

[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](#)
remove
 - [st.columns](#)
 - [st.container](#)
 - [st.dialoglink](#)
 - [st.empty](#)
 - [st.expander](#)
 - [st.formlink](#)
 - [st.popover](#)
 - [st.sidebar](#)
 - [st.tabs](#)
 - [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
- [Streamlit Community Cloud](#)
add
- [Snowflake](#)
- [Other platforms](#)
add
- [school](#)

[Knowledge base](#)

- [FAQ](#)
- [Installing dependencies](#)
- [Deployment issues](#)
- [Home/](#)
- [Develop/](#)
- [API reference/](#)
- [Layouts and containers/](#)
- [st.empty](#)

st.empty



Streamlit Version Version 1.41.0 ▼

Insert a single-element container.

Inserts a container into your app that can be used to hold a single element. This allows you to, for example, remove elements at any point, or replace several elements at once (using a child multi-element container).

To insert/replace/clear an element on the returned container, you can use with notation or just call methods directly on the returned object. See examples below.

Function signature[\[source\]](#)

st.empty()

Examples

Inside a with `st.empty():` block, each displayed element will replace the previous one.

```
import streamlit as st
import time

with st.empty():
    for seconds in range(10):
        st.write(f"⌚ {seconds} seconds have passed")
        time.sleep(1)
    st.write(":material/check: 10 seconds over!")
st.button("Rerun")
```



[Built with Streamlit](#) 📌

[Fullscreen open in new](#)

You can use an `st.empty` to replace multiple elements in succession. Use `st.container` inside `st.empty` to display (and later replace) a group of elements.

```
import streamlit as st
import time

st.button("Start over")

placeholder = st.empty()
placeholder.markdown("Hello")
time.sleep(1)

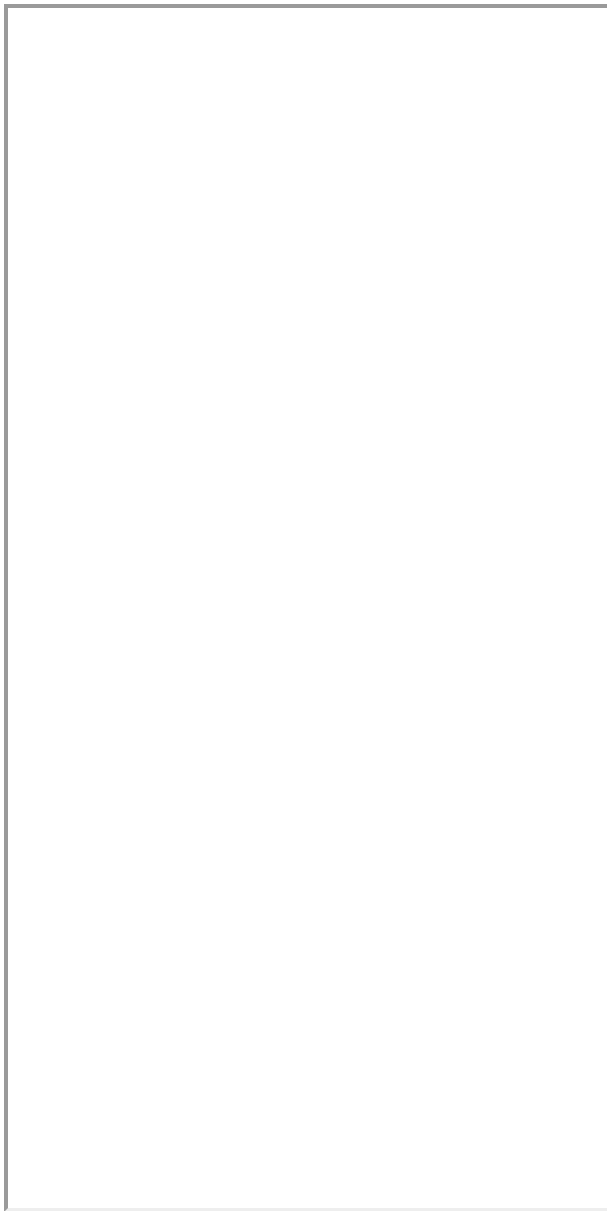
placeholder.progress(0, "Wait for it...")
time.sleep(1)
placeholder.progress(50, "Wait for it...")
time.sleep(1)
placeholder.progress(100, "Wait for it...")
time.sleep(1)

with placeholder.container():
    st.line_chart({"data": [1, 5, 2, 6]})
    time.sleep(1)
    st.markdown("3...")
    time.sleep(1)
    st.markdown("2...")
    time.sleep(1)
```

```
st.markdown("1...")
time.sleep(1)

placeholder.markdown("Poof!")
time.sleep(1)

placeholder.empty()
```



[Built with Streamlit](#) 

[Fullscreen open in new](#)

[←Previous: st.dialog](#)[Next: st.expander→](#)
forum

Still have questions?

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

[Home](#)[Contact Us](#)[Community](#)



© 2025 Snowflake Inc. [Cookie policy](#)

forum [Ask AI](#)