Documentation

search

Search

- *rocket_launch*

  Get started

  - Installation
    add
  - Fundamentals
    add
  - First steps
    add
- *code*

  Develop

  - Concepts
    add
  - API reference
    remove
    - PAGE ELEMENTS

    - Write and magic
      add
    - Text elements
      add
    - Data elements
      add
    - Chart elements
      add
    - Input widgets
      remove
      - BUTTONS

      - st.button
      - st.download_button
      - st.form_submit_button*link*
      - st.link_button
      - st.page_link
      - SELECTIONS

      - st.checkbox
      - st.color_picker
      - st.feedback
      - st.multiselect
      - st.pills
      - st.radio
      - st.segmented_control
      - st.selectbox
      - st.select_slider
      - st.toggle
      - NUMERIC

- st.number_input
- st.slider
- DATE & TIME

- st.date_input
- st.time_input
- TEXT

- st.chat_input*link*
- st.text_area
- st.text_input
- MEDIA & FILES

- st.audio_input
- st.camera_input
- st.data_editor*link*
- st.file_uploader

- Media elements
  add
- Layouts and containers
  add
- Chat elements
  add
- Status elements
  add
- Third-party components*open_in_new*
- APPLICATION LOGIC

- Navigation and pages
  add
- Execution flow
  add
- Caching and state
  add
- Connections and secrets
  add
- Custom components
  add
- Utilities
  add
- Configuration
  add
- TOOLS

- App testing
  add
- Command line
  add

- Tutorials
  add
- Quick reference
  add

- *web_asset*

  Deploy

  - Concepts
    add

st.select_slider

Streamlit Version Version 1.41.0

Display a slider widget to select items from a list.

This also allows you to render a range slider by passing a two-element tuple or list as the `value`.

The difference between `st.select_slider` and `st.slider` is that `select_slider` accepts any datatype and takes an iterable set of options, while `st.slider` only accepts numerical or date/time data and takes a range as input.

Function signature[source]

st.select_slider(label, options=(), value=None, format_func=special_internal_function, key=None, help=None, on_change=None, args=None, kwargs=None, *, disabled=False, label_visibility="visible")

Parameters

label (str)	A short label explaining to the user what this slider is for. The label can optionally contain GitHub-flavored Markdown of the following types: Bold, Italics, Strikethroughs, Inline Code, Links, and Images. Images display like icons, with a max height equal to the font height. Unsupported Markdown elements are unwrapped so only their children (text contents) render. Display unsupported elements as literal characters by backslash-escaping them. E.g., `"1\. Not an ordered list"`. See the `body` parameter of `st.markdown` for additional, supported Markdown directives. For accessibility reasons, you should never set an empty label, but you can hide it with `label_visibility` if needed. In the future, we may disallow empty labels by raising an exception.

Returns

(any value or tuple of any value)	The current value of the slider widget. The return type will match the data type of the value parameter.

Function signature[source]

st.select_slider(label, options=(), value=None, format_func=special_internal_function, key=None, help=None, on_change=None, args=None, kwargs=None, *, disabled=False, label_visibility="visible")

options (Iterable)	Labels for the select options in an `Iterable`. This can be a `list`, `set`, or anything supported by `st.dataframe`. If `options` is dataframe-like, the first column will be used. Each label will be cast to `str` internally by default.
value (a supported type or a tuple/list of supported types or None)	The value of the slider when it first renders. If a tuple/list of two values is passed here, then a range slider with those lower and upper bounds is rendered. For example, if set to *(1, 10)* the slider will have a selectable range between 1 and 10. Defaults to first option.
format_func (function)	Function to modify the display of the labels from the options. argument. It receives the option as an argument and its output will be cast to str.
key (str or int)	An optional string or integer to use as the unique key for the widget. If this is omitted, a key will be generated for the widget based on its content. No two widgets may have the same key.
help (str)	An optional tooltip that gets displayed next to the widget label. Streamlit only displays the tooltip when `label_visibility="visible"`.
on_change (callable)	An optional callback invoked when this select_slider's value changes.
args (tuple)	An optional tuple of args to pass to the callback.
kwargs (dict)	An optional dict of kwargs to pass to the callback.
disabled (bool)	An optional boolean that disables the select slider if set to `True`. The default is `False`.
label_visibility ("visible", "hidden", or "collapsed")	The visibility of the label. The default is `"visible"`. If this is `"hidden"`, Streamlit displays an empty spacer instead of the label, which can help keep the widget alligned with other widgets. If this is `"collapsed"`, Streamlit displays no label or spacer.
Returns	
(any value or tuple of any value)	The current value of the slider widget. The return type will match the data type of the value parameter.

Examples

```
import streamlit as st

color = st.select_slider(
    "Select a color of the rainbow",
    options=[
        "red",
        "orange",
```

```
        "yellow",
        "green",
        "blue",
        "indigo",
        "violet",
    ],
)
st.write("My favorite color is", color)
```

And here's an example of a range select slider:

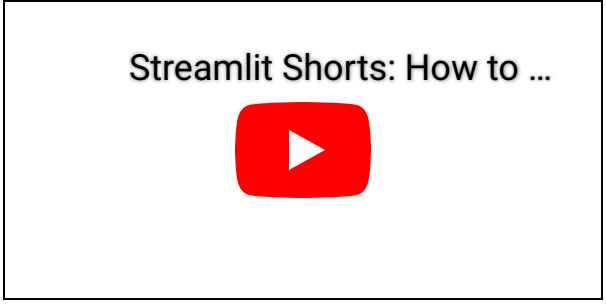
```
import streamlit as st

start_color, end_color = st.select_slider(
    "Select a range of color wavelength",
    options=[
        "red",
        "orange",
        "yellow",
        "green",
        "blue",
        "indigo",
        "violet",
    ],
    value=("red", "blue"),
)
st.write("You selected wavelengths between", start_color, "and", end_color)
```

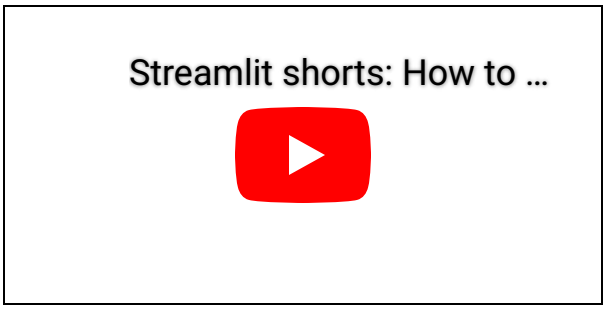
Built with Streamlit 🎈
Fullscreen *open_in_new*

Featured videos

Check out our video on how to use one of Streamlit's core functions, the select slider! 🎈



In the video below, we'll take it a step further and make a double-ended slider.



←Previous: st.selectboxNext: st.toggle→
forum

Still have questions?

Our forums are full of helpful information and Streamlit experts.

---

forum Ask AI