, 2025

A JupyterLab extension that checks and suggests fixes for a11y issues.

Navigation

- Project description
- Release history
- Download files

Verified details

These details have been verified by PyPI

Maintainers

a11y-checker

Unverified details

*These details have **not** been verified by PyPI*

Project links

- Bug Tracker
- Homepage
- Repository

Meta

- **License:** BSD License (BSD 3-Clause License)

Project description

jupyterlab-a11y-checker

jupyterLab-a11y-checker is a JupyterLab extension that helps authors detect and fix accessibility issues in Jupyter Notebooks, aligning with WCAG 2.1 AA guidelines. It enables authors to identify accessibility issues in their notebooks and provides actionable suggestions to fix them. It combines the strengths of axe-core, a widely used accessibility engine, with custom notebook-specific detection algorithms that address issues axe cannot reliably cover in JupyterLab.

Here's how the extension looks like:

On JupyterLab, this extension is detecting accessibility issues. On the left panel, there is a sample Jupyter Notebook, while on the right side, this extension is displaying image and heading related issues.

Tool Description

Issue Detection

While there are many possible a11y issues in Jupyter Notebooks, we prioritized the issues discussed in a study on Jupyter Notebooks, Notably Inaccessible — Data Driven Understanding of Data Science Notebook (In)Accessibility. To address them, we implement custom detection logic for the issues listed in Rule Description. In addition, we integrate axe-core to detect other standard accessibility issues beyond these main issues, which are listed in Axe Rule Description.

Issue Resolution

Please share your feedback about JupyterLab-a11y-checker extension!



https://bit.ly/jupytera11y

Resources

- Documentation for the a11y-checker extension: https://a11y-checker-guide.datahub.berkeley.edu/
- Extension is currently available for instructors at UC Berkeley via: https://a11y.datahub.berkeley.edu/
- GitHub repository: https://github.com/berkeley-dsep-infra/jupyterlab-a11y-checker
- PyPI: https://pypi.org/project/jupyterlab-a11y-checker/
- WCAG 101: Understanding the Web Content Accessibility Guidelines: https://www.wcag.com/resource/what-is-wcag/
- WCAG POUR Principles: https://accessible.org/wcag-pour-principles-eaa-compliance/

Resources (Continued)

- UC Berkeley Disabled Students' Program Annual Report and Data: https://dsp.berkeley.edu/about/annual-report-and-data
- UC Berkeley DataHub: https://datahub.berkeley.edu
- UC Berkeley By the Numbers: https://www.berkeley.edu/about/by-the-numbers/
- Venkatesh Potluri, Sudheesh Singanamalla, Nussara Tieanklin, and Jennifer Mankoff. 2023. Notably Inaccessible — Data Driven Understanding of Data Science Notebook (In)Accessibility. In The 25th International ACM SIGACCESS Conference on Computers and Accessibility (ASSETS '23), October 22--25, 2023, New York, NY, USA. ACM, New York, NY, USA 19 Pages. https://doi.org/10.1145/3597638.3608417
- Americans with Disabilities Act Title II Regulations: https://www.ada.gov/law-and-regs/regulations/title-ii-2010-regulations/

Acknowledgements

Team



Chanbin Park
Student
Developer



Balaji Alwar
Service Lead
DataHub

Contributors



Vivian Liu
Student
Developer



Shreyas Rana
Student
Developer



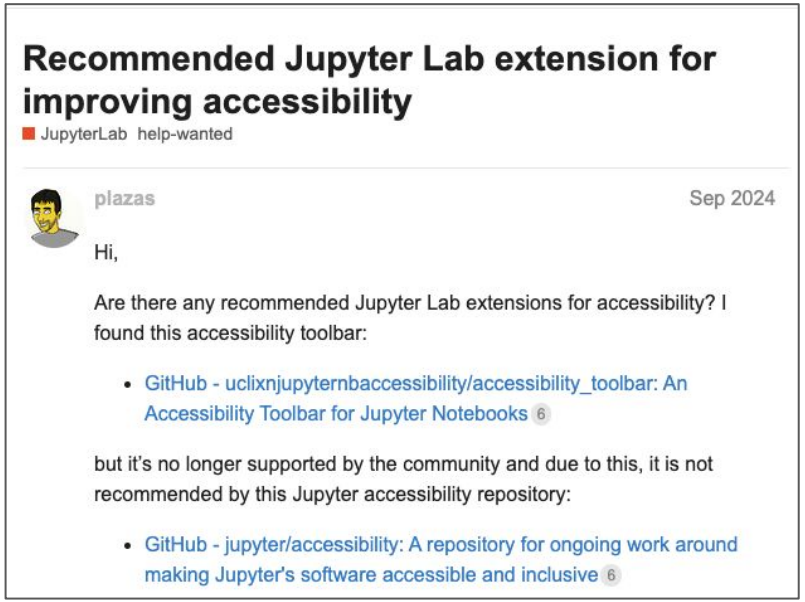
Joe Feria Galicia
Ally Expert



Ryan Lovett
Jupyter
Consultant

Community Input

A11y-checker extension was highlighted in the Jupyter discourse forum



Recommended Jupyter Lab extension for improving accessibility

JupyterLab help-wanted

plazas Sep 2024

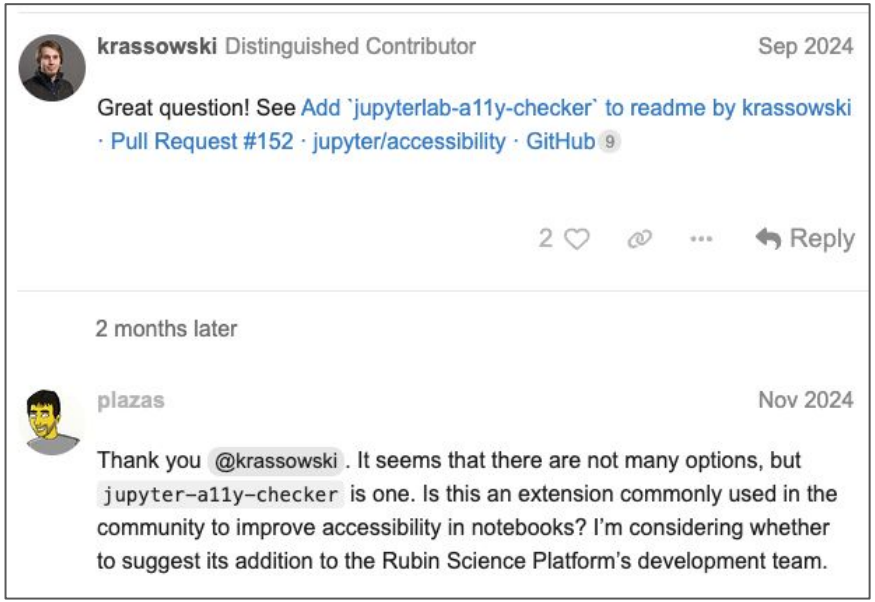
Hi,

Are there any recommended Jupyter Lab extensions for accessibility? I found this accessibility toolbar:

- GitHub - uclixnjupyternbaccessibility/accessibility_toolbar: An Accessibility Toolbar for Jupyter Notebooks 6

but it's no longer supported by the community and due to this, it is not recommended by this Jupyter accessibility repository:

- GitHub - jupyter/accessibility: A repository for ongoing work around making Jupyter's software accessible and inclusive 6



krassowski Distinguished Contributor Sep 2024

Great question! See Add `jupyterlab-a11y-checker` to readme by krassowski · Pull Request #152 · jupyter/accessibility · GitHub 9

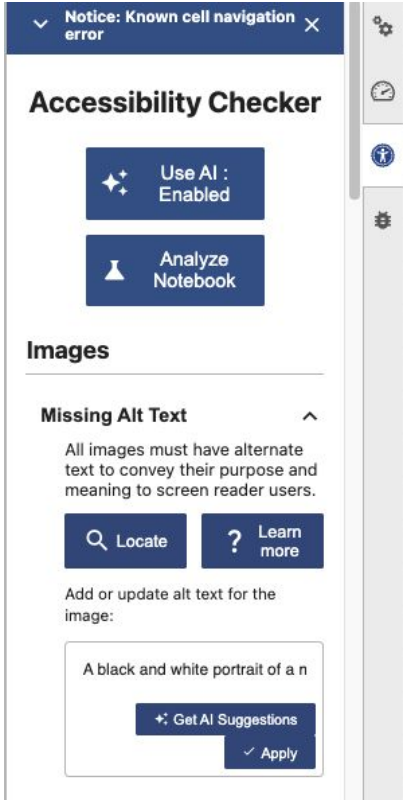
2 Reply

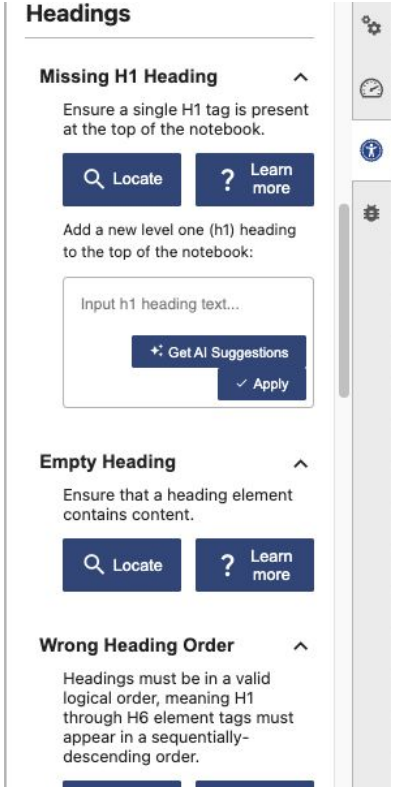
2 months later

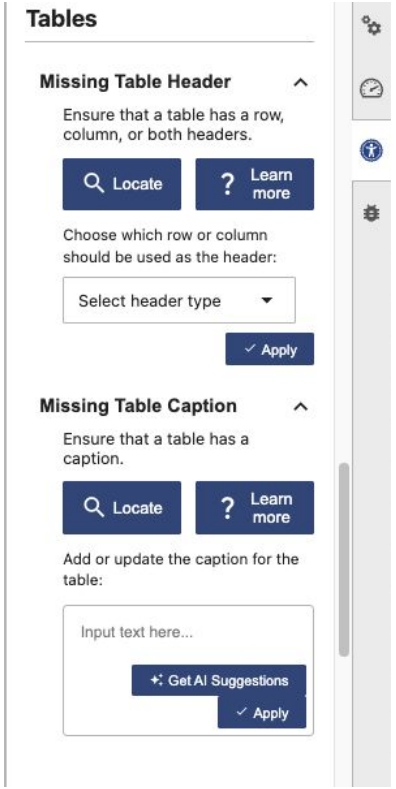
plazas Nov 2024

Thank you @krassowski. It seems that there are not many options, but `jupyter-a11y-checker` is one. Is this an extension commonly used in the community to improve accessibility in notebooks? I'm considering whether to suggest its addition to the Rubin Science Platform's development team.

Extension Screenshots








jupyterhub