



Plotly FigureWidget Overview in Python

Introduction to the new Plotly FigureWidget

Figure

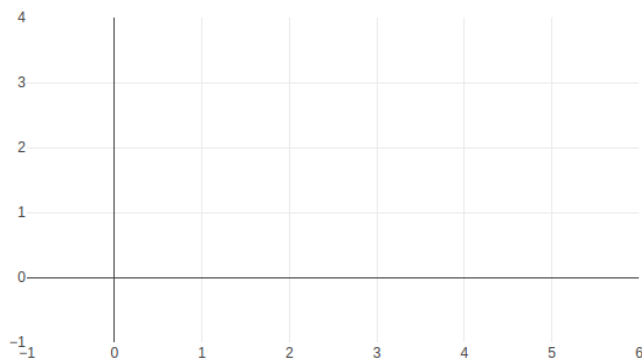
Plotly Studio: Transform any dataset into an interactive data application in minutes with AI. [Sign up for early access now.](https://plotly.com/studio/?utm_medium=graphing-libraries&utm_campaign=studio_early_access&utm_content=sidebar) (https://plotly.com/studio/?utm_medium=graphing-libraries&utm_campaign=studio_early_access&utm_content=sidebar)

Create a Simple FigureWidget

Create an empty FigureWidget and then view it.

```
import plotly.graph_objects as go

f = go.FigureWidget()
f
```

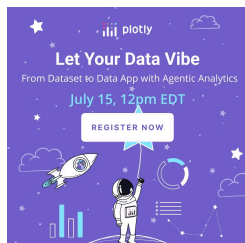


Add traces or update the layout and then watch the output above update in real time.

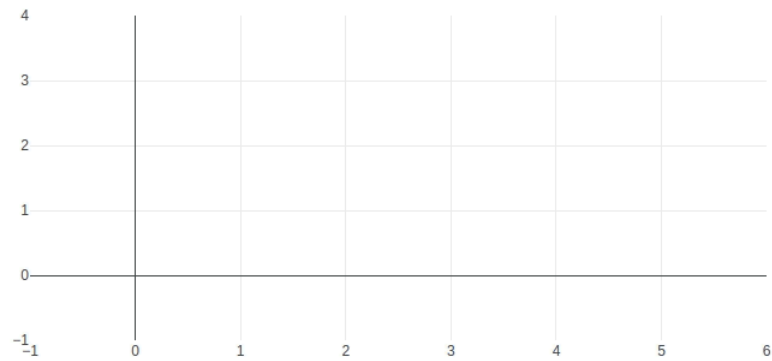
```
f.add_scatter(y=[2, 1, 4, 3]);
```

```
f.add_bar(y=[1, 4, 3, 2]);
```

```
f.layout.title = 'Hello FigureWidget'
```



Figure



Update the Data and the Layout

```
# update scatter data
scatter = f.data[0]
scatter.y = [3, 1, 4, 3]
```

```
# update bar data
bar = f.data[1]
bar.y = [5, 3, 2, 8]
```

```
f.layout.title.text = 'This is a new title'
```

Construct a FigureWidget from a Figure graph object

A standard Figure object can be passed to the FigureWidget constructor.

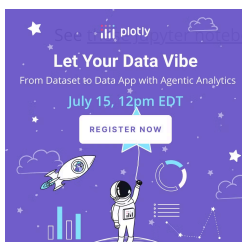
```
import plotly.graph_objects as go

trace = go.Heatmap(z=[[1, 20, 30, 50, 1], [20, 1, 60, 80, 30], [30, 60, 1, -10, 20]],
                  x=['Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday'],
                  y=['Morning', 'Afternoon', 'Evening'])
data=[trace]
layout = go.Layout(title=dict(text='Activity Heatmap'))

figure = go.Figure(data=data, layout=layout)

f2 = go.FigureWidget(figure)
f2
```

Reference



books (https://github.com/jonmmease/plotly_ipywidget_notebooks) for even more FigureWidget examples.

```
help(go.FigureWidget)
```

Help on class FigureWidget in module plotly.graph_objs._figurewidget:

```
class FigureWidget(plotly.basewidget.BaseFigureWidget)
| FigureWidget(data=None, layout=None, frames=None, skip_invalid=False, **kwargs)
|
| Method resolution order:
| FigureWidget
| plotly.basewidget.BaseFigureWidget
| plotly.basedatatypes.BaseFigure
| anywidget.widget.AnyWidget
| ipywidgets.widgets.domwidget.DOMWidget
| ipywidgets.widgets.widget.Widget
| inwidgets.widgets.widget.NoOpWidgetHasTraits
```

What About Dash?

[Dash \(https://dash.plot.ly/\)](https://dash.plot.ly/) is an open-source framework for building analytical applications, with no Javascript required, and it is tightly integrated with the Plotly graphing library.

Learn about how to install Dash at <https://dash.plot.ly/installation> (https://dash.plot.ly/installation).

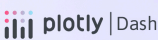
Everywhere in this page that you see fig.show(), you can display the same figure in a Dash application by passing it to the figure argument of the [Graph component](https://dash.plot.ly/dash-core-components/graph) (https://dash.plot.ly/dash-core-components/graph) from the built-in dash_core_components package like this:

```
import plotly.graph_objects as go # or plotly.express as px
fig = go.Figure() # or any Plotly Express function e.g. px.bar(...)
# fig.add_trace( ... )
# fig.update_layout( ... )

from dash import Dash, dcc, html

app = Dash()
app.layout = html.Div([
    dcc.Graph(figure=fig)
])

app.run(debug=True, use_reloader=False) # Turn off reloader if inside Jupyter
```



Dash your way to interactive web apps.

No JavaScript required!

GET STARTED NOW


My First App with Data, Graph, and Controls

pop

lifeExp

gdpPerCap

country	pop	continent	lifeExp	gdpPerCap
Afghanistan	31889923	Asia	43.828	974.5883384
Albania	3600523	Europe	76.423	5937.029525999999
Algeria	3333226	Africa	72.381	6223.367655
Angola	12428476	Africa	42.731	4707.231267
Argentina	6000927	Americas	75.32	12779.37964
Australia	20434176	Oceania	81.235	34435.367439999995
Austria	8199783	Europe	79.829	36126.4927
Bahrain	708073	Asia	75.635	29796.04834
Bangladesh	150448339	Asia	64.062	1701.253792
Belgium	10902226	Europe	79.641	33662.46588
Benin	8078316	Africa	56.728	1441.284873
Bolivia	913932	Americas	65.554	3822.337884

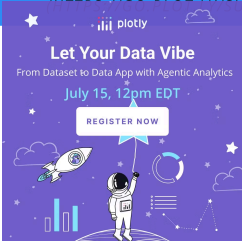


(https://dash.plotly.com/tutorial?utm_medium=graphing_libraries&utm_content=python_footer)

JOIN OUR MAILING LIST

Sign up to stay in the loop with all things Plotly — from Dash Club to product updates, webinars, and more!

SUBSCRIBE (DESCRIPTION)



Products

Dash (https://plotly.com/dash/)

Consulting and Training (https://plotly.com/consulting-and-oem/)

Pricing

Enterprise Pricing (https://plotly.com/get-pricing/)

About Us

Careers (<https://plotly.com/careers>)
Resources (<https://plotly.com/resources/>)
Blog (<https://medium.com/@plotlygraphs>)

Support

Community Support (<https://community.plot.ly/>)
Documentation (<https://plotly.com/graphing-libraries>)

Copyright © 2025 Plotly. All rights reserved.

[Terms of Service \(https://community.plotly.com/tos\)](https://community.plotly.com/tos) [Privacy Policy \(https://plotly.com/privacy/\)](https://plotly.com/privacy/)

ure

