Spyder and the NumFOCUS SDG program: better UI/UX, improved code completion and lessons learned

About Spyder
Spyder is a powerful scientific environment written in Python, for Python, and designed by and for scientists, engineers and data analysts

About NumFOCUS
NumFOCUS is a 501(c)(3) public charity in the United States supporting open code for better science

NumFOCUS Small Development Grants
The Small Development Grants program is a NumFOCUS microgrant program benefitting NumFOCUS Sponsored and Affiliated projects.
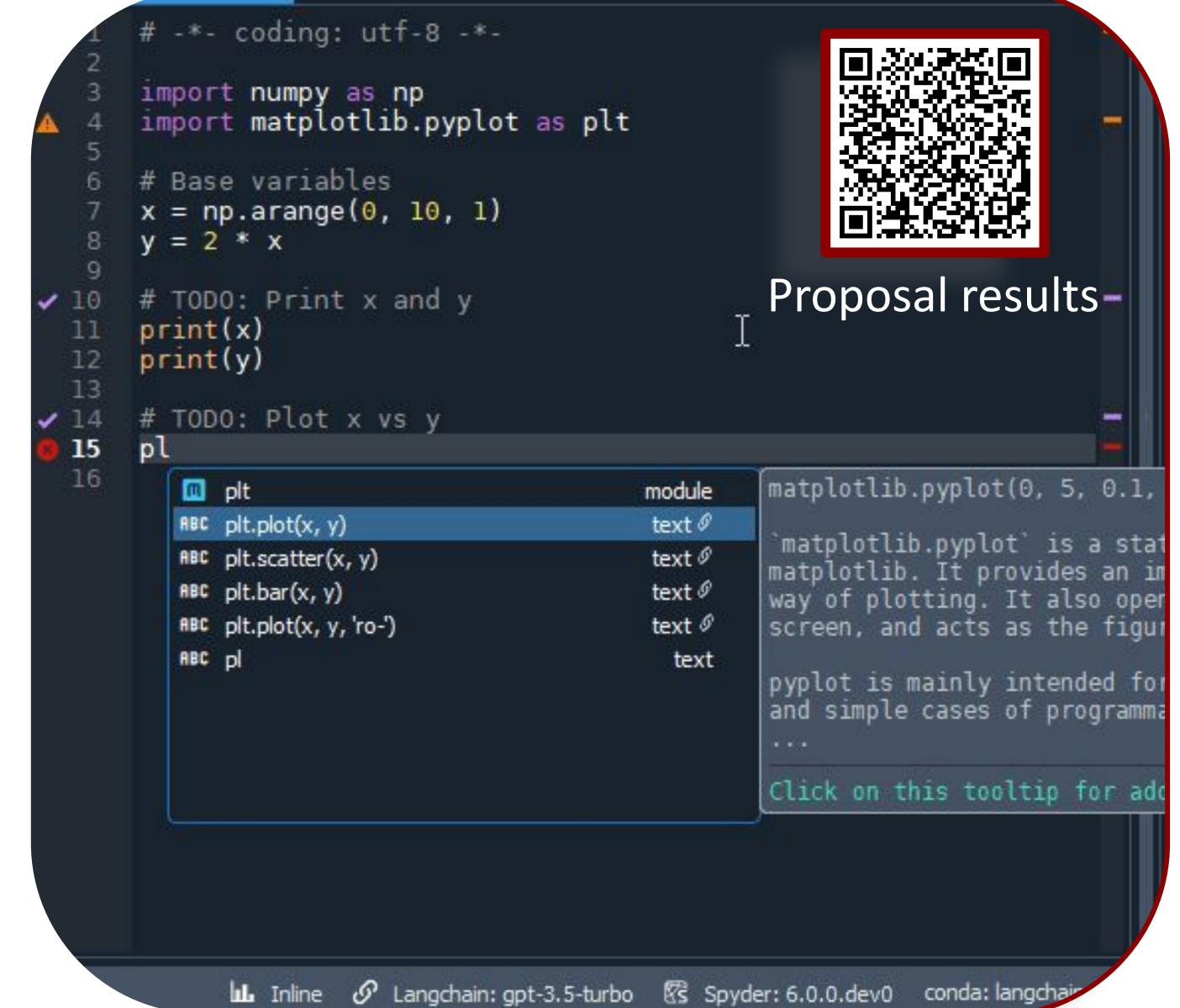
Small Development Grants page

Improving Spyder IDE UX/UI
- Cells support improvements (button to add Cells and full width Editor length divider)
- Menu to create IPython consoles for specific Python environments/interpreters
- Show an explanatory message for panes that appear empty at startup
- Spyder 2 icon set removal and other improvement to the UI elements (tabs and preference tooltips)
- Improve the usability of the Variable Explorer (filter variables, reload variables, and tooltips additions)

Full proposal   Proposal results

Improving external completion plugins development support in Spyder

Full proposal

```
1   # -*- coding: utf-8 -*-
2
3   import numpy as np
4   import matplotlib.pyplot as plt
5
6   # Base variables
7   x = np.arange(0, 10, 1)
8   y = 2 * x
9
10  # TODO: Print x and y
11  print(x)
12  print(y)
13
14  # TODO: Plot x vs y
15  pl
16
```

Proposal results

```
    plt                            module
ABC plt.plot(x, y)                 text
ABC plt.scatter(x, y)              text
ABC plt.bar(x, y)                  text
ABC plt.plot(x, y, 'ro-')          text
ABC pl                             text
```

```
matplotlib.pyplot(0, 5, 0.1,
`matplotlib.pyplot` is a stat
matplotlib. It provides an im
way of plotting. It also ope
screen, and acts as the figu

pyplot is mainly intended for
and simple cases of programma
...

Click on this tooltip for add
```

Inline   Langchain: gpt-3.5-turbo   Spyder: 6.0.0.dev0   conda: langchain

Lessons learned
- Start logistics as quickly as possible when proposals get approved and keep an eye on the processes that need to be followed (contractor agreement draft and signing, be sure funds are available to make invoices, follow invoice approvals and payments).
- Importance of planning ahead possible topics for proposals (keep an eye on the grant round cycles start and end dates)
- Being sure to keep track of feedback received when making proposals to structure future ideas
- The need for continuous discussion when planning and executing a proposal (even more when changes to be made are UX/UI related).
- Having a well-scoped plan for a proposal is essential, but understanding that the plan needs to have some level of flexibility is also important.
- Even though you could get interesting feedback while a proposal is being made, be sure to always have in mind and communicate the proposal scope. Use the good feedback to define future work

Daniel Althviz Moré | Quansight Labs, Spyder Core Dev

Juan Sebastian Bautista Rojas | Spyder Junior Dev