

New to Gradio? Start here: **Getting Started**

See the **Release History**

← ClearButton

ColorPicker →

Code

gradio.Code(...)

Description

Creates a Code editor for entering, editing or viewing code.

Behavior

As input: passes a `str` of code into the function.

As output: expects the function to return a `str` of code or a single-element `tuple`: `(string_filepath,)`

Initialization

Parameter

Description

value

`str` | `tuple[str]` | `None`

default: `None`

Default value to show in the code editor. If callable, the function will be called whenever the app loads to set the initial value of the component.

language

`Literal[('python', 'markdown', 'json', 'html', 'css', 'javascript', 'typescript', 'yaml', 'dockerfile', 'shell', 'r')]` | `None`

default: `None`

The language to display the code as. Supported languages listed in `gr.Code.languages`.

every

`float` | `None`

default: `None`

If `value` is a callable, run the function 'every' number of seconds while the client connection is open. Has no effect otherwise. Queue must be enabled. The event can be accessed (e.g. to cancel it) via this component's `.load_event` attribute.

meter	Description
<div>lines</div> <div><i>int</i></div> <div>default: 5</div>	
<div>label</div> <div><i>str None</i></div> <div>default: None</div>	The label for this component. Appears above the component and is also used as the header if there are a table of examples for this component. If None and used in a <code>gr.Interface</code> , the label will be the name of the parameter this component is assigned to.
<div>interactive</div> <div><i>bool None</i></div> <div>default: None</div>	Whether user should be able to enter code or only view it.
<div>show_label</div> <div><i>bool None</i></div> <div>default: None</div>	if True, will display label.
<div>container</div> <div><i>bool</i></div> <div>default: True</div>	If True, will place the component in a container - providing some extra padding around the border.
<div>scale</div> <div><i>int None</i></div> <div>default: None</div>	relative width compared to adjacent Components in a Row. For example, if Component A has scale=2, and Component B has scale=1, A will be twice as wide as B. Should be an integer.
<div>min_width</div> <div><i>int</i></div> <div>default: 160</div>	minimum pixel width, will wrap if not sufficient screen space to satisfy this value. If a certain scale value results in this Component being narrower than min_width, the min_width parameter will be respected first.
<div>visible</div> <div><i>bool</i></div> <div>default: True</div>	If False, component will be hidden.

meter	Description
<div>elem_id</div> <div>str None</div> <div>default: None</div>	An optional string that is assigned as the id of this component in the HTML DOM. Can be used for targeting CSS styles.
<div>elem_classes</div> <div>list[str] str None</div> <div>default: None</div>	An optional list of strings that are assigned as the classes of this component in the HTML DOM. Can be used for targeting CSS styles.
<div>render</div> <div>bool</div> <div>default: True</div>	If False, component will not render be rendered in the Blocks context. Should be used if the intention is to assign event listeners now but render the component later.

Shortcuts

Class	Interface String Shortcut	Initialization
<div>gradio.Code</div>	"code"	Uses default values

Event Listeners

Description

Event listeners allow you to capture and respond to user interactions with the UI components you've defined in a Gradio Blocks app. When a user interacts with an element, such as changing a slider value or uploading an image, a function is called.

Supported Event Listeners

The Code component supports the following event listeners. Each event listener takes the same parameters, which are listed in the Event Arguments table below.

Listener	Description
<div>gradio.Code.languages(fn, ...)</div>	['python', 'markdown', 'json', 'html', 'css', 'javascript', 'typescript', 'yaml', 'dockerfile', 'shell', 'r', None]



Listener

Description

<code>gradio.Code.change(fn, ...)</code>	Triggered when the value of the Code changes either because of user input (e.g. a user types in a textbox) OR because of a function update (e.g. an image receives a value from the output of an event trigger). See <code>.input()</code> for a listener that is only triggered by user input.
<code>gradio.Code.input(fn, ...)</code>	This listener is triggered when the user changes the value of the Code.
<code>gradio.Code.focus(fn, ...)</code>	This listener is triggered when the Code is focused.
<code>gradio.Code.blur(fn, ...)</code>	This listener is triggered when the Code is unfocused/blurred.

Event Arguments

Parameter

Description

← ClearButton

ColorPicker →