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
      remove
      - CALLOUTS

      - st.success
      - st.info
      - st.warning
      - st.error
      - st.exception
      - OTHER

      - st.progress
      - st.spinner
      - st.status

        - st.toast
        - st.balloons
        - st.snow
    - 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

st.toast

Streamlit Version Version 1.41.0

Display a short message, known as a notification "toast".

The toast appears in the app's bottom-right corner and disappears after four seconds.

Warning

`st.toast` is not compatible with Streamlit's caching and cannot be called within a cached function.

Function signature[source]

st.toast(body, *, icon=None)

Parameters

body (str)	The string to display as GitHub-flavored Markdown. Syntax information can be found at: https://github.github.com/gfm. See the `body` parameter of `st.markdown` for additional, supported Markdown directives.
icon (str, None)	An optional emoji or icon to display next to the alert. If `icon` is `None` (default), no icon is displayed. If `icon` is a string, the following options are valid: • A single-character emoji. For example, you can set `icon="🚨"` or `icon="🔥"`. Emoji short codes are not supported. • An icon from the Material Symbols library (rounded style) in the format `":material/icon_name:"` where "icon_name" is the name of the icon in snake case. For example, `icon=":material/thumb_up:"` will display the Thumb Up icon. Find additional icons in the Material Symbols font library.

Example

```
import streamlit as st

st.toast('Your edited image was saved!', icon='😍')
```

When multiple toasts are generated, they will stack. Hovering over a toast will stop it from disappearing. When hovering ends, the toast will disappear after four more seconds.

```
import streamlit as st import time if st.button('Three cheers'): st.toast('Hip!') time.sleep(.5)
st.toast('Hip!') time.sleep(.5) st.toast('Hooray!', icon='🎉')
```

Built with Streamlit 🎈
Fullscreen *open_in_new*

Toast messages can also be updated. Assign `st.toast(my_message)` to a variable and use the `.toast()` method to update it. Note: if a toast has already disappeared or been dismissed, the update will not be seen.

```
import streamlit as st import time def cook_breakfast(): msg = st.toast('Gathering ingredients...') time.sleep(1) msg.toast('Cooking...') time.sleep(1) msg.toast('Ready!', icon = "🥞") if st.button('Cook breakfast'): cook_breakfast()
```

Built with Streamlit 🎈
Fullscreen *open_in_new*

←Previous: st.statusNext: st.balloons→
forum

Still have questions?

Our forums are full of helpful information and Streamlit experts.

© 2025 Snowflake Inc. Cookie policy
forum Ask AI