

---

# filehandling

*Release 0.0.1*

**Aug 19, 2022**



---

## Contents

---

<b>1 File Handling</b>	<b>1</b>
<b>Python Module Index</b>	<b>5</b>
<b>Index</b>	<b>7</b>



# CHAPTER 1

---

## File Handling

---

```
class filehandling.BatchProcess (directory, extension=None, relative=False, smart_sort=None,  
                                reverse_sort=False)
```

Bases: object

BatchProcess is a generator that enables you to easily iterate through a selection of files in a directory.

**num\_files**

**Type** int The number of files in the selection

**current**

**Type** int The index of the file currently pointed at

**files**

**Type** list A list of strings of the filenames to be iterated over

**Returns** file

**Return type** str A filename

### Examples

directory is a path to a folder or expression for pattern matching. eg. /Documents/Example/a\*b?.txt This returns files beginning in a with a b as the penultimate letter and file extension .txt

```
for filename in BatchProcess(directory): print(filename)
```

```
filehandling.create_directory (initialdir='/', title='Create directory', parent=None)
```

Opens a filedialog to create a directory

**Parameters**

- **initialdir** (*str*) – The initial directory
- **title** (*str*) – Message box title
- **parent** (*widget*) – If using with tkinter gui provide the parent so focus is returned.

**Returns filename** – The path of the created directory

**Return type** str

`filehandling.create_filename(initialdir='/', title='Save File', filetypes=(('all files', '*.*'), ), append_time=False, parent=None)`

Opens a filedialog to save a file.

**Parameters**

- **initialdir** (*str*) – The initial directory
- **title** (*str*) – Message box title
- **filetypes** (*tuple*) – Sequence of (label, pattern) tuples. The same label may occur with several patterns. Use “\*” as the pattern to indicate all files.
- **parent** (*widget*) – If using with tkinter gui provide the parent so focus is returned.

**Returns filename** – The selected filename

**Return type** str

`filehandling.datetime_stamp(format_string='%Y%m%d_%H%M%S')`

Get string for current date and time

`filehandling.get_directory(initialdir='/', title='Select a directory', parent=None)`

Opens a filedialog to select a directory

**Parameters**

- **initialdir** (*str*) – The initial directory
- **title** (*str*) – Message box title
- **parent** (*widget*) – If using with tkinter gui provide the parent so focus is returned.

**Returns filename** – The selected directory

**Return type** str

`filehandling.get_directory_filenames(directory, reverse_sort=False, smart_sort=None, relative=False, extension=None)`

Returns all the files from a directory.

Can set the filetype using extension.

**Parameters**

- **directory** (*str*) – Filepath pointing to the directory with the final / Can use this with glob wildcards to use more complicated patterns.
- **reverse\_sort** (*bool*) – If true files returns in reverse alphabetical order
- **relative** (*bool*) – If True files will be returned without the directory
- **smart\_sort** (*function\_handle or None*) –
- **extension** (*str*) – Extension filetype to be used as filter.

**Returns files** – List of all the files that match the pattern.

**Return type** list

`filehandling.get_ext(filepath)`

Returns the extension from a filepath

`filehandling.get_filename(initialdir='/', title='Select File', filetypes=(('all files', '*.*'), ), parent=None)`

Opens a filedialog to return a filename to open

#### Parameters

- **initialdir** (*str*) – The initial directory
- **title** (*str*) – Message box title
- **filetypes** (*tuple*) – Sequence of (label, pattern) tuples. The same label may occur with several patterns. Use “\*” as the pattern to indicate all files.
- **parent** (*widget*) – If using with tkinter gui provide the parent so focus is returned.

**Returns filename** – The selected filename

**Return type** *str*

`filehandling.get_filenames()`

`filehandling.get_name(filepath)`

Returns the filename of a path without the directory

*/foo/bar/name.txt -> name.txt*

`filehandling.list_files(directory, reverse_sort=False, smart_sort=None, relative=False, extension=None)`

Returns all the files from a directory.

Can set the filetype using extension.

#### Parameters

- **directory** (*str*) – Filepath pointing to the directory with the final / Can use this with glob wildcards to use more complicated patterns.
- **reverse\_sort** (*bool*) – If true files returns in reverse alphabetical order
- **relative** (*bool*) – If True files will be returned without the directory
- **smart\_sort** (*function\_handle or None*) –
- **extension** (*str*) – Extension filetype to be used as filter.

**Returns files** – List of all the files that match the pattern.

**Return type** *list*

`filehandling.open_directory(initialdir='/', title='Select a directory', parent=None)`

Opens a filedialog to select a directory

#### Parameters

- **initialdir** (*str*) – The initial directory
- **title** (*str*) – Message box title
- **parent** (*widget*) – If using with tkinter gui provide the parent so focus is returned.

**Returns filename** – The selected directory

**Return type** *str*

`filehandling.open_filename(initialdir='/', title='Select File', filetypes=(('all files', '*.*'), ), parent=None)`

Opens a filedialog to return a filename to open

#### Parameters

- **initialdir** (*str*) – The initial directory
- **title** (*str*) – Message box title
- **filetypes** (*tuple*) – Sequence of (label, pattern) tuples. The same label may occur with several patterns. Use “\*” as the pattern to indicate all files.
- **parent** (*widget*) – If using with tkinter gui provide the parent so focus is returned.

**Returns filename** – The selected filename

**Return type** *str*

`filehandling.remove_ext (filepath)`

Returns the file without extension from a filepath

`filehandling.remove_file (filepath)`

Returns the top directory from a filepath

`filehandling.remove_path (filepath)`

Returns the name of the file from a filepath

`filehandling.save_filename (initialdir='/', title='Save File', filetypes=(('all files', '*.*'), ), append_time=False, parent=None)`

Opens a filedialog to save a file.

**Parameters**

- **initialdir** (*str*) – The initial directory
- **title** (*str*) – Message box title
- **filetypes** (*tuple*) – Sequence of (label, pattern) tuples. The same label may occur with several patterns. Use “\*” as the pattern to indicate all files.
- **parent** (*widget*) – If using with tkinter gui provide the parent so focus is returned.

**Returns filename** – The selected filename

**Return type** *str*

`filehandling.smart_number_sort (filenames)`



**f**

filehandling, 1



### B

`BatchProcess` (*class in filehandling*), 1

### C

`create_directory()` (*in module filehandling*), 1

`create_filename()` (*in module filehandling*), 2

`current` (*filehandling.BatchProcess attribute*), 1

### D

`datetime_stamp()` (*in module filehandling*), 2

### F

`filehandling` (*module*), 1

`files` (*filehandling.BatchProcess attribute*), 1

### G

`get_directory()` (*in module filehandling*), 2

`get_directory_filenames()` (*in module filehandling*), 2

`get_ext()` (*in module filehandling*), 2

`get_filename()` (*in module filehandling*), 2

`get_filenames()` (*in module filehandling*), 3

`get_name()` (*in module filehandling*), 3

### L

`list_files()` (*in module filehandling*), 3

### N

`num_files` (*filehandling.BatchProcess attribute*), 1

### O

`open_directory()` (*in module filehandling*), 3

`open_filename()` (*in module filehandling*), 3

### R

`remove_ext()` (*in module filehandling*), 4

`remove_file()` (*in module filehandling*), 4

`remove_path()` (*in module filehandling*), 4

### S

`save_filename()` (*in module filehandling*), 4

`smart_number_sort()` (*in module filehandling*), 4