

---

# SafeMyWork Documentation

*Release 0.5*

**Thykof**

**Mar 23, 2020**



---

## Contents

---

<b>1</b>	<b>How it works</b>	<b>3</b>
<b>2</b>	<b>Automatic saving</b>	<b>5</b>
<b>3</b>	<b>Synchronization</b>	<b>7</b>
<b>4</b>	<b>Safer: save files</b>	<b>9</b>
<b>5</b>	<b>Sync: synchronize folders</b>	<b>11</b>
5.1	A module for simple functions . . . . .	11
	<b>Python Module Index</b>	<b>13</b>
	<b>Index</b>	<b>15</b>



The current version is 0.5.

<https://github.com/Thykof/SafeMyWork>

SafeMyWork is intended for people who handle a lot of files which can be texts, images, songs... The aim is to **avoid losing** documents.



# CHAPTER 1

---

## How it works

---

SafeMyWork looks for your modified files and copies them to another directory. The second feature is the synchronization of two folders.





## CHAPTER 2

---

### Automatic saving

---

Run SafeMyWork and select your working folders. Then while you are working, every a certain amount of time, all files in this folder will be copied into a separate folder. It keeps your files safe if you forget to save your work or delete accidentally files.



## CHAPTER 3

---

### Synchronization

---

The application can synchronize two folders. After scanning the two folders selected by the user, a dialog message shows potential conflicts. A conflict occurs when a file has the same path and the same name in both folders. The user selects the file that she/he wants to keep. The size of the files and the last modification time are shown to help the choose. Then, a dialog shows what files would be lost (conflict resolution) and what files would be created (missing files in a folder).



## CHAPTER 4

---

Safer: save files

---



---

## Sync: synchronize folders

---

```
class safer.sync.Sync(local_path, ext_path)

    set_safe_doc(safe_doc)
    set_config(config)
    scan_dir(dirname)
    scan_compare(local_path=None, ext_path=None)
    sync(comparison=None)
    exec_sync(local_path=None, ext_path=None)
    compare(locals_files=None, ext_files=None)
    solve_conflicts()
    change_dst(path_type, num)
    add_in_local(filename)
    remove_in_local(filename)
    add_in_ext(filename)
    remove_in_ext(filename)
```

### 5.1 A module for simple functions

Define some functions.

```
safer.helpers.combine_list(list1, list2)
    Create a list with only elements in common of the two lists.
```

#### Parameters

- **list1** (*list*) – the first list
- **list2** (*list*) – the second list

**Returns** the created list

**Return type** `list`

`safer.helpers.path_without_root` (*path*)  
Return the *path* without the first directory.

e.g. “root/foo/bar” -> “foo/bar”

`safer.helpers.missing_item` (*list1*, *list2*)  
Return the list of directories in *list1* but not in *list2*.

`safer.helpers.split_path` (*path*)  
Return a list with all directory names in the given path.



**S**

`safer.helpers`, 11



## A

`add_in_ext()` (*safer.sync.Sync method*), 11  
`add_in_local()` (*safer.sync.Sync method*), 11

## C

`change_dst()` (*safer.sync.Sync method*), 11  
`combine_list()` (*in module safer.helpers*), 11  
`compare()` (*safer.sync.Sync method*), 11

## E

`exec_sync()` (*safer.sync.Sync method*), 11

## M

`missing_item()` (*in module safer.helpers*), 12

## P

`path_without_root()` (*in module safer.helpers*), 12

## R

`remove_in_ext()` (*safer.sync.Sync method*), 11  
`remove_in_local()` (*safer.sync.Sync method*), 11

## S

`safer.helpers` (*module*), 11  
`scan_compare()` (*safer.sync.Sync method*), 11  
`scan_dir()` (*safer.sync.Sync method*), 11  
`set_config()` (*safer.sync.Sync method*), 11  
`set_safe_doc()` (*safer.sync.Sync method*), 11  
`solve_conflicts()` (*safer.sync.Sync method*), 11  
`split_path()` (*in module safer.helpers*), 12  
`Sync` (*class in safer.sync*), 11  
`sync()` (*safer.sync.Sync method*), 11