

SOFTWARE-DEVELOPMENT

The Good, the Bad and the DataSpell

An honest review of JetBrains's Data Science IDE after a year of using it

*

*



Dor Meir

Published in Towards Data Science

6 min read · Sep 17, 2022

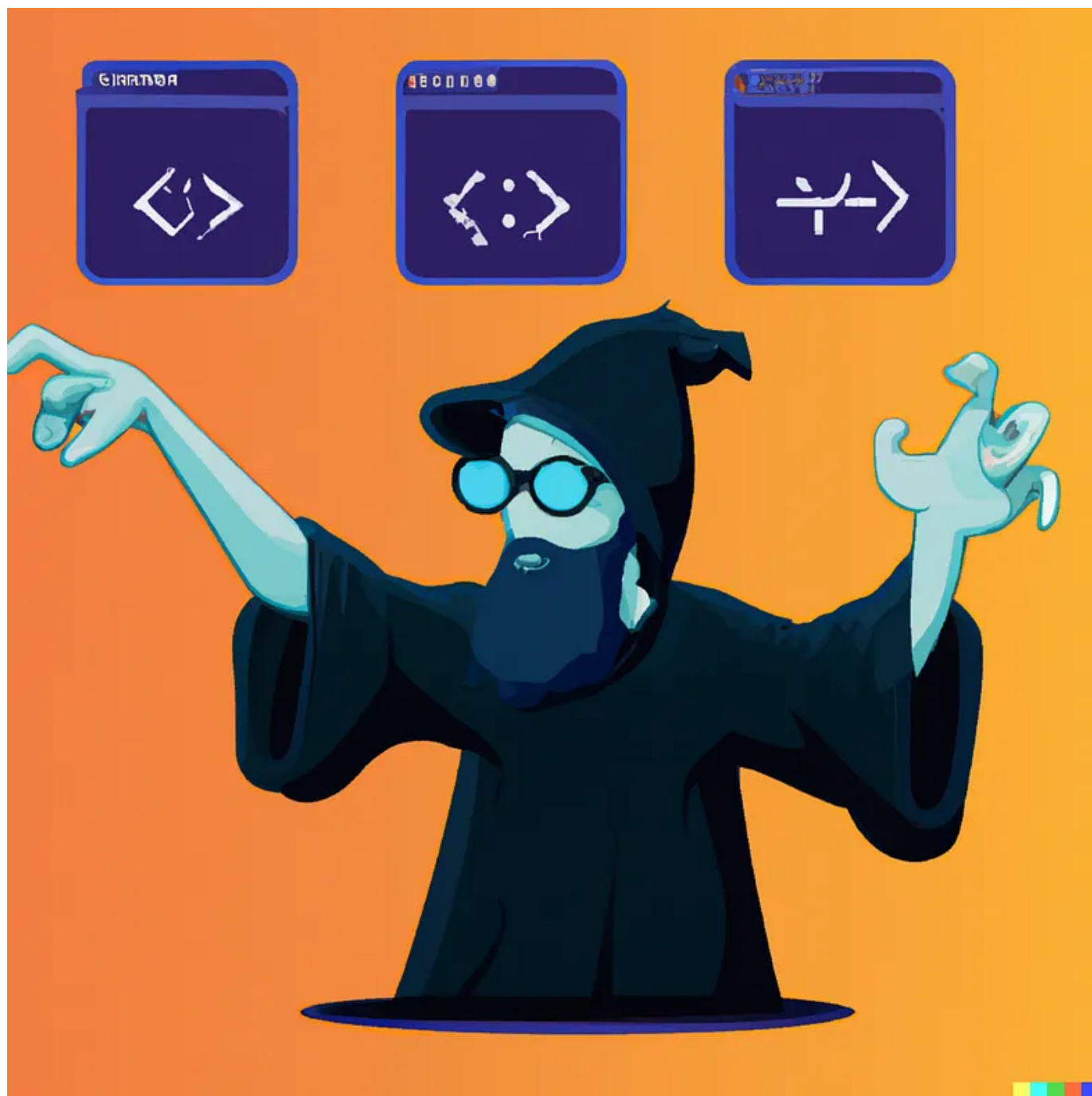


Listen



Share

... More



“a code editor casting a spell”, at least according to [DALL·E 2](#)

Disclaimer (originated in here): This is not a sponsored article. I don’t have any affiliation with DataSpell or its creators. The article shows an unbiased overview of the IDE, intending to make data science tools accessible to the broader masses.

DataSpell was officially released in last November, yet Towards Data Science diligent authors have posted some reviews of the preview edition as early as in August and September. One might say all major DataSpell capabilities were already revealed, **so why another piece on this matter?**

First, things have changed in the IDE since last year. But most importantly, I've been using DataSpell for Data Science research and for developing Machine Learning applications for a whole year now. Hence, **my mileage with the tool is probably high enough to reduce its traits to the most useful and the most painful ones.**

Let us begin with three DataSpell pains. I advise you, though, to not despair from the beginning and keep on reading through, as the tools' three perks are invaluable.

DataSpell issues

1. High usage of resources

DataSpell is probably not even a close competitor in this aspect to other IDE's such as Visual Studio. While JetBrains advices 8 GB RAM is enough for running the program, if you work with datasets of 1M rows and more you shouldn't settle for anything less than 16 GB. CPU-wise, it'll also be unwise to have an older generation than i7 (or other vendors' equivalents), unless you have a lot of time to burn. **The IDE uses considerable amount of resources specifically on start-up** (while initializing the interpreter and scanning files and packages), or if you've attached a folder that's auto-syncing to cloud. Nevertheless, it gets to a pretty descent speed when finishing the start up, or at least to a similar speed to its predecessor, PyCharm.

Name	Status	Command	CPU	Memory	Disk	Network	GPU
DataSpell (8)			44%	84%	22%	0%	0%
workspace		C:\Users\...	26.1%	2,449.1 MB	53.1 MB/s	0 Mbps	0%
rwrapper-x64-windows.exe		C:\Users\...	0%	102.3 MB	0 MB/s	0 Mbps	0%
Python		"c:\Users\...	5.6%	28.0 MB	0 MB/s	0 Mbps	0%
Python		C:\Users\...	0%	7.8 MB	0 MB/s	0 Mbps	0%
Console Window Host		\\?\C:\Wi...	0%	6.8 MB	0 MB/s	0 Mbps	0%
Filesystem events processor		C:\Users\...	0%	1.9 MB	0 MB/s	0 Mbps	0%
Filesystem events processor		C:\Users\...	0%	1.8 MB	0 MB/s	0 Mbps	0%
jupyter-notebook.exe		C:\Users\...	0%	1.8 MB	0 MB/s	0 Mbps	0%
Google Chrome (56)			0.1%	2,385.9 MB	0.1 MB/s	0.1 Mbps	0%
Microsoft Teams (8)			0%	305.7 MB	0 MB/s	0 Mbps	0%

Not the most agile IDE you can find... Image by Author

2. Bugs

The official release still has some bugs, nonetheless, upgrading to newer versions solved more than a few bugs for me. For instance, the embedded Jupyter server had a major bug where it wasn't possible to load a saved notebook, but one later release fixed that issue for me. Another solved bug occurred while connecting to the license server using a proxy, which simply didn't work in prior releases. **One yet unresolved bug is that Jupyter debugger doesn't always work**, and when it does it's quite slow. I cope with that one by simply copying the code to a regular script file and performing the debug over there. I do hope it's only a matter of time until they fix that one, since Jupyter debugging can be useful. In any case, the bugs forum seem to be a very active one, so when encountering a bug you can always ask for a fix and maybe even find a solution.



DataSpell has great features, but quite a few bugs as well. Photo by [Justin Lauria](#) on [Unsplash](#)

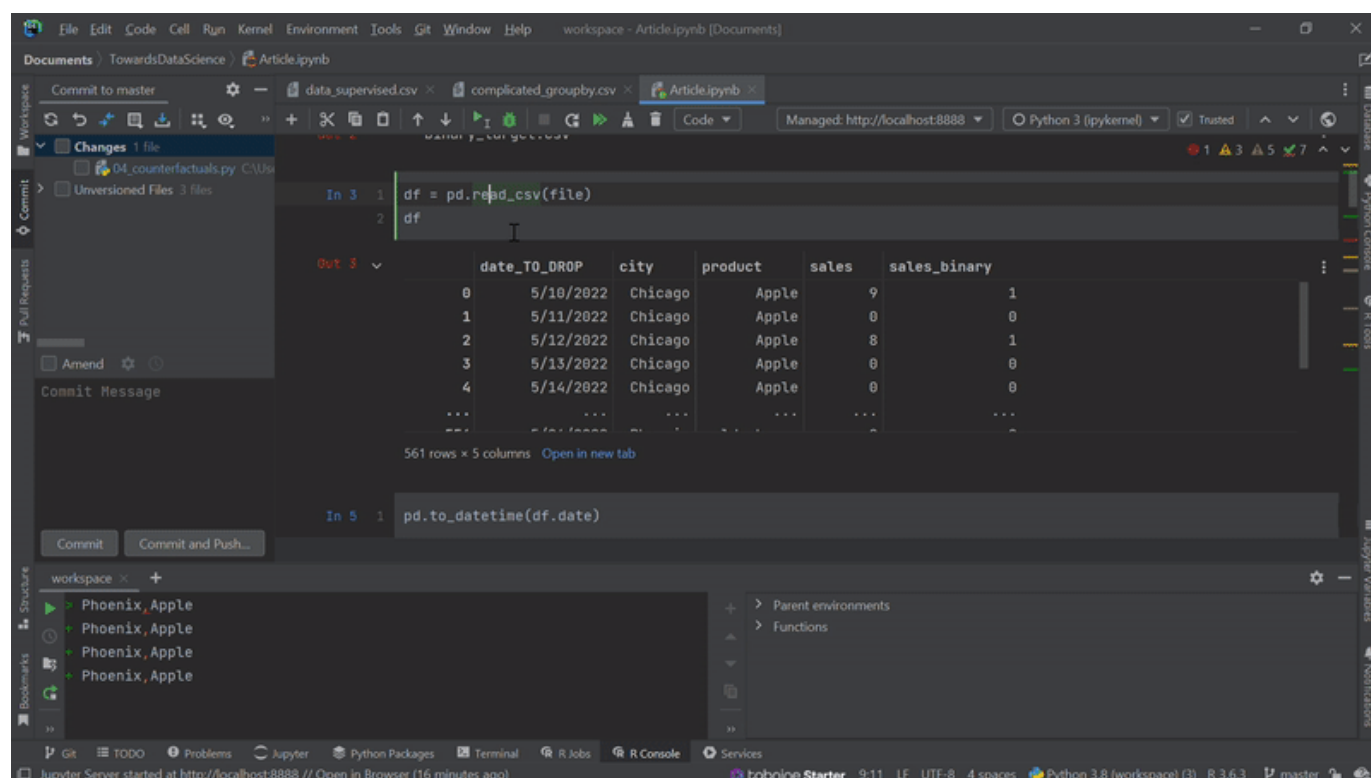
3. It's (sort of) not free

It costs 9\$ a month for personal use, but it's free for students or teachers as part of JetBrains educational package.

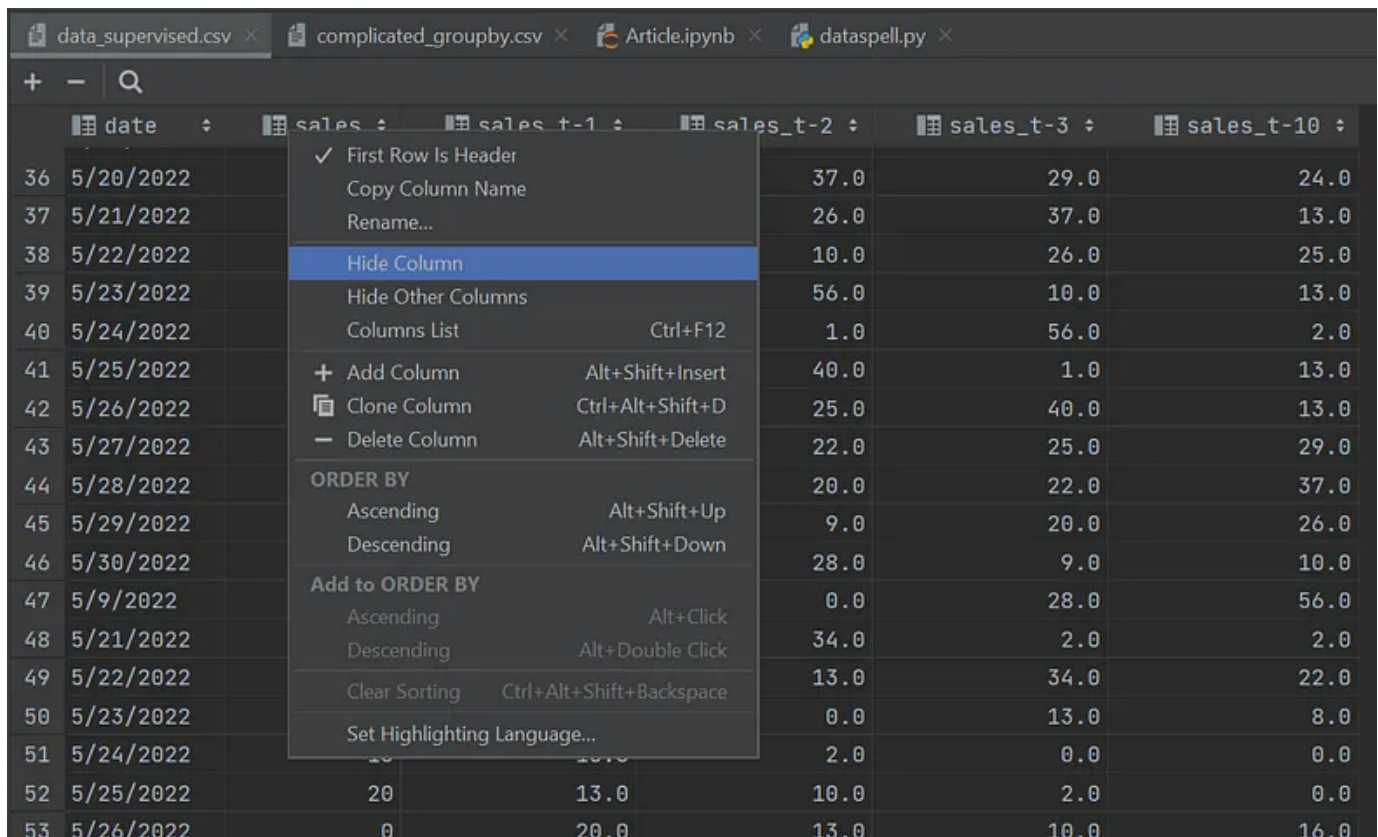
DataSpell perks

1. Comfortable Data view

Clicking *Open in a new tab* opens a wide and long view of the DataFrame. Not only does it allow you to sort or hide columns with a mere mouse click, **running a debug when the data panel's open in a split tab lets you observe how your DataFrame changes *with every line of code* (!)**. This turns out to be remarkably useful when troubleshooting for reasons the code behaves unexpectedly for only some specific cells.



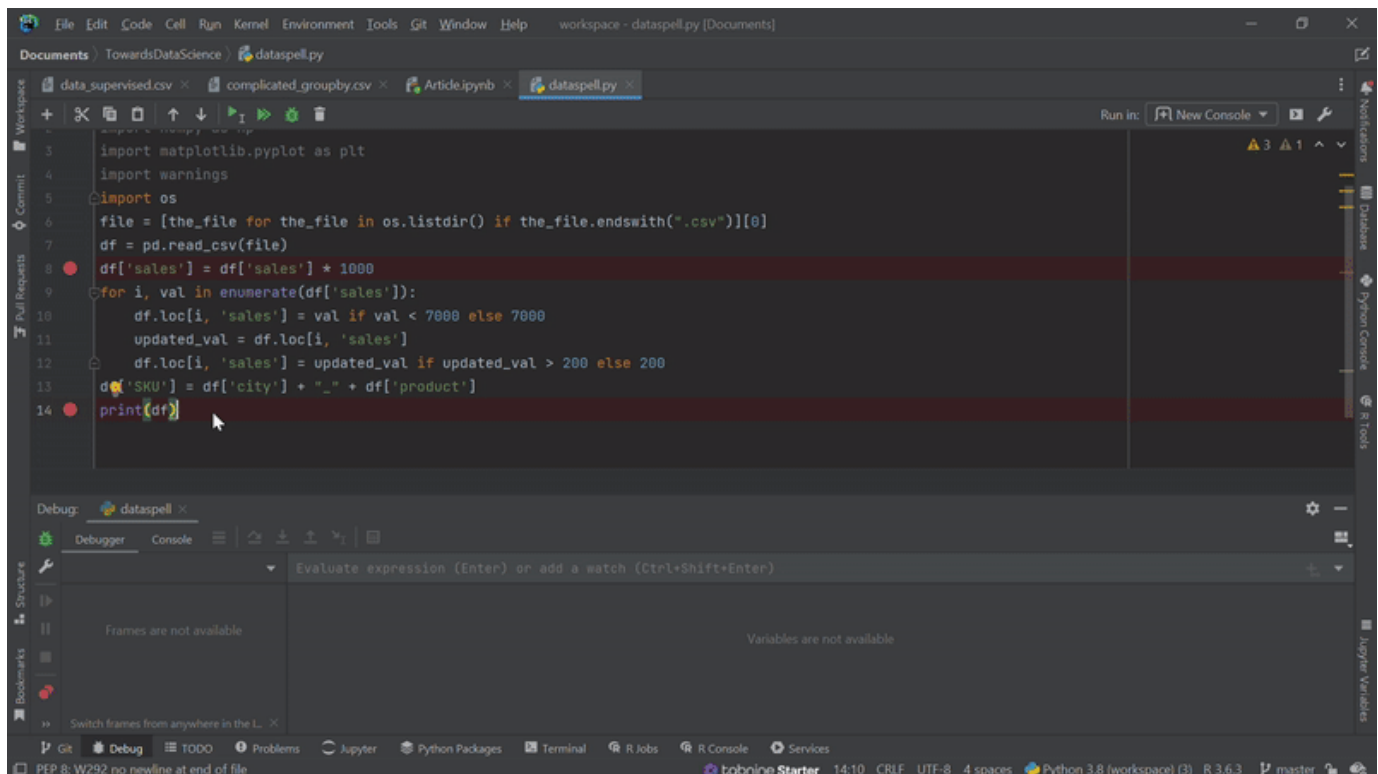
You can forget about `sort_values()` and `pd.options.display.max_rows()`. Image by Author



The screenshot shows the DataSpell application with a CSV file named 'data_supervised.csv' open. A context menu is displayed over the 'sales' column, listing various actions such as 'First Row Is Header', 'Copy Column Name', 'Rename...', 'Hide Column', 'Hide Other Columns', 'Columns List', 'Add Column', 'Clone Column', 'Delete Column', 'ORDER BY' (Ascending, Descending), 'Add to ORDER BY' (Ascending, Descending), 'Clear Sorting', and 'Set Highlighting Language...'. The table data includes columns for 'date', 'sales', 'sales_t-1', 'sales_t-2', 'sales_t-3', and 'sales_t-10'.

	date	sales	sales_t-1	sales_t-2	sales_t-3	sales_t-10
36	5/20/2022			37.0	29.0	24.0
37	5/21/2022			26.0	37.0	13.0
38	5/22/2022			10.0	26.0	25.0
39	5/23/2022			56.0	10.0	13.0
40	5/24/2022			1.0	56.0	2.0
41	5/25/2022			40.0	1.0	13.0
42	5/26/2022			25.0	40.0	13.0
43	5/27/2022			22.0	25.0	29.0
44	5/28/2022			20.0	22.0	37.0
45	5/29/2022			9.0	20.0	26.0
46	5/30/2022			28.0	9.0	10.0
47	5/9/2022			0.0	28.0	56.0
48	5/21/2022			34.0	2.0	2.0
49	5/22/2022			13.0	34.0	22.0
50	5/23/2022			0.0	13.0	8.0
51	5/24/2022			2.0	0.0	0.0
52	5/25/2022	20	13.0	10.0	2.0	0.0
53	5/26/2022	0	20.0	13.0	10.0	16.0

The columns display options. Explaining your work has never been easier. Image by Author



The screenshot shows the DataSpell IDE with a Python script in the editor. The script imports matplotlib.pyplot as plt, import warnings, and import os. It then reads a CSV file, manipulates the 'sales' column, and prints the resulting DataFrame. The debug console at the bottom shows the execution of the script, with a message indicating that frames are not available.

```

1 import matplotlib.pyplot as plt
2 import warnings
3 import os
4
5 file = [the_file for the_file in os.listdir() if the_file.endswith(".csv")][0]
6 df = pd.read_csv(file)
7 df['sales'] = df['sales'] * 1000
8
9 for i, val in enumerate(df['sales']):
10     df.loc[i, 'sales'] = val if val < 7000 else 7000
11     updated_val = df.loc[i, 'sales']
12     df.loc[i, 'sales'] = updated_val if updated_val > 200 else 200
13
14 df['SKU'] = df['city'] + "_" + df['product']
15 print(df)

```

Debug Console:

```

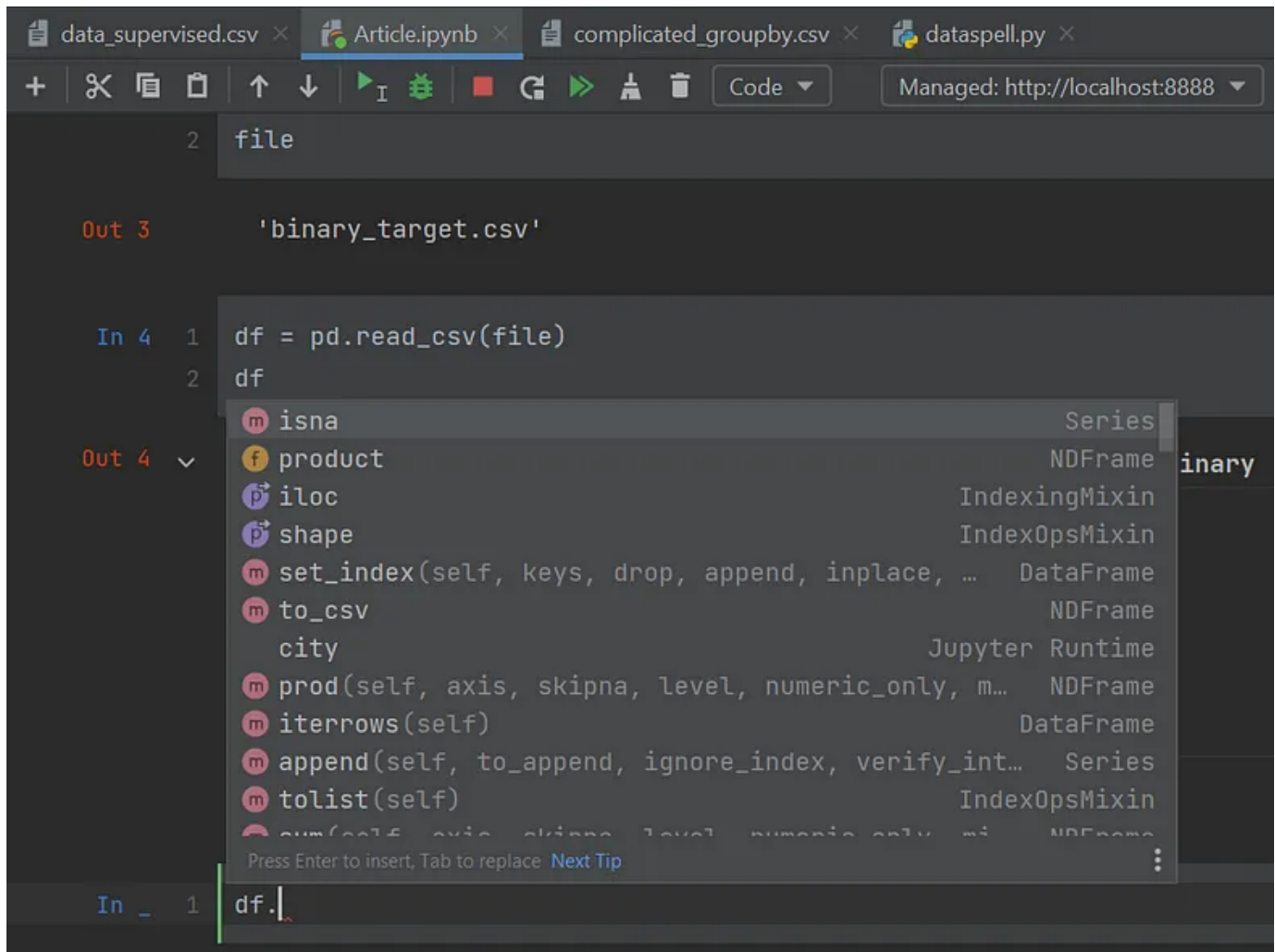
Evaluate expression (Enter) or add a watch (Ctrl+Shift+Enter)
Frames are not available
Variables are not available

```

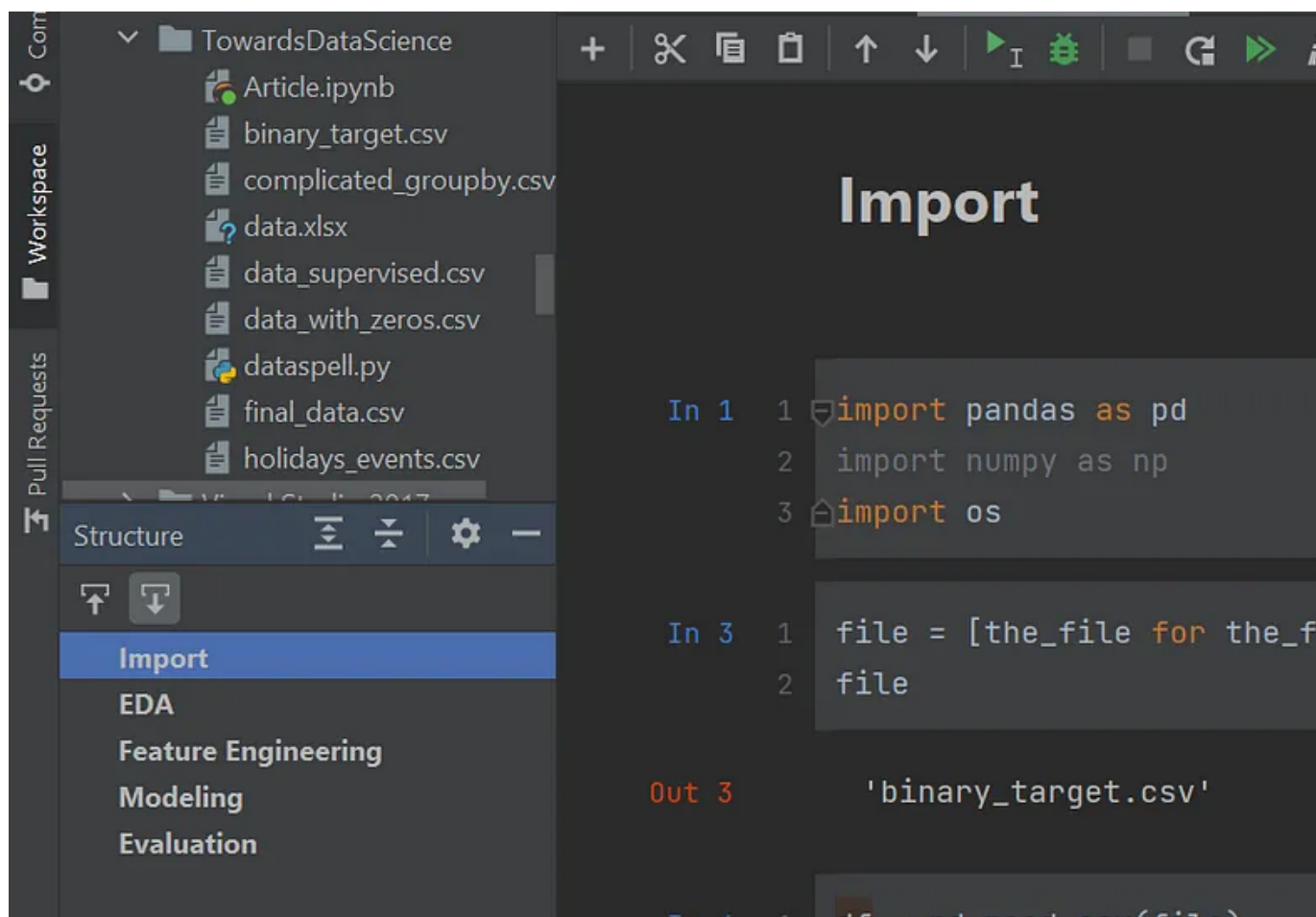
Debug the data manipulation, cell by cell. Pretty cool. Image by Author

2. Enhanced Jupyter

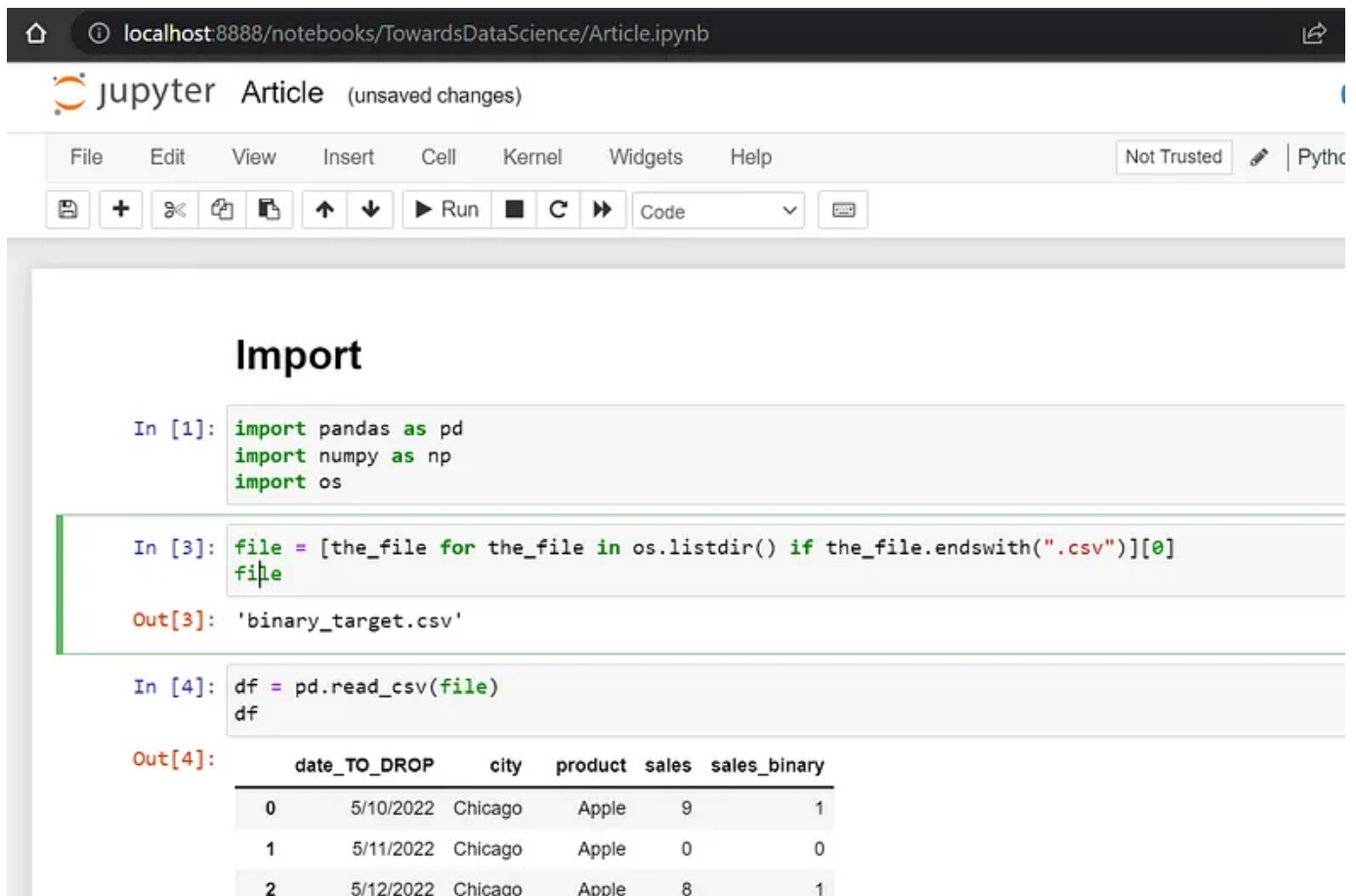
DataSpell improves Jupyter significantly: with a faster and more exhaustive code completion, with an embedded files explorer (no need of Jupyter Lab anymore), with an already embedded table of contents (no need of installing add-ons or figuring out how to make the menu float), and with rows numbering already pre-set for all cells. To sum it up, once you get use to it, **the overall Jupyter experience is simply better.**



DataSpell has basically embedded PyCharm code completion into Jupyter notebook. Image by Author



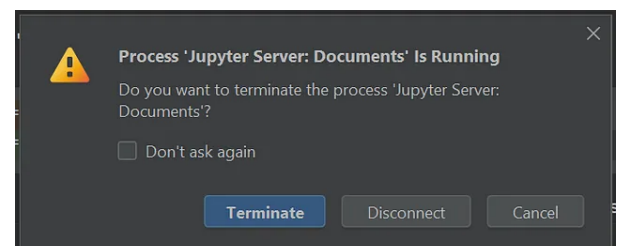
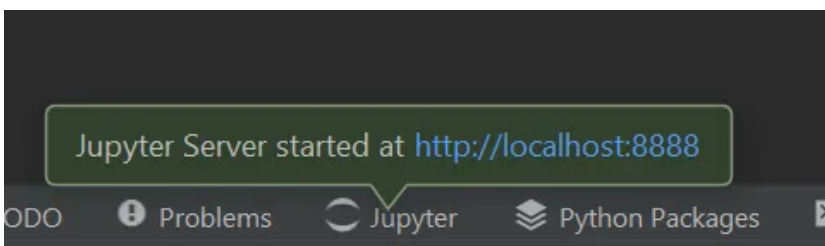
Exploring files, viewing the table of contents, and having line numbering — without any setup. Image by Author



There's no way I'm going back to this... Image by Author

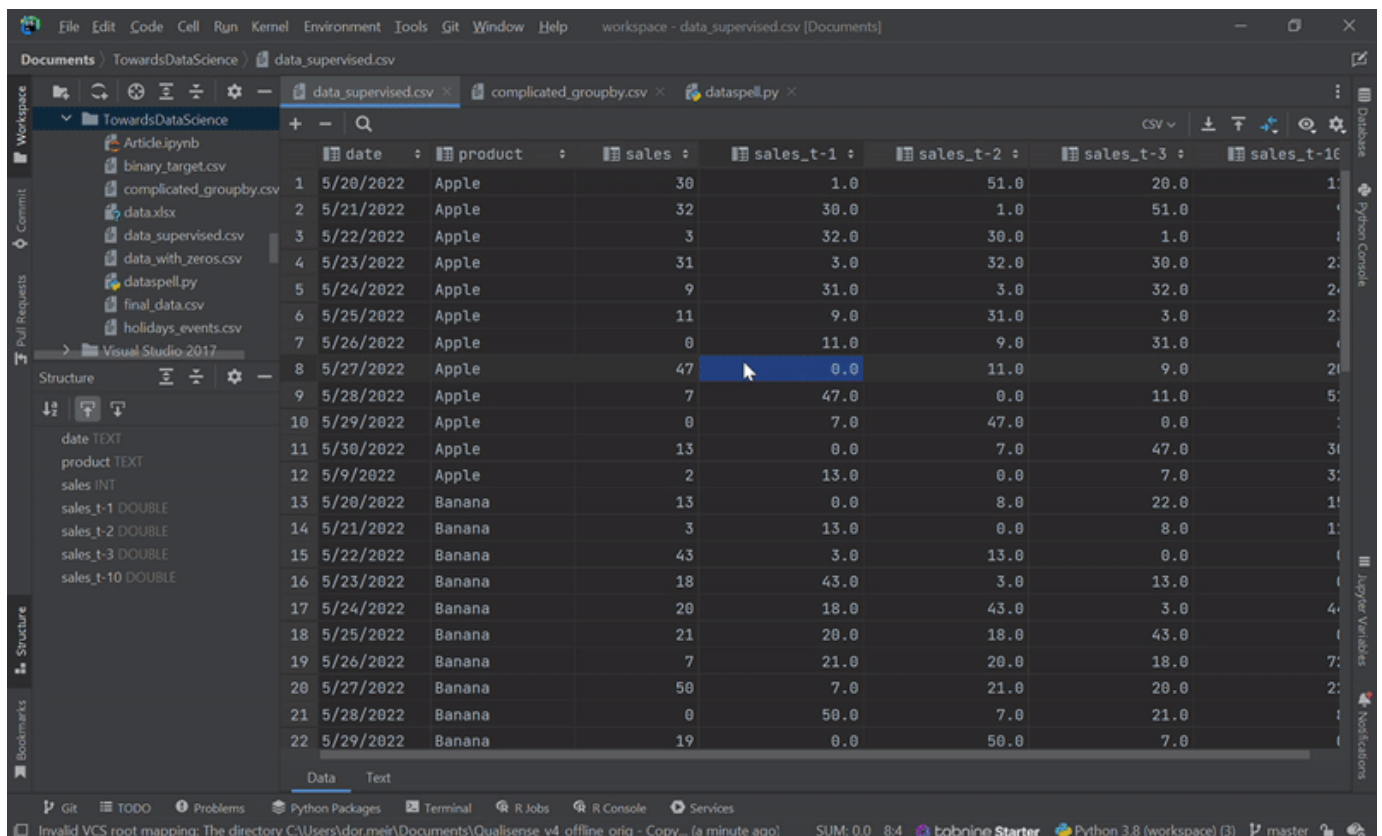
3. One-stop shop

After using DataSpell for a year now, I almost forgot how long and painful it is to do the presumably elementary thing of opening a Jupyter notebook: you have to open anaconda, run *jupyter notebook*, wait for the web browser to open and make sure the folder path is correct. **In DataSpell, however, you directly open a notebook file inside the IDE, and that's about it.** Running the first block of code will start Jupyter server in the background, and a few seconds later you can manipulate the notebook as if it was just another Python script.



It all came down to this... Amazing! Images by Author

Switching between notebooks and scripts will now be outstandingly easy. Moreover, **Research and Development in DataSpell is much more intertwined: instead of writing a load of functions in a notebook and copying them in the end to a script, DataSpell makes it immensely easier to write functions directly to production script and import them back to the notebook, for further research.** Thus, Dataspell is effectively allowing us to enjoy the true benefits of two tools: both the interactive and graphical outputs of Jupyter, and the fast and efficient code writing of PyCharm.



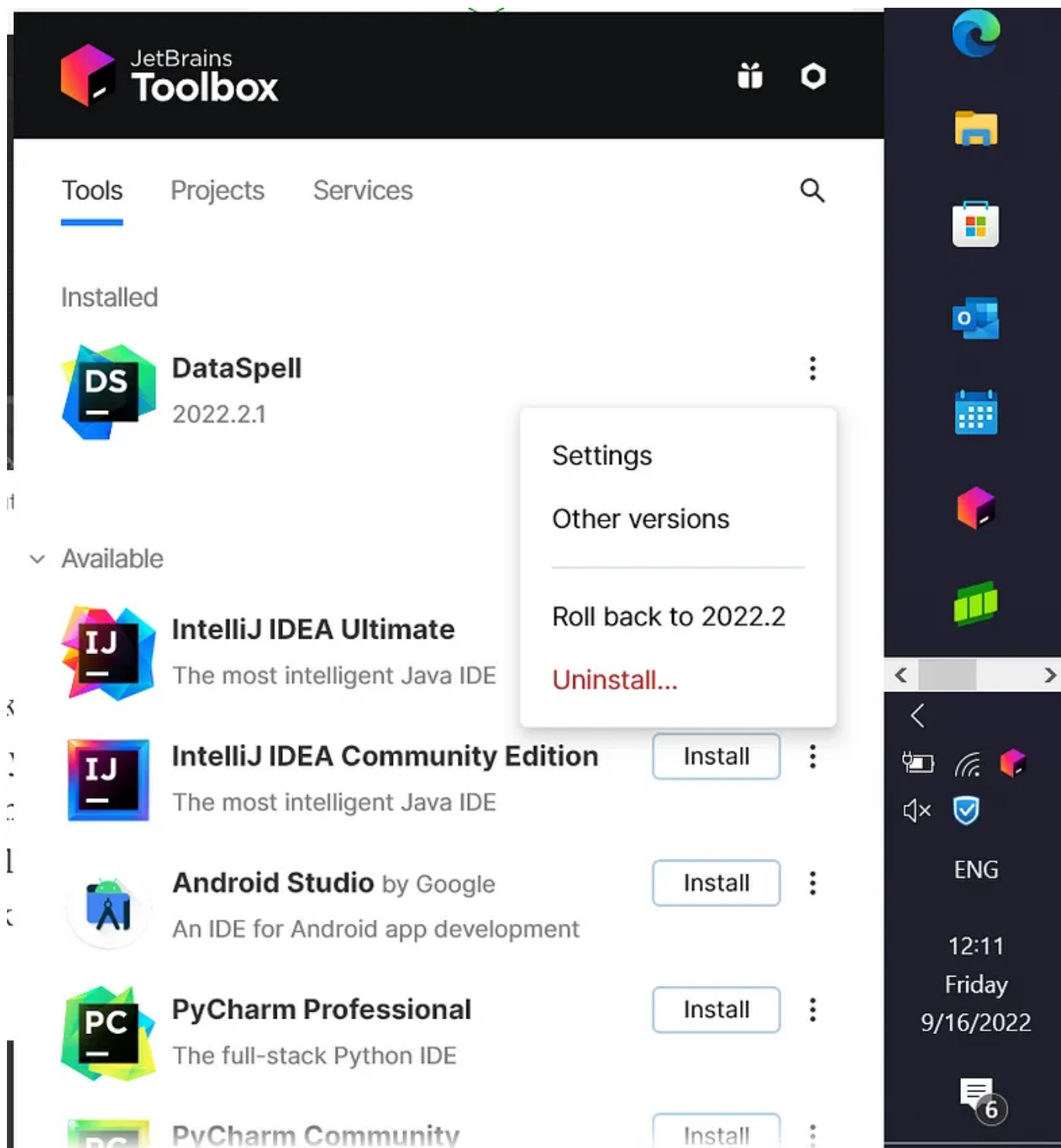
You just need to run one block of code to open Jupyter. Image by Author

So, it's fairly straightforward to run Jupyter from DataSpell. More importantly, the IDE functions as a true one-stop shop, as: performing research (in a notebook), exploring the data (in the data panel), developing and debugging production ready code (in a script) — are all done in the same place!

The IDE can be viewed as a true one-stop shop, as: performing research (in a notebook), exploring the

data (in the data panel), developing and debugging production-ready code (in a script) — are all done in the same place!

This DataSpell trait becomes crucial when working in a client's environment which is highly secured and sensitive. You no longer must explain the need for separate installations of PyCharm, Jupyter Notebook or Lab, or Microsoft Excel. You can get most of these tools' benefits by installing just one exe file. Furthermore, if you're concerned about updating the tool regularly and managing other JetBrains tools, try installing the JetBrains Toolbox that enables updating and downgrading DataSpell (and other tools) in a manner of a mouse click.



JetBrains Toolbox can easily open, update, roll-back and uninstall DataSpell. Image by Author

Summary

DataSpell is JetBrains' flagship IDE for Data Scientists. It's not the most lightweight IDE, it still suffers from bugs and most chances you'd have to pay for it. Still, the benefits probably outweigh the pains: the interactive way of viewing data, the enhanced Jupyter experience and having Research and Development all in one place and intertwined — make it worthwhile.

• • •

Feel free to share your feedback and contact me on LinkedIn.

Thank you for reading, and good luck! 🍀

Dataspell

Jetbrains

Data Science

Jupyter Notebook

Development