API reference Classes Dialog

Dialog

Dialog objects are dispatched by page via the page.on("dialog") event.

An example of using `Dialog` class:

Sync Async

```
from playwright.sync_api import sync_playwright, Playwright

def handle_dialog(dialog):
    print(dialog.message)
    dialog.dismiss()

def run(playwright: Playwright):
    chromium = playwright.chromium
    browser = chromium.launch()
    page = browser.new_page()
    page.on("dialog", handle_dialog)
    page.evaluate("alert('1')")
    browser.close()

with sync_playwright() as playwright:
    run(playwright)
```

> **NOTE**
>
> Dialogs are dismissed automatically, unless there is a page.on("dialog") listener. When listener is present, it **must** either dialog.accept() or dialog.dismiss() the dialog - otherwise the page will freeze waiting for the dialog, and actions like click will never finish.

Methods

accept

Added in: v1.8

Returns when the dialog has been accepted.

Usage

```
dialog.accept()
dialog.accept(**kwargs)
```

Arguments

- `prompt_text` str *(optional)*

  A text to enter in prompt. Does not cause any effects if the dialog's `type` is not prompt. Optional.

dismiss

Added in: v1.8

Returns when the dialog has been dismissed.

Usage

```
dialog.dismiss()
```

Properties

default_value

Added in: v1.8

If dialog is prompt, returns default prompt value. Otherwise, returns empty string.

Usage

```
dialog.default_value
```

Returns

- str

message

Added in: v1.8

A message displayed in the dialog.

Usage

```
dialog.message
```

Returns

- str

page

Added in: v1.34

The page that initiated this dialog, if available.

Usage

```
dialog.page
```

Returns

- NoneType|Page

type

Added in: v1.8

Returns dialog's type, can be one of `alert`, `beforeunload`, `confirm` or `prompt`.

Usage

```
dialog.type
```

Returns

- str