IPyslides From Markdown

Author: Abdul Saboor

Creating Slides

Assuming you have `ls = ipyslides.LiveSlides()`

- Proceed to create slides:
  - Edit and test cells in `ls.convert2slides(False)` mode.
  - Run cells in `ls.convert2slides(True)` mode from top to bottom.
  - `%%slide integer` on cell top auto picks slide and `%%title` auto picks title page.
  - `%%slide integer -m` can be used to create slide from full markdown (extended one).
  - You can use context managers like `with ls.slide(): ...` and `with ls.title(): ...` in place of `%%slide` and `%%title` respectively.

Python

```python
import ipyslides as isd
ls = isd.LiveSlides()
ls.settings.set_animation('zoom')
```

Python

```
%%title
# create a rich content title page
```

Python

Slide 1

This is inline markdown parsed by magic

Version: 1.6.5 as executed from below code in markdown.

Python

```
import ipyslides as isd
version = isd.__version__
%xmd #### This is inline markdown parsed by magic {.Note .Warning}
```

Slide 2

Created using `%%slide 2 -m` with markdown only

Column A

Sub column A

Column B

Sub column B

That version from last slide is still in memory. See it is there 1.6.5

Slide 3

You can call few functions in double pairs of curly brackets to add different content directly in markdown. Get list of them by `ipyslides.parsers.markdown_callables`

('cite', 'citations_html', 'insert_notes', 'notes.insert', 'source.from_callable', 'source.from_file', 'details', 't

I am Alerted and I am *colored and italic text*

IPySlides Online Running Sources

ⓘ Launch as voila prs (may not work as expected [1]) launch binder

ⓘ Edit on Kaggle

ⓘ Launch example Notebook launch binder

1. Add references like this per slide. Use prs.cite() to add citations generally. ↩

IPython Display Objects

Any object with following methods could be in `write` command:

`_repr_pretty_` , `_repr_html_` , `_repr_markdown_` , `_repr_svg_` , `_repr_png_` , `_repr_jpeg_` , `_repr_latex_` , `_repr_json_` , `_repr_javascript_` , `_repr_pdf_` Such as `IPython.display.<HTML,SVG,Markdown,Code>` etc. or third party such as `plotly.graph_objects.Figure` .

Plots and Other Data Types

These objects are implemented to be writable in `write` command: `matplotlib.pyplot.Figure`, `altair.Chart`, `pygal.Graph`, `pydeck.Deck`, `pandas.DataFrame`, `bokeh.plotting.Figure`, `IPython.display.Image` Many will be extentended in future. If an object is not implemented, use `display(obj)` to show inline or use library's specific command to show in Notebook outside `write`.

Interactive Widgets

Any object in `ipywidgets` Link to ipywidgtes right here using `textbox` command

or libraries based on ipywidgtes such as `bqplot` , `ipyvolume` ,plotly's `FigureWidget` [2](reference at end) can be included in `iwrite` command as well as other objects that can be passed to `write` with caveat of Javascript.

Commands which do all Magic!

LiveSlides.**write**(*columns, width_percents=None, className=None)

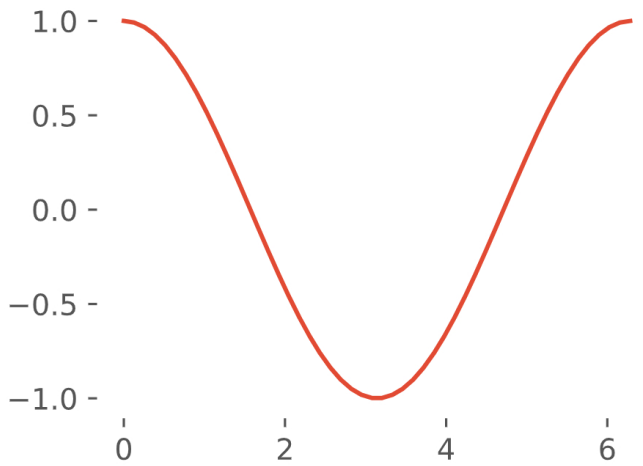
Writes markdown strings or IPython object with method `_repr_<html,svg,png,...>_` in each column of same with. If width_percents is given, column width is adjusted. Each column should be a valid object (text/markdown/html/ have *repr or to* method) or list/tuple of objects to form rows or explictly call `rows`.

- Pass int,float,dict,function etc. Pass list/tuple in a wrapped list for correct print as they used for rows writing too.
- Give a code object from `LiveSlides.source.context[from_...]` to it, syntax highlight is enabled.
- Give a matplotlib `figure/Axes` to it or use `ipyslides.objs_formatter.plt2html()`.
- Give an interactive plotly figure.
- Give a pandas dataframe `df` or `df.to_html()`.
- Give any object which has `to_html` method like Altair chart. (Note that chart will not remain interactive, use display(chart) if need interactivity like brushing etc.)
- Give an IPython object which has `_repr_<repr>_` method where is one of ('html','markdown','svg','png','jpeg','javascript','pdf','pretty','json','latex').
- Give a function/class/module (without calling) and it will be displayed as a pretty printed code block.
- Give a registered object using `@LiveSlides.serializer.registor` decorator.

If an object is not in above listed things, `obj.__repr__()` will be printed. If you need to show other than **repr**, use `display(obj)` outside `write` command or use methods specific to that library to show in jupyter notebook.

If you give a className, add CSS of it using `format_css` function and provide it to `write` function. Get a

Plotting with Matplotlib')



Python

```python
import numpy as np, matplotlib.pyplot as plt
x = np.linspace(0,2*np.pi)
with plt.style.context('ggplot'):
    fig, ax = plt.subplots(figsize=(3.4,2.6))
    _ = ax.plot(x,np.cos(x))
write([ax, s.focus_lines([1,3,4])])
```

Watching Youtube Video?

Python

```python
from IPython.display import YouTubeVideo
prs.write(YouTubeVideo('Z3iR551KgpI',width='100%',height='266px'))
@prs.notify_later()
def push():
    t = time.localtime()
    return f'You are watching Youtube at Time-{t.tm_hour:02}:{t.tm_min:02}'
s.display()
```

Data Tables

h1	h2	h3
d1	d2	d3
r1	r2	r3

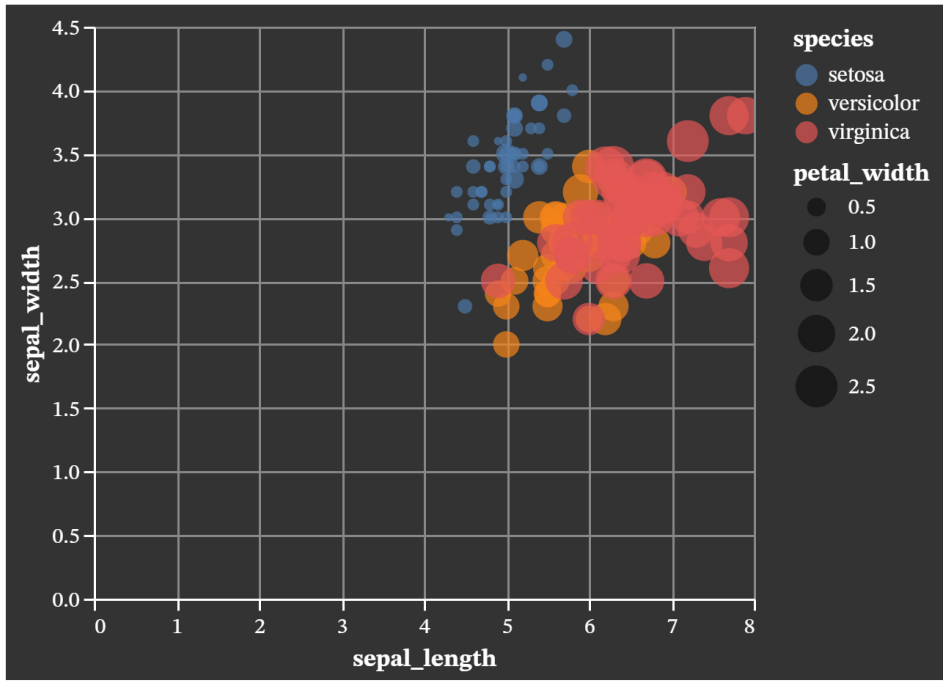
Plotly and Pandas DataFrame only show if you have installed

Writing Pandas DataFrame

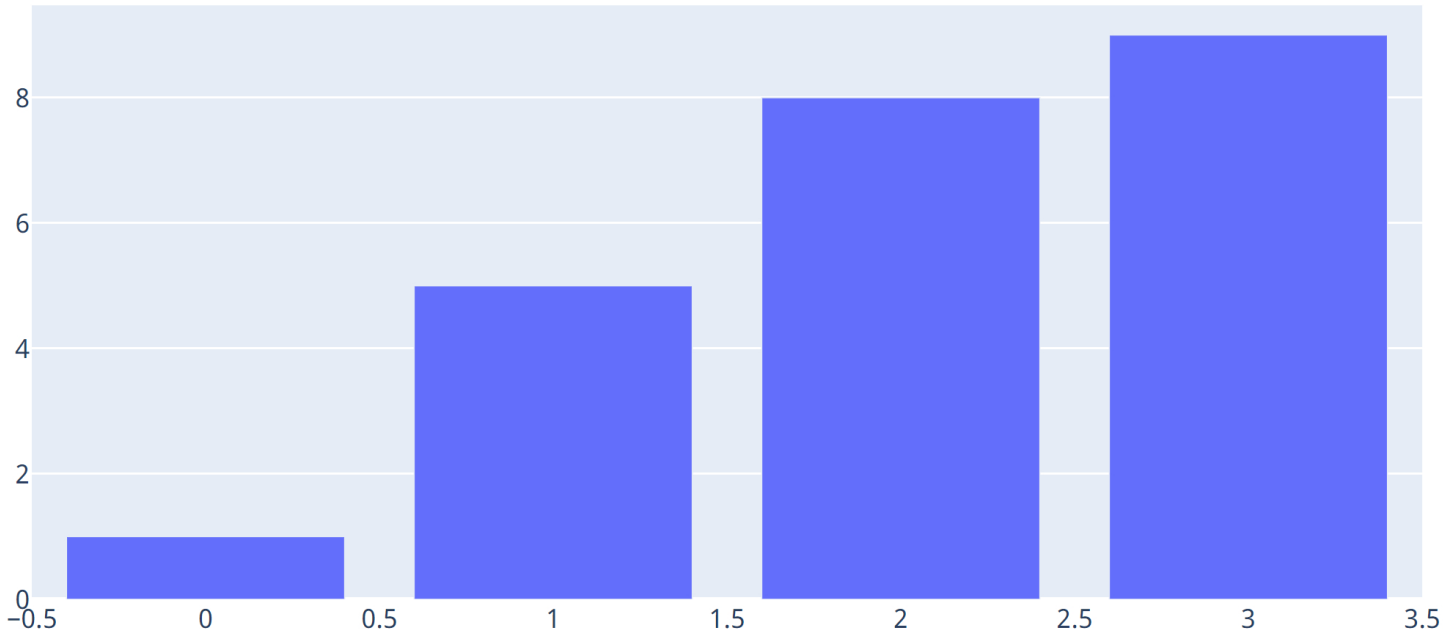
	sepal_length	sepal_width	petal_length	petal_width
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.057333	3.758000	1.199333
std	0.828066	0.435866	1.765298	0.762238
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

Writing Altair Chart

ⓘ May not work everywhere, needs javascript

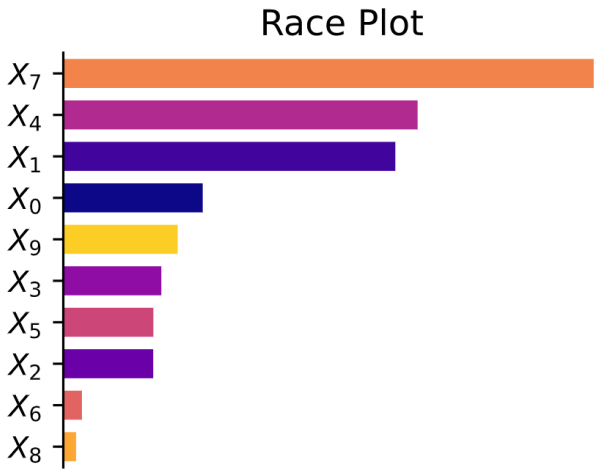


Writing Plotly Figure



Interactive Apps on Slide

Use `ipywidgets` , `bqplot` , `ipyvolume` , `plotly Figurewidget` etc. to show live apps like this!



Click me to update race plot

Check out this app

Python

```python
import ipywidgets as ipw
import numpy as np, matplotlib.pyplot as plt
grid, [(plot,button, _), code] = prs.iwrite([
        '## Plot will be here! Click button below to activate it!',
        ipw.Button(description='Click me to update race plot',layout=ipw.Layout(width='max-content')),
        "[Check out this app](https://massgh.github.io/pivotpy/Widgets.html#VasprunApp)"],src.focus_lines([4,5,6,7,*range(24,30)]))

def update_plot():
    x = np.linspace(0,0.9,10)
    y = np.random.random((10,))
    _sort = np.argsort(y)

    fig,ax = plt.subplots(figsize=(3.4,2.6))
    ax.barh(x,y[_sort],height=0.07,color=plt.cm.get_cmap('plasma')(x[_sort]))

    for s in ['right','top','bottom']:
```

Displaying image from url from somewhere in Kashmir (کشمیر)

$\LaTeX$ in Slides

Use `$ $` or `$$ $$` to display latex in Markdown, or embed images of equations

$$\int_0^1 \frac{1}{1-x^2} dx$$

Built-in CSS styles')

Use any or combinations of these styles in className argument of writing functions:

className = 'Center'	------Text------
className = 'Left'	Text----------
className = 'Right'	----------Text
className = 'RTL'	------ اردو عربی
className = 'Info'	Blue Text
className = 'Warning'	Orange Text
className = 'Success'	Green Text
className = 'Error'	Red Text
className = 'Note'	Text with info icon
className = 'slides-only'	Text will not appear in exported html with `build_report`
className = 'report-only'	Text will not appear on slides. Useful to fill content in report.

سارے جہاں میں دھوم ہماری زباں کی ہے۔

markdown.md

```
# IPyslides From Markdown
<center> Author: Abdul Saboor
```python run
import textwrap, time
from io import StringIO
from IPython import get_ipython
from ipyslides.utils import textbox
from ipyslides.writers import write, iwrite
from ipyslides.formatter import libraries, __reprs__
from ipyslides._base.intro import how_to_slide
import ipyslides.parsers as prs
prs.write(how_to_slide)
repr_methods = ', '.join([f'`repr_{rep}_` for rep in __reprs__])
custom_methods = ', '.join([f"{lib['name']}.{lib['obj']}" for lib in libraries])
```

Slide 1 {.Success}
```python run source
```
```