



Python (/python) > Figure Reference (/python/reference/index/) > layout.updatemenus

Python Figure Reference: layout.updatemenus

updatemenus

Code: `fig.update_layout(updatemenus=list(...))`

Type: list of dict where each dict has one or more of the keys listed below.

active

Parent: `layout.updatemenus[]`

Type: integer greater than or equal to -1

Default: 0

Determines which button (by index starting from 0) is considered active.

bgcolor

Parent: `layout.updatemenus[]`

Type: color

Sets the background color of the update menu buttons.

bordercolor

Parent: `layout.updatemenus[]`

Type: color

Default: "#BEC8D9"

Sets the color of the border enclosing the update menu.

borderwidth

Parent: `layout.updatemenus[]`

Type: number greater than or equal to 0

Default: 1

Sets the width (in px) of the border enclosing the update menu.

buttons

Parent: `layout.updatemenus[]`

Type: list of dict where each dict has one or more of the keys listed below.

args

Parent: `layout.updatemenus[].buttons[]`

Type: list

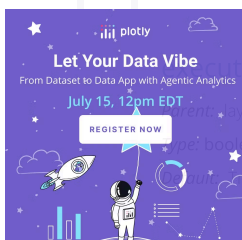
Sets the arguments values to be passed to the Plotly method set in `method` on click.

args2

Parent: `layout.updatemenus[].buttons[]`

Type: list

Sets a 2nd set of `args`, these arguments values are passed to the Plotly method set in `method` when clicking this button while in the active state. Use this to create toggle buttons.



When True, the API method is executed. When False, all other behaviors are the same and command execution is skipped. This may be useful when hooking into, for example, the `plotly_buttonclicked` method and executing the API command manually without losing the benefit of the update menu automatically binding to the state of the plot through the specification of `method` and `args`.

label

Parent: `layout.updatemenus[].buttons[]`

Type: string

Default: ""

Sets the text label to appear on the button.

method

Parent: `layout.updatemenus[].buttons[]`

Type: enumerated , one of ("restyle" | "relayout" | "animate" | "update" | "skip")

Default: "restyle"

Sets the Plotly method to be called on click. If the `'skip'` method is used, the API update menu will function as normal but will perform no API calls and will not bind automatically to state updates. This may be used to create a component interface and attach to update menu events manually via JavaScript.

name

Parent: `layout.updatemenus[].buttons[]`

Type: string

When used in a template, named items are created in the output figure in addition to any items the figure already has in this array. You can modify these items in the output figure by making your own item with `'templateitemname'` matching this `'name'` alongside your modifications (including `'visible: False'` or `'enabled: False'` to hide it). Has no effect outside of a template.

templateitemname

Parent: `layout.updatemenus[].buttons[]`

Type: string

Used to refer to a named item in this array in the template. Named items from the template will be created even without a matching item in the input figure, but you can modify one by making an item with `'templateitemname'` matching its `'name'`, alongside your modifications (including `'visible: False'` or `'enabled: False'` to hide it). If there is no template or no matching item, this item will be hidden unless you explicitly show it with `'visible: True'`.

visible

Parent: `layout.updatemenus[].buttons[]`

Type: boolean

Determines whether or not this button is visible.

direction

Parent: `layout.updatemenus[]`

Type: enumerated , one of ("left" | "right" | "up" | "down")

Default: "down"

Determines the direction in which the buttons are laid out, whether in a dropdown menu or a row/column of buttons. For `'left'` and `'up'`, the buttons will still appear in left-to-right or top-to-bottom order respectively.

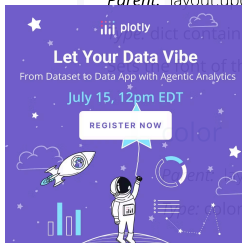
font

Parent: `layout.updatemenus[]`

Type: string, one or more of the keys listed below.

Default: "label", the update menu button text.

Parent: `layout.updatemenus[].font`



family

Parent: `layout.update_menus().font`

Type: string

HTML font family - the typeface that will be applied by the web browser. The web browser will only be able to apply a font if it is available on the system which it operates. Provide multiple font families, separated by commas, to indicate the preference in which to apply fonts if they aren't available on the system. The Chart Studio Cloud (at <https://chart-studio.plotly.com> or on-premise) generates images on a server, where only a select number of fonts are installed and supported. These include "Arial", "Balto", "Courier New", "Droid Sans", "Droid Serif", "Droid Sans Mono", "Gravitas One", "Old Standard TT", "Open Sans", "Overpass", "PT Sans Narrow", "Raleway", "Times New Roman".

lineposition

Parent: `layout.update_menus().font`

Type: flaglist string. Any combination of "under", "over", "through" joined with a "+" OR "none".

Examples: "under", "over", "under+over", "under+over+through", "none"

Default: "none"

Sets the kind of decoration line(s) with text, such as an "under", "over" or "through" as well as combinations e.g.

"under+over", etc.

shadow

Parent: `layout.update_menus().font`

Type: string

Default: "none"

Sets the shape and color of the shadow behind text. "auto" places minimal shadow and applies contrast text font color.

See <https://developer.mozilla.org/en-US/docs/Web/CSS/text-shadow> for additional options.

size

Parent: `layout.update_menus().font`

Type: number greater than or equal to 1

style

Parent: `layout.update_menus().font`

Type: enumerated, one of ("normal" | "italic")

Default: "normal"

Sets whether a font should be styled with a normal or italic face from its family.

textcase

Parent: `layout.update_menus().font`

Type: enumerated, one of ("normal" | "word caps" | "upper" | "lower")

Default: "normal"

Sets capitalization of text. It can be used to make text appear in all-uppercase or all-lowercase, or with each word capitalized.

variant

Parent: `layout.update_menus().font`

Type: enumerated, one of ("normal" | "small-caps" | "all-small-caps" | "all-petite-caps" | "petite-caps" | "unicase")

Default: "normal"

Sets the variant of the font.

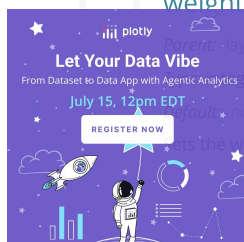
weight

Parent: `layout.update_menus().font`

Type: number between or equal to 1 and 1000

Default: "normal"

Sets the weight (or boldness) of the font.



name

Parent: layout.updatemenus[]

Type: string

When used in a template, named items are created in the output figure in addition to any items the figure already has in this array. You can modify these items in the output figure by making your own item with `templateitemname` matching this `name` alongside your modifications (including `visible: False` or `enabled: False` to hide it). Has no effect outside of a template.

pad

Parent: layout.updatemenus[]

Type: dict containing one or more of the keys listed below.

Sets the padding around the buttons or dropdown menu.

b

Parent: layout.updatemenus[].pad

Type: number

Default: 0

The amount of padding (in px) along the bottom of the component.

l

Parent: layout.updatemenus[].pad

Type: number

Default: 0

The amount of padding (in px) on the left side of the component.

r

Parent: layout.updatemenus[].pad

Type: number

Default: 0

The amount of padding (in px) on the right side of the component.

t

Parent: layout.updatemenus[].pad

Type: number

Default: 0

The amount of padding (in px) along the top of the component.

showactive

Parent: layout.updatemenus[]

Type: boolean

Default: True

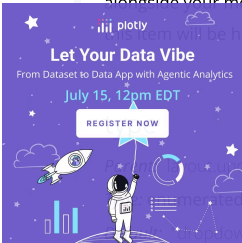
Highlights active dropdown item or active button if True.

templateitemname

Parent: layout.updatemenus[]

Type: string

Used to refer to a named item in this array in the template. Named items from the template will be created even without a matching item in the input figure, but you can modify one by making an item with `templateitemname` matching its `name`, alongside your modifications (including `visible: False` or `enabled: False` to hide it). If there is no template or no matching item, the item will be hidden unless you explicitly show it with `visible: True`.



Parent: layout.updatemenus[]

Type: one of ("dropdown" | "buttons")

Determines whether the buttons are accessible via a dropdown menu or whether the buttons are stacked horizontally or vertically

visible

Parent: layout.update_menus[]

Type: boolean

Determines whether or not the update menu is visible.

x

Parent: layout.update_menus[]

Type: number between or equal to -2 and 3

Default: -0.05

Sets the x position (in normalized coordinates) of the update menu.

xanchor

Parent: layout.update_menus[]

Type: enumerated , one of ("auto" | "left" | "center" | "right")

Default: "right"

Sets the update menu's horizontal position anchor. This anchor binds the `x` position to the "left", "center" or "right" of the range selector.

y

Parent: layout.update_menus[]

Type: number between or equal to -2 and 3

Default: 1

Sets the y position (in normalized coordinates) of the update menu.

yanchor

Parent: layout.update_menus[]

Type: enumerated , one of ("auto" | "top" | "middle" | "bottom")

Default: "top"

Sets the update menu's vertical position anchor This anchor binds the `y` position to the "top", "middle" or "bottom" of the range selector.

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
(<https://go.plot.ly/subscriptions>)

About Us

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

Products

Dash (<https://plotly.com/dash/>)
Consulting and Training
(<https://plotly.com/consulting-and-oem/>)

Pricing

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

Support

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

