

[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](#)
add
 - [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](#)
remove
 - [st.testing.v1.AppTest](#)
 - [Testing element classes](#)
- [Command line](#)
add

- [Tutorials](#)
add
- [Quick reference](#)
add

- [web asset](#)

[Deploy](#)

- [Concepts](#)
add
- [Streamlit Community Cloud](#)
add
- [Snowflake](#)
- [Other platforms](#)
add

- [school](#)

[Knowledge base](#)

- [FAQ](#)
- [Installing dependencies](#)
- [Deployment issues](#)

- [Home/](#)
- [Develop/](#)
- [API reference/](#)
- [App testing/](#)
- [Testing element classes](#)

Testing element classes



st.testing.v1.element_tree.Block



The `Block` class has the same methods and attributes as `AppTest`. A `Block` instance represents a container of elements just as `AppTest` represents the entire app. For example, `Block.button` will produce a `WidgetList` of `Button` in the same manner as [AppTest.button](#).

`ChatMessage`, `Column`, and `Tab` all inherit from `Block`. For all container classes, parameters of the original element can be obtained as properties. For example, `ChatMessage.avatar` and `Tab.label`.

st.testing.v1.element_tree.Element



Streamlit Version Version 1.41.0

Element base class for testing.

This class's methods and attributes are universal for all elements implemented in testing. For example, `Caption`, `Code`, `Text`, and `Title` inherit from `Element`. All widget classes also inherit from `Element`, but have additional methods specific to each widget type. See the `AppTest` class for the full list of supported elements.

For all element classes, parameters of the original element can be obtained as properties. For example, `Button.label`, `Caption.help`, and `Toast.icon`.

Class description[\[source\]](#)

`st.testing.v1.element_tree.Element(proto, root)`

Methods

[run](#)(*, timeout=None) Run the `AppTest` script which contains the element.

Attributes

[value](#) The value or contents of the element.

`st.testing.v1.element_tree.Button`



Streamlit Version Version 1.41.0

A representation of `st.button` and `st.form_submit_button`.

Class description[\[source\]](#)

`st.testing.v1.element_tree.Button(proto, root)`

Methods

[click](#)() Set the value of the button to True.

[run](#)(*, timeout=None) Run the `AppTest` script which contains the element.

[set_value](#)(v) Set the value of the button.

Attributes

[value](#) The value of the button. (bool)

`st.testing.v1.element_tree.ChatInput`



A representation of `st.chat_input`.

Class description[\[source\]](#)

`st.testing.v1.element_tree.ChatInput(proto, root)`

Methods

[run](#)(* , timeout=None) Run the `AppTest` script which contains the element.

[set_value](#)(v) Set the value of the widget.

Attributes

[value](#) The value of the widget. (str)

st.testing.v1.element_tree.Checkbox



A representation of `st.checkbox`.

Class description[\[source\]](#)

`st.testing.v1.element_tree.Checkbox(proto, root)`

Methods

[check](#)() Set the value of the widget to True.

[run](#)(* , timeout=None) Run the `AppTest` script which contains the element.

[set_value](#)(v) Set the value of the widget.

[unchecked](#)() Set the value of the widget to False.

Attributes

[value](#) The value of the widget. (bool)

st.testing.v1.element_tree.ColorPicker



A representation of `st.color_picker`.

Class description[\[source\]](#)

st.testing.v1.element_tree.ColorPicker(proto, root)

Methods

[pick](#)(v) Set the value of the widget as a hex string. May omit the "#" prefix.

[run](#)(* , timeout=None) Run the `AppTest` script which contains the element.

[set_value](#)(v) Set the value of the widget as a hex string.

Attributes

[value](#) The currently selected value as a hex string. (str)

st.testing.v1.element_tree.DateInput



Streamlit Version

A representation of `st.date_input`.

Class description[\[source\]](#)

st.testing.v1.element_tree.DateInput(proto, root)

Methods

[run](#)(* , timeout=None) Run the `AppTest` script which contains the element.

[set_value](#)(v) Set the value of the widget.

Attributes

[value](#) The value of the widget. (date or Tuple of date)

st.testing.v1.element_tree.Multiselect



Streamlit Version

A representation of `st.multiselect`.

Class description[\[source\]](#)

st.testing.v1.element_tree.Multiselect(proto, root)

st.testing.v1.element_tree.Multiselect(proto, root)

Methods

run (*, timeout=None)	Run the <code>AppTest</code> script which contains the element.
select (v)	Add a selection to the widget. Do nothing if the value is already selected. If testing a multiselect widget with repeated options, use <code>set_value</code> instead.
set_value (v)	Set the value of the multiselect widget. (list)
unselect (v)	Remove a selection from the widget. Do nothing if the value is not already selected. If a value is selected multiple times, the first instance is removed.

Attributes

format_func	The widget's formatting function for displaying options. (callable)
indices	The indices of the currently selected values from the options. (list)
value	The currently selected values from the options. (list)

st.testing.v1.element_tree.NumberInput



Streamlit Version

A representation of `st.number_input`.

Class description[\[source\]](#)

st.testing.v1.element_tree.NumberInput(proto, root)

Class description[\[source\]](#)

st.testing.v1.element_tree.NumberInput(proto, root)

Methods

[decrement\(\)](#) Decrement the `st.number_input` widget as if the user clicked "-".

[increment\(\)](#) Increment the `st.number_input` widget as if the user clicked "+".

[run](#)(*, timeout=None) Run the `AppTest` script which contains the element.

[set_value](#)(v) Set the value of the `st.number_input` widget.

Attributes

[value](#) Get the current value of the `st.number_input` widget.

st.testing.v1.element_tree.Radio



Streamlit Version

A representation of `st.radio`.

Class description[\[source\]](#)

st.testing.v1.element_tree.Radio(proto, root)

Methods

[run](#)(*, timeout=None) Run the `AppTest` script which contains the element.

[set_value](#)(v) Set the selection by value.

Attributes

[format_func](#) The widget's formatting function for displaying options. (callable)

[index](#) The index of the current selection. (int)

[value](#) The currently selected value from the options. (Any)

st.testing.v1.element_tree.SelectSlider



A representation of `st.select_slider`.

Class description[\[source\]](#)

`st.testing.v1.element_tree.SelectSlider(proto, root)`

Methods

[run](#)(* , timeout=None) Run the `AppTest` script which contains the element.

[set_range](#)(lower, upper) Set the ranged selection by values.

[set_value](#)(v) Set the (single) selection by value.

Attributes

[format_func](#) The widget's formatting function for displaying options. (callable)

[value](#) The currently selected value or range. (Any or Sequence of Any)

st.testing.v1.element_tree.Selectbox



A representation of `st.selectbox`.

Class description[\[source\]](#)

`st.testing.v1.element_tree.Selectbox(proto, root)`

st.testing.v1.element_tree.Selectbox(proto, root)

Methods

[run](#)(*, timeout=None) Run the `AppTest` script which contains the element.

[select](#)(v) Set the selection by value.

[select_index](#)(index) Set the selection by index.

[set_value](#)(v) Set the selection by value.

Attributes

[format_func](#) The widget's formatting function for displaying options. (callable)

[index](#) The index of the current selection. (int)

[value](#) The currently selected value from the options. (Any)

st.testing.v1.element_tree.Slider



Streamlit Version

A representation of `st.slider`.

st.testing.v1.element_tree.Slider(proto, root)

Methods

[run](#)(*, timeout=None) Run the `AppTest` script which contains the element.

[set_range](#)(lower, upper) Set the ranged value of the slider.

[set_value](#)(v) Set the (single) value of the slider.

Attributes

[value](#) The currently selected value or range. (Any or Sequence of Any)

st.testing.v1.element_tree.TextArea



Streamlit Version Version 1.41.0

A representation of `st.text_area`.

Class description[\[source\]](#)

`st.testing.v1.element_tree.TextArea(proto, root)`

Methods

[input](#)(v) Set the value of the widget only if the value does not exceed the maximum allowed characters.

[run](#)(* , timeout=None) Run the `AppTest` script which contains the element.

[set_value](#)(v) Set the value of the widget.

Attributes

[value](#) The current value of the widget. (str)

st.testing.v1.element_tree.TextInput



Streamlit Version Version 1.41.0

A representation of `st.text_input`.

Class description[\[source\]](#)

`st.testing.v1.element_tree.TextInput(proto, root)`

Methods

[input](#)(v) Set the value of the widget only if the value does not exceed the maximum allowed characters.

[run](#)(* , timeout=None) Run the `AppTest` script which contains the element.

[set_value](#)(v) Set the value of the widget.

Attributes

[value](#) The current value of the widget. (str)

st.testing.v1.element_tree.TimeInput



A representation of `st.time_input`.

Class description[\[source\]](#)

`st.testing.v1.element_tree.TimeInput(proto, root)`

Methods

[decrement\(\)](#) Select the previous available time.

[increment\(\)](#) Select the next available time.

[run](#)(*, timeout=None) Run the `AppTest` script which contains the element.

[set_value](#)(v) Set the value of the widget.

Attributes

[value](#) The current value of the widget. (time)

st.testing.v1.element_tree.Toggle



A representation of `st.toggle`.

Class description[\[source\]](#)

`st.testing.v1.element_tree.Toggle(proto, root)`

Methods

[run](#)(*, timeout=None) Run the `AppTest` script which contains the element.

[set_value](#)(v) Set the value of the widget.

Attributes

[value](#) The current value of the widget. (bool)

[←Previous: st.testing.v1.AppTestNext: Command line→](#)

[forum](#)

Still have questions?

Our [forums](#) are full of helpful information and Streamlit experts.

