API reference Classes FileChooser

FileChooser

FileChooser objects are dispatched by the page in the page.on("filechooser") event.

Sync Async

```
with page.expect_file_chooser() as fc_info:
    page.get_by_text("Upload file").click()
file_chooser = fc_info.value
file_chooser.set_files("myfile.pdf")
```

Methods

set_files

Added in: v1.8

Sets the value of the file input this chooser is associated with. If some of the `filePaths` are relative paths, then they are resolved relative to the current working directory. For empty array, clears the selected files.

Usage

```
file_chooser.set_files(files)
file_chooser.set_files(files, **kwargs)
```

Arguments

- `files` Union[str, pathlib.Path]|List[Union[str, pathlib.Path]]|Dict|List[Dict]
    - `name` str

      File name
    - `mimeType` str

    File type

  - `buffer` bytes

    File content

- `no_wait_after` bool *(optional)*

  Actions that initiate navigations are waiting for these navigations to happen and for pages to start loading. You can opt out of waiting via setting this flag. You would only need this option in the exceptional cases such as navigating to inaccessible pages. Defaults to `false`.

- `timeout` float *(optional)*

  Maximum time in milliseconds. Defaults to `30000` (30 seconds). Pass `0` to disable timeout. The default value can be changed by using the browser_context.set_default_timeout() or page.set_default_timeout() methods.

Properties

element

Added in: v1.8

Returns input element associated with this file chooser.

Usage

```
file_chooser.element
```

Returns

- ElementHandle

is_multiple

Added in: v1.8

Returns whether this file chooser accepts multiple files.

Usage

```
file_chooser.is_multiple()
```

Returns

- bool

page

Added in: v1.8

Returns page this file chooser belongs to.

Usage

```
file_chooser.page
```

Returns

- Page