Environment

JAVA_HOME	/Library/Java/JavaVirtualMachines/jdk1.8.0_121.jdk/Contents/Home
Packages	{'pytest': '3.0.7', 'py': '1.4.33', 'pluggy': '0.4.0'}
Platform	Darwin-16.5.0-x86_64-i386-64bit
Plugins	{'metadata': '1.4.0', 'html': '1.14.2'}
Python	3.6.0

Summary

245 tests ran in 3.50 seconds.

(Un)check the boxes to filter the results.

✓ 126 passed, ✓ 0 skipped, ✓ 119 failed, ✓ 0 errors, ✓ 0 expected failures, ✓ 0 unexpected passes

Results

Show all details / Hide all details

Result	Test	Duration	Links
Failed (hide details)	tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithImageFile::test_get_datetime_contains_special_case	0.00	

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithImageFile testMethod=test_get_datetime_contains_special_case>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2010-01-31_161251.jpg")

tests/unit_test_analyze_filename.py:32:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1040f9438>
path = "~/dev/projects/autonameow/test_files/2010-01-31_161251.jpg"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithImageFile::test_get_datetime_does_not_return_none | 0.00 | |

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithImageFile testMethod=test_get_datetime_does_not_return_none>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2010-01-31_161251.jpg")

tests/unit_test_analyze_filename.py:32:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1060cd978>
path = "~/dev/projects/autonameow/test_files/2010-01-31_161251.jpg"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithImageFile::test_get_datetime_special_case_is_valid | 0.00 | |

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithImageFile testMethod=test_get_datetime_special_case_is_valid>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2010-01-31_161251.jpg")

tests/unit_test_analyze_filename.py:32:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10613fb70>
path = "~/dev/projects/autonameow/test_files/2010-01-31_161251.jpg"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithImageFile::test_get_tags_does_not_return_none | 0.00 | |

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithImageFile testMethod=test_get_tags_does_not_return_none>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2010-01-31_161251.jpg")

tests/unit_test_analyze_filename.py:32:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1060c1550>
path = "~/dev/projects/autonameow/test_files/2010-01-31_161251.jpg"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithImageFile::test_get_tags_returns_empty_list | 0.00 | |

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithImageFile testMethod=test_get_tags_returns_empty_list>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2010-01-31_161251.jpg")

tests/unit_test_analyze_filename.py:32:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x106143a90>
path = "~/dev/projects/autonameow/test_files/2010-01-31_161251.jpg"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithImageFile::test_get_title_returns_empty_list | 0.00 | |

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithImageFile testMethod=test_get_title_returns_empty_list>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2010-01-31_161251.jpg")

tests/unit_test_analyze_filename.py:32:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x106169518>
path = "~/dev/projects/autonameow/test_files/2010-01-31_161251.jpg"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithImageFile::test_setup | 0.00 | |

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithImageFile testMethod=test_setup>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2010-01-31_161251.jpg")

tests/unit_test_analyze_filename.py:32:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x106162b70>
path = "~/dev/projects/autonameow/test_files/2010-01-31_161251.jpg"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithEmptyFile::test_get_datetime_does_not_return_none | 0.00 | |

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithEmptyFile testMethod=test_get_datetime_does_not_return_none>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/empty")

tests/unit_test_analyze_filename.py:69:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x106175f26>
path = "~/dev/projects/autonameow/test_files/empty"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithEmptyFile::test_get_datetime_returns_empty_list | 0.00 | |

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithEmptyFile testMethod=test_get_datetime_returns_empty_list>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/empty")

tests/unit_test_analyze_filename.py:69:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1061439e8>
path = "~/dev/projects/autonameow/test_files/empty"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithEmptyFile::test_get_tags_does_not_return_none | 0.00 | |

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithEmptyFile testMethod=test_get_tags_does_not_return_none>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/empty")

tests/unit_test_analyze_filename.py:69:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10617bc98>
path = "~/dev/projects/autonameow/test_files/empty"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithEmptyFile::test_get_tags_returns_empty_list | 0.00 | |

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithEmptyFile testMethod=test_get_tags_returns_empty_list>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/empty")

tests/unit_test_analyze_filename.py:69:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x106167f98>
path = "~/dev/projects/autonameow/test_files/empty"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithEmptyFile::test_get_title_does_not_return_none | 0.00 | |

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithEmptyFile testMethod=test_get_title_does_not_return_none>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/empty")

tests/unit_test_analyze_filename.py:69:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10617bfd0>
path = "~/dev/projects/autonameow/test_files/empty"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithEmptyFile::test_get_title_return_is_valid | 0.00 | |

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithEmptyFile testMethod=test_get_title_return_is_valid>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/empty")

tests/unit_test_analyze_filename.py:69:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10617b908>
path = "~/dev/projects/autonameow/test_files/empty"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithEmptyFile::test_setup | 0.00 | |

<tests.unit_test_analyze_filename.TestFilenameAnalyzerWithEmptyFile testMethod=test_setup>

```
    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/empty")

tests/unit_test_analyze_filename.py:69:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x1061a57f0>
path = "~/dev/projects/autonameow/test_files/empty"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithTaggedFile::test_get_datetime_contains_special_case   0.00

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithTaggedFile testMethod=test_get_datetime_contains_special_case>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv")

tests/unit_test_analyze_filename.py:101:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x1061790e0>
path = "~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithTaggedFile::test_get_datetime_does_not_return_none   0.00

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithTaggedFile testMethod=test_get_datetime_does_not_return_none>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv")

tests/unit_test_analyze_filename.py:101:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x1061a5898>
path = "~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithTaggedFile::test_get_datetime_special_case_is_valid   0.00

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithTaggedFile testMethod=test_get_datetime_special_case_is_valid>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv")

tests/unit_test_analyze_filename.py:101:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x106179c50>
path = "~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithTaggedFile::test_get_tags_does_not_return_none   0.00

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithTaggedFile testMethod=test_get_tags_does_not_return_none>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv")

tests/unit_test_analyze_filename.py:101:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x1061a44e0>
path = "~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithTaggedFile::test_get_tags_return_is_valid   0.00

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithTaggedFile testMethod=test_get_tags_return_is_valid>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv")

tests/unit_test_analyze_filename.py:101:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x106179f98>
path = "~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithTaggedFile::test_get_title_contains_filename   0.00

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithTaggedFile testMethod=test_get_title_contains_filename>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv")

tests/unit_test_analyze_filename.py:101:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x1061a4208>
path = "~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithTaggedFile::test_get_title_does_not_return_none   0.00

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithTaggedFile testMethod=test_get_title_does_not_return_none>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv")

tests/unit_test_analyze_filename.py:101:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x1061cedd8>
path = "~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithTaggedFile::test_get_title_return_is_valid   0.00

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithTaggedFile testMethod=test_get_title_return_is_valid>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv")

tests/unit_test_analyze_filename.py:101:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x1061bb710>
path = "~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_analyze_filename.py::TestFilenameAnalyzerWithTaggedFile::test_setup   0.00

```
self = <tests.unit_test_analyze_filename.TestFilenameAnalyzerWithTaggedFile testMethod=test_setup>

    def setUp(self):
>       self.fo = FileObject("~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv")

tests/unit_test_analyze_filename.py:101:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x1061b0780>
path = "~/dev/projects/autonameow/test_files/2015-07-03_163838 Keeping notes in Vim -- dv017a dev.ogv"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameNoTags::test__filenamepart_base   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameNoTags testMethod=test__filenamepart_base>

    def setUp(self):
>       self.fo = FileObject("20160722 Descriptive name.txt")

tests/unit_test_fileobject.py:31:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x106427ab0>
path = "20160722 Descriptive name.txt"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameNoTags::test__filenamepart_ext   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameNoTags testMethod=test__filenamepart_ext>

    def setUp(self):
>       self.fo = FileObject("20160722 Descriptive name.txt")

tests/unit_test_fileobject.py:31:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x10641 1be0>
path = "20160722 Descriptive name.txt"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameNoTags::test__filenamepart_tags   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameNoTags testMethod=test__filenamepart_tags>

    def setUp(self):
>       self.fo = FileObject("20160722 Descriptive name.txt")

tests/unit_test_fileobject.py:31:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x104488998>
path = "20160722 Descriptive name.txt"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameNoTags::test__filenamepart_ts   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameNoTags testMethod=test__filenamepart_ts>

    def setUp(self):
>       self.fo = FileObject("20160722 Descriptive name.txt")

tests/unit_test_fileobject.py:31:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x1045 0abe0>
path = "20160722 Descriptive name.txt"

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameNoTags::test_setUp   0.00

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameNoTags testMethod=test_setUp>

```
    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name.txt')

tests/unit_test_fileobject.py:31:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10450ab70>
path = '20160722 Descriptive name.txt'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameWithTags::test__filenamepart_base   0.00

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameWithTags testMethod=test__filenamepart_base>

```
    def setUp(self):
>       self.fo = FileObject('19990212 Descriptive name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:59:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x106498780>
path = '19990212 Descriptive name -- firsttag tagtwo.txt'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameWithTags::test__filenamepart_ext   0.00

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameWithTags testMethod=test__filenamepart_ext>

```
    def setUp(self):
>       self.fo = FileObject('19990212 Descriptive name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:59:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x106498cf8>
path = '19990212 Descriptive name -- firsttag tagtwo.txt'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameWithTags::test__filenamepart_tags   0.00

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameWithTags testMethod=test__filenamepart_tags>

```
    def setUp(self):
>       self.fo = FileObject('19990212 Descriptive name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:59:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10446e710>
path = '19990212 Descriptive name -- firsttag tagtwo.txt'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameWithTags::test__filenamepart_ts   0.00

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameWithTags testMethod=test__filenamepart_ts>

```
    def setUp(self):
>       self.fo = FileObject('19990212 Descriptive name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:59:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x106167fd0>
path = '19990212 Descriptive name -- firsttag tagtwo.txt'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameWithTags::test_setUp   0.00

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameWithTags testMethod=test_setUp>

```
    def setUp(self):
>       self.fo = FileObject('19990212 Descriptive name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:59:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x106455f0>
path = '19990212 Descriptive name -- firsttag tagtwo.txt'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameWithTagsDashesInName::test__filenamepart_base   0.00

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameWithTagsDashesInName testMethod=test__filenamepart_base>

```
    def setUp(self):
>       self.fo = FileObject('19920722 --Descriptive-- name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:90:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10449e4a0>
path = '19920722 --Descriptive-- name -- firsttag tagtwo.txt'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameWithTagsDashesInName::test__filenamepart_ext   0.00

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameWithTagsDashesInName testMethod=test__filenamepart_ext>

```
    def setUp(self):
>       self.fo = FileObject('19920722 --Descriptive-- name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:90:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1061b04e0>
path = '19920722 --Descriptive-- name -- firsttag tagtwo.txt'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameWithTagsDashesInName::test__filenamepart_tags   0.00

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameWithTagsDashesInName testMethod=test__filenamepart_tags>

```
    def setUp(self):
>       self.fo = FileObject('19920722 --Descriptive-- name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:90:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10447bdd8>
path = '19920722 --Descriptive-- name -- firsttag.tat'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameWithTagsDashesInName::test__filenamepart_ts   0.00

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameWithTagsDashesInName testMethod=test__filenamepart_ts>

```
    def setUp(self):
>       self.fo = FileObject('19920722 --Descriptive-- name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:90:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10449c518>
path = '19920722 --Descriptive-- name -- firsttag tagtwo.txt'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningLongNameWithTagsDashesInName::test_setUp   0.00

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningLongNameWithTagsDashesInName testMethod=test_setUp>

```
    def setUp(self):
>       self.fo = FileObject('19920722 --Descriptive-- name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:90:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10446d860>
path = '19920722 --Descriptive-- name -- firsttag tagtwo.txt'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningHiddenFileNoExtensionNoTags::test__filenamepart_base   0.00

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileNoExtensionNoTags testMethod=test__filenamepart_base>

```
    def setUp(self):
>       self.fo = FileObject('.name')

tests/unit_test_fileobject.py:121:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10449eb36>, path = '.name'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningHiddenFileNoExtensionNoTags::test__filenamepart_ext   0.00

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileNoExtensionNoTags testMethod=test__filenamepart_ext>

```
    def setUp(self):
>       self.fo = FileObject('.name')

tests/unit_test_fileobject.py:121:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x104630f0>, path = '.name'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningHiddenFileNoExtensionNoTags::test__filenamepart_tags   0.00

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileNoExtensionNoTags testMethod=test__filenamepart_tags>

```
    def setUp(self):
>       self.fo = FileObject('.name')

tests/unit_test_fileobject.py:121:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1064c8a90>, path = '.name'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningHiddenFileNoExtensionNoTags::test__filenamepart_ts   0.00

setUp tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileNoExtensionNoTags testMethod=test__filenamepart_ts>

```
    def setUp(self):
>       self.fo = FileObject(".name")

tests/unit_test_fileobject.py:121:
```

```
self = <core.fileobject.FileObject object at 0x1044bc18>, path = '.name' self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileNoExtensionNoTags testMethod=test_setUp>

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
tests/unit_test_fileobject.py:121: InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
self = <core.fileobject.FileObject object at 0x1044bc18>, path = '.name'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileNoTags testMethod=test__filenamepart_base>

    def setUp(self):
>       self.fo = FileObject(".name.jpg")

tests/unit_test_fileobject.py:143:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1044bcf26>, path = '.name.jpg'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileNoTags testMethod=test__filenamepart_ext>

    def setUp(self):
>       self.fo = FileObject(".name.jpg")

tests/unit_test_fileobject.py:143:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x10434dd8>, path = '.name.jpg'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileNoTags testMethod=test__filenamepart_tags>

    def setUp(self):
>       self.fo = FileObject(".name.jpg")

tests/unit_test_fileobject.py:143:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x10434f60>, path = '.name.jpg'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileNoTags testMethod=test__filenamepart_ts>

    def setUp(self):
>       self.fo = FileObject(".name.jpg")

tests/unit_test_fileobject.py:143:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1045c2b0>, path = '.name.jpg'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileNoTags testMethod=test_setUp>

    def setUp(self):
>       self.fo = FileObject(".name.jpg")

tests/unit_test_fileobject.py:143:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1044da89>, path = '.name.jpg'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileCompoundSuffix testMethod=test__filenamepart_base>

    def setUp(self):
>       self.fo = FileObject(".name.tar.gz")

tests/unit_test_fileobject.py:169:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1044c8e48>, path = '.name.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileCompoundSuffix testMethod=test__filenamepart_ext>

    def setUp(self):
>       self.fo = FileObject(".name.tar.gz")

tests/unit_test_fileobject.py:169:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x106557a20>, path = '.name.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileCompoundSuffix testMethod=test__filenamepart_tags>

    def setUp(self):
>       self.fo = FileObject(".name.tar.gz")

tests/unit_test_fileobject.py:169:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x10652497b>, path = '.name.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileCompoundSuffix testMethod=test__filenamepart_ts>

    def setUp(self):
>       self.fo = FileObject(".name.tar.gz")

tests/unit_test_fileobject.py:169:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1044a990f>, path = '.name.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileCompoundSuffix testMethod=test_setUp>

    def setUp(self):
>       self.fo = FileObject(".name.tar.gz")

tests/unit_test_fileobject.py:169:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1044c7e10>, path = '.name.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileWithTags testMethod=test__filenamepart_base>

    def setUp(self):
>       self.fo = FileObject(".name -- firsttag 2ndtag.jpg")

tests/unit_test_fileobject.py:196:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1044e4b00>
path = '.name -- firsttag 2ndtag.jpg'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileWithTags testMethod=test__filenamepart_ext>

    def setUp(self):
>       self.fo = FileObject(".name -- firsttag 2ndtag.jpg")

tests/unit_test_fileobject.py:196:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x106502ac8>
path = '.name -- firsttag 2ndtag.jpg'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileWithTags testMethod=test__filenamepart_tags>

    def setUp(self):
>       self.fo = FileObject(".name -- firsttag 2ndtag.jpg")

tests/unit_test_fileobject.py:196:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1044cfba8>
path = '.name -- firsttag 2ndtag.jpg'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

setUp (<tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileWithTags testMethod=test__filenamepart_ts>

```
    def setUp(self):
>       self.fo = FileObject('.name -- firsttag 2ndtag.jpg')

tests/unit_test_fileobject.py:196:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
```

Failed (hide details)   tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningHiddenFileWithTags:test_setUp   0.00

```
self = <core.fileobject.FileObject object at 0x104625fe8>
path = '.name -- firsttag 2ndtag.jpg'
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileWithTags testMethod=test_setUp>

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

self = <core.fileobject.FileObject object at 0x104d7d30>
path = '.name -- firsttag 2ndtag.jpg'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningHiddenFileCompoundSuffixTags:test__filenamepart_base   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileCompoundSuffixTags testMethod=test__filenamepart_base>

    def setUp(self):
>       self.fo = FileObject('.name -- firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:225:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1044ea6d8>
path = '.name -- firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningHiddenFileCompoundSuffixTags:test__filenamepart_ext   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileCompoundSuffixTags testMethod=test__filenamepart_ext>

    def setUp(self):
>       self.fo = FileObject('.name -- firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:225:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10461bcf8>
path = '.name -- firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningHiddenFileCompoundSuffixTags:test__filenamepart_tags   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileCompoundSuffixTags testMethod=test__filenamepart_tags>

    def setUp(self):
>       self.fo = FileObject('.name -- firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:225:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10462f6a0>
path = '.name -- firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningHiddenFileCompoundSuffixTags:test__filenamepart_ts   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileCompoundSuffixTags testMethod=test__filenamepart_ts>

    def setUp(self):
>       self.fo = FileObject('.name -- firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:225:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x104629%8>
path = '.name -- firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningHiddenFileCompoundSuffixTags:test_setUp   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningHiddenFileCompoundSuffixTags testMethod=test_setUp>

    def setUp(self):
>       self.fo = FileObject('.name -- firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:225:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10462f048>
path = '.name -- firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningDifficultCombination:test__filenamepart_base   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningDifficultCombination testMethod=test__filenamepart_base>

    def setUp(self):
>       self.fo = FileObject('.name -- tar firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:259:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10463e908>
path = '.name -- tar firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningDifficultCombination:test__filenamepart_ext   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningDifficultCombination testMethod=test__filenamepart_ext>

    def setUp(self):
>       self.fo = FileObject('.name -- tar firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:259:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10462feb8>
path = '.name -- tar firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningDifficultCombination:test__filenamepart_tags   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningDifficultCombination testMethod=test__filenamepart_tags>

    def setUp(self):
>       self.fo = FileObject('.name -- tar firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:259:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x104547750>
path = '.name -- tar firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningDifficultCombination:test__filenamepart_ts   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningDifficultCombination testMethod=test__filenamepart_ts>

    def setUp(self):
>       self.fo = FileObject('.name -- tar firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:259:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10462f208>
path = '.name -- tar firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningDifficultCombination:test_setUp   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningDifficultCombination testMethod=test_setUp>

    def setUp(self):
>       self.fo = FileObject('.name -- tar firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:259:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x104559438>
path = '.name -- tar firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningAnotherDifficultCombination:test__filenamepart_base   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningAnotherDifficultCombination testMethod=test__filenamepart_base>

    def setUp(self):
>       self.fo = FileObject('.tar name -- gz firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:294:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1065421f0>
path = '.tar name -- gz firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningAnotherDifficultCombination:test__filenamepart_ext   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningAnotherDifficultCombination testMethod=test__filenamepart_ext>

    def setUp(self):
>       self.fo = FileObject('.tar name -- gz firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:294:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x104962b0>
path = '.tar name -- gz firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningAnotherDifficultCombination:test__filenamepart_tags   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningAnotherDifficultCombination testMethod=test__filenamepart_tags>

    def setUp(self):
>       self.fo = FileObject('.tar name -- gz firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:294:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1044ea7b8>
path = '.tar name -- gz firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningAnotherDifficultCombination testMethod=test__filenamepart_ts>

    def setUp(self):
>       self.fo = FileObject('.tar name -- gs firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:294:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10652a4a0>
path = '.tar name -- gs firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError("Path does not exist")
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningAnotherDifficultCombination::test_setUp   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningAnotherDifficultCombination testMethod=test_setUp>

    def setUp(self):
>       self.fo = FileObject('.tar name -- gs firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:294:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1065242b0>
path = '.tar name -- gs firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError("Path does not exist")
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningEvenMoreDifficultCombination::test__filenamepart_base   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningEvenMoreDifficultCombination testMethod=test__filenamepart_base>

    def setUp(self):
>       self.fo = FileObject('.tar.gz name -- gs firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:329:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1064df080>
path = '.tar.gz name -- gs firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError("Path does not exist")
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningEvenMoreDifficultCombination::test__filenamepart_ext   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningEvenMoreDifficultCombination testMethod=test__filenamepart_ext>

    def setUp(self):
>       self.fo = FileObject('.tar.gz name -- gs firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:329:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x106463898>
path = '.tar.gz name -- gs firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError("Path does not exist")
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningEvenMoreDifficultCombination::test__filenamepart_tags   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningEvenMoreDifficultCombination testMethod=test__filenamepart_tags>

    def setUp(self):
>       self.fo = FileObject('.tar.gz name -- gs firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:329:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10647f5c0>
path = '.tar.gz name -- gs firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError("Path does not exist")
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningEvenMoreDifficultCombination::test__filenamepart_ts   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningEvenMoreDifficultCombination testMethod=test__filenamepart_ts>

    def setUp(self):
>       self.fo = FileObject('.tar.gz name -- gs firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:329:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1064daf98>
path = '.tar.gz name -- gs firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError("Path does not exist")
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningEvenMoreDifficultCombination::test_setUp   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningEvenMoreDifficultCombination testMethod=test_setUp>

    def setUp(self):
>       self.fo = FileObject('.tar.gz name -- gs firsttag 2ndtag.tar.gz')

tests/unit_test_fileobject.py:329:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1064483550>
path = '.tar.gz name -- gs firsttag 2ndtag.tar.gz'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError("Path does not exist")
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueType::test__filenamepart_base_returns_string_or_none   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueType testMethod=test__filenamepart_base_returns_string_or_none>

    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:364:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1046bee80>
path = '20160722 Descriptive name -- firsttag tagtwo.txt'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError("Path does not exist")
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueType::test__filenamepart_ext_returns_string_or_none   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueType testMethod=test__filenamepart_ext_returns_string_or_none>

    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:364:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1064474b70>
path = '20160722 Descriptive name -- firsttag tagtwo.txt'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError("Path does not exist")
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueType::test__filenamepart_tags_returns_list_of_string_or_empty_list   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueType testMethod=test__filenamepart_tags_returns_list_of_string_or_empty_list>

    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:364:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1064dd320>
path = '20160722 Descriptive name -- firsttag tagtwo.txt'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError("Path does not exist")
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueType::test__filenamepart_ts_returns_string_or_none   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueType testMethod=test__filenamepart_ts_returns_string_or_none>

    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:364:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10652cf98>
path = '20160722 Descriptive name -- firsttag tagtwo.txt'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError("Path does not exist")
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueType::test_setUp   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueType testMethod=test_setUp>

    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name -- firsttag tagtwo.txt')

tests/unit_test_fileobject.py:364:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1064517c50>
path = '20160722 Descriptive name -- firsttag tagtwo.txt'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError("Path does not exist")
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueTypeNoTags::test__filenamepart_base_returns_string_or_none   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueTypeNoTags testMethod=test__filenamepart_base_returns_string_or_none>

    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name.txt')

tests/unit_test_fileobject.py:386:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1064e65c0>
path = '20160722 Descriptive name.txt'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError("Path does not exist")
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueTypeNoTags::test__filenamepart_ext_returns_string_or_none   0.00

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueTypeNoTags testMethod=test__filenamepart_ext_returns_string_or_none>

    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name.txt')

tests/unit_test_fileobject.py:386:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1044bc50>
path = '20160722 Descriptive name.txt'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError("Path does not exist")
>           raise InvalidFileArgumentError("Path does not exist")
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)   tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueTypeNoTags::test__filenamepart_tags_returns_list_of_string_or_empty_list   0.00

| | | Duration | Links |

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueTypeNoTags testMethod=test__filenamepart_tags_returns_list_of_string_or_empty_list>

```
    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name.txt')

tests/unit_test_fileobject.py:386:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1045515978>
path = '20160722 Descriptive name.txt'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueTypeNoTags::test__filenamepart_ts_returns_string_or_none | 0.00 | |

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueTypeNoTags testMethod=test__filenamepart_ts_returns_string_or_none>

```
    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name.txt')

tests/unit_test_fileobject.py:386:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1045515c0>
path = '20160722 Descriptive name.txt'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueTypeNoTags::test_setUp | 0.00 | |

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueTypeNoTags testMethod=test_setUp>

```
    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name.txt')

tests/unit_test_fileobject.py:386:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x104517358>
path = '20160722 Descriptive name.txt'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueTypeNoTagsNoExt::test__filenamepart_base_returns_string_or_none | 0.00 | |

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueTypeNoTagsNoExt testMethod=test__filenamepart_base_returns_string_or_none>

```
    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name')

tests/unit_test_fileobject.py:408:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x106633390>
path = '20160722 Descriptive name'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueTypeNoTagsNoExt::test__filenamepart_ext_returns_string_or_none | 0.00 | |

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueTypeNoTagsNoExt testMethod=test__filenamepart_ext_returns_string_or_none>

```
    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name')

tests/unit_test_fileobject.py:408:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1064f3e80>
path = '20160722 Descriptive name'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueTypeNoTagsNoExt::test__filenamepart_tags_returns_list_of_string_or_empty_list | 0.00 | |

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueTypeNoTagsNoExt testMethod=test__filenamepart_tags_returns_list_of_string_or_empty_list>

```
    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name')

tests/unit_test_fileobject.py:408:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x104515518>
path = '20160722 Descriptive name'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueTypeNoTagsNoExt::test__filenamepart_ts_returns_string_or_none | 0.00 | |

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueTypeNoTagsNoExt testMethod=test__filenamepart_ts_returns_string_or_none>

```
    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name')

tests/unit_test_fileobject.py:408:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1045952b0>
path = '20160722 Descriptive name'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningReturnValueTypeNoTagsNoExt::test_setUp | 0.00 | |

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningReturnValueTypeNoTagsNoExt testMethod=test_setUp>

```
    def setUp(self):
>       self.fo = FileObject('20160722 Descriptive name')

tests/unit_test_fileobject.py:408:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10453f7b8>
path = '20160722 Descriptive name'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningWithActualFilename::test_filenamepart_base | 0.00 | |

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename testMethod=test_filenamepart_base>

```
    def setUp(self):
>       self.fo = FileObject('2016-08-05 18-46-34 Working on PLL-monstret -- projects frfx.png')

tests/unit_test_fileobject.py:431:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x104967b8>
path = '2016-08-05 18-46-34 Working on PLL-monstret -- projects frfx.png'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningWithActualFilename::test_filenamepart_ext | 0.00 | |

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename testMethod=test_filenamepart_ext>

```
    def setUp(self):
>       self.fo = FileObject('2016-08-05 18-46-34 Working on PLL-monstret -- projects frfx.png')

tests/unit_test_fileobject.py:431:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1045561d0>
path = '2016-08-05 18-46-34 Working on PLL-monstret -- projects frfx.png'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningWithActualFilename::test_filenamepart_tags | 0.00 | |

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename testMethod=test_filenamepart_tags>

```
    def setUp(self):
>       self.fo = FileObject('2016-08-05 18-46-34 Working on PLL-monstret -- projects frfx.png')

tests/unit_test_fileobject.py:431:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10460de80>
path = '2016-08-05 18-46-34 Working on PLL-monstret -- projects frfx.png'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningWithActualFilename::test_filenamepart_ts | 0.00 | |

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename testMethod=test_filenamepart_ts>

```
    def setUp(self):
>       self.fo = FileObject('2016-08-05 18-46-34 Working on PLL-monstret -- projects frfx.png')

tests/unit_test_fileobject.py:431:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10458068d0>
path = '2016-08-05 18-46-34 Working on PLL-monstret -- projects frfx.png'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningWithActualFilename::test_setUp | 0.00 | |

self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename testMethod=test_setUp>

```
    def setUp(self):
>       self.fo = FileObject('2016-08-05 18-46-34 Working on PLL-monstret -- projects frfx.png')

tests/unit_test_fileobject.py:431:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x104c4dd8>
path = '2016-08-05 18-46-34 Working on PLL-monstret -- projects frfx.png'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| **Failed** (hide details) | tests/unit_test_fileobject.py::TestFileObjectFilenamePartitioningWithActualFilename2::test_filenamepart_base | 0.00 | |

```
self <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename2 testMethod=test_filenamepart_base>

    def setUp(self):
>       self.fo = FileObject('2016-07-30T175241 Drawing with Wanvon tablet in krita_x86_xp_2.8.1.1 -- projects.png')
```

| Failed (hide details) | tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningWithActualFilename2:test_filenamepart_ext | 0.00 | |

```
tests/unit_test_fileobject.py:451:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x10695a58>
path = '2016-07-30T175241 Drawing with Wanvon tablet in krita_x86_xp_2.8.1.1 -- projects.png'
    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

```
path = '2016-07-30T175241 Drawing with Wanvon tablet in krita_x86_xp_2.8.1.1 -- projects.png'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| Failed (hide details) | tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningWithActualFilename2:test_filenamepart_tags | 0.00 | |

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename2 testMethod=test_filenamepart_tags>

    def setUp(self):
>       self.fo = FileObject('2016-07-30T175241 Drawing with Wanvon tablet in krita_x86_xp_2.8.1.1 -- projects.png')

tests/unit_test_fileobject.py:451:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x106508d0>
path = '2016-07-30T175241 Drawing with Wanvon tablet in krita_x86_xp_2.8.1.1 -- projects.png'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| Failed (hide details) | tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningWithActualFilename2:test_filenamepart_ts | 0.00 | |

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename2 testMethod=test_filenamepart_ts>

    def setUp(self):
>       self.fo = FileObject('2016-07-30T175241 Drawing with Wanvon tablet in krita_x86_xp_2.8.1.1 -- projects.png')

tests/unit_test_fileobject.py:451:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1064dd128>
path = '2016-07-30T175241 Drawing with Wanvon tablet in krita_x86_xp_2.8.1.1 -- projects.png'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| Failed (hide details) | tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningWithActualFilename2:test_setUp | 0.00 | |

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename2 testMethod=test_setUp>

    def setUp(self):
>       self.fo = FileObject('2016-07-30T175241 Drawing with Wanvon tablet in krita_x86_xp_2.8.1.1 -- projects.png')

tests/unit_test_fileobject.py:451:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1065a17f0>
path = '2016-07-30T175241 Drawing with Wanvon tablet in krita_x86_xp_2.8.1.1 -- projects.png'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| Failed (hide details) | tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningWithActualFilename3:test_filenamepart_base | 0.00 | |

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename3 testMethod=test_filenamepart_base>

    def setUp(self):
>       self.fo = FileObject('2016-08-01_104304_pages.stern.nyu.edu oyepa - Linux tagging filesystem -- workflow pim.html')

tests/unit_test_fileobject.py:472:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x106633be0>
path = '2016-08-01_104304_pages.stern.nyu.edu oyepa - Linux tagging filesystem -- workflow pim.html'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| Failed (hide details) | tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningWithActualFilename3:test_filenamepart_ext | 0.00 | |

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename3 testMethod=test_filenamepart_ext>

    def setUp(self):
>       self.fo = FileObject('2016-08-01_104304_pages.stern.nyu.edu oyepa - Linux tagging filesystem -- workflow pim.html')

tests/unit_test_fileobject.py:472:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1066a4940>
path = '2016-08-01_104304_pages.stern.nyu.edu oyepa - Linux tagging filesystem -- workflow pim.html'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| Failed (hide details) | tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningWithActualFilename3:test_filenamepart_tags | 0.00 | |

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename3 testMethod=test_filenamepart_tags>

    def setUp(self):
>       self.fo = FileObject('2016-08-01_104304_pages.stern.nyu.edu oyepa - Linux tagging filesystem -- workflow pim.html')

tests/unit_test_fileobject.py:472:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1064feld0>
path = '2016-08-01_104304_pages.stern.nyu.edu oyepa - Linux tagging filesystem -- workflow pim.html'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| Failed (hide details) | tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningWithActualFilename3:test_filenamepart_ts | 0.00 | |

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename3 testMethod=test_filenamepart_ts>

    def setUp(self):
>       self.fo = FileObject('2016-08-01_104304_pages.stern.nyu.edu oyepa - Linux tagging filesystem -- workflow pim.html')

tests/unit_test_fileobject.py:472:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1064f3898>
path = '2016-08-01_104304_pages.stern.nyu.edu oyepa - Linux tagging filesystem -- workflow pim.html'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| Failed (hide details) | tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningWithActualFilename3:test_setUp | 0.00 | |

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename3 testMethod=test_setUp>

    def setUp(self):
>       self.fo = FileObject('2016-08-01_104304_pages.stern.nyu.edu oyepa - Linux tagging filesystem -- workflow pim.html')

tests/unit_test_fileobject.py:472:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x10653fef0>
path = '2016-08-01_104304_pages.stern.nyu.edu oyepa - Linux tagging filesystem -- workflow pim.html'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| Failed (hide details) | tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningWithActualFilename4:test_filenamepart_base | 0.00 | |

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename4 testMethod=test_filenamepart_base>

    def setUp(self):
>       self.fo = FileObject('2010-01-31_161251.jpg')

tests/unit_test_fileobject.py:493:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1065a05f8>
path = '2010-01-31_161251.jpg'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| Failed (hide details) | tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningWithActualFilename4:test_filenamepart_ext | 0.00 | |

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename4 testMethod=test_filenamepart_ext>

    def setUp(self):
>       self.fo = FileObject('2010-01-31_161251.jpg')

tests/unit_test_fileobject.py:493:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1064617c50>
path = '2010-01-31_161251.jpg'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| Failed (hide details) | tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningWithActualFilename4:test_filenamepart_tags | 0.00 | |

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename4 testMethod=test_filenamepart_tags>

    def setUp(self):
>       self.fo = FileObject('2010-01-31_161251.jpg')

tests/unit_test_fileobject.py:493:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1064c8da0>
path = '2010-01-31_161251.jpg'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| Failed (hide details) | tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningWithActualFilename4:test_filenamepart_ts | 0.00 | |

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename4 testMethod=test_filenamepart_ts>

    def setUp(self):
>       self.fo = FileObject('2010-01-31_161251.jpg')

tests/unit_test_fileobject.py:493:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1065f4c18>
path = '2010-01-31_161251.jpg'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

| Failed (hide details) | tests/unit_test_fileobject.py:TestFileObjectFilenamePartitioningWithActualFilename4:test_setUp | 0.00 | |

```
self = <tests.unit_test_fileobject.TestFileObjectFilenamePartitioningWithActualFilename4 testMethod=test_setUp>

    def setUp(self):
>       self.fo = FileObject('2010-01-31_161251.jpg')

tests/unit_test_fileobject.py:493:
- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
self = <core.fileobject.FileObject object at 0x1064689b0>
path = '2010-01-31_161251.jpg'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Result (hide details)	tests/unit_test_fileobject.py::TestFileObjectFilenameIsInFiletagsFormat::test_has_filetags_format	0.00 Duration	Links

```
self = <tests.unit_test_fileobject.TestFileObjectFilenameIsInFiletagsFormat testMethod=test_has_filetags_format>

    def test_has_filetags_format(self):
>       self.fo = FileObject('2016-08-05_18-46-34 Working on PLL-monstret -- projects frfx.png')

tests/unit_test_fileobject.py:526:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10685434>
path = '2016-08-05_18-46-34 Working on PLL-monstret -- projects frfx.png'

    def __init__(self, path):
        if not os.path.exists(path):
>           raise InvalidFileArgumentError('Path does not exist')
E           exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)	tests/unit_test_fileobject.py::TestFileObjectFilenameIsInFiletagsFormat::test_has_filetags_format_date_only	0.00	

```
self = <tests.unit_test_fileobject.TestFileObjectFilenameIsInFiletagsFormat testMethod=test_has_filetags_format_date_only>

    def test_has_filetags_format_date_only(self):
>       self.fo = FileObject('2016-08-05 Working on PLL-monstret -- projects frfx.png')

tests/unit_test_fileobject.py:534:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10d595908>
path = '2016-08-05 Working on PLL-monstret -- projects frfx.png'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError('Path does not exist')
>           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)	tests/unit_test_fileobject.py::TestFileObjectFilenameIsInFiletagsFormat::test_has_filetags_format_no_extension	0.00	

```
self = <tests.unit_test_fileobject.TestFileObjectFilenameIsInFiletagsFormat testMethod=test_has_filetags_format_no_extension>

    def test_has_filetags_format_no_extension(self):
>       self.fo = FileObject('2016-08-05_18-46-34 Working on PLL-monstret -- projects frfx')

tests/unit_test_fileobject.py:530:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10d6b4208>
path = '2016-08-05_18-46-34 Working on PLL-monstret -- projects frfx'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError('Path does not exist')
>           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)	tests/unit_test_fileobject.py::TestFileObjectFilenameNotInFiletagsFormat::test_doesnt_have_filetags_format_missing_fnpart_base	0.00	

```
self = <tests.unit_test_fileobject.TestFileObjectFilenameNotInFiletagsFormat testMethod=test_doesnt_have_filetags_format_missing_fnpart_base>

    def test_doesnt_have_filetags_format_missing_fnpart_base(self):
>       self.fo = FileObject('2016-08-05_18-46-34 -- projects frfx.png')

tests/unit_test_fileobject.py:545:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10657dc98>
path = '2016-08-05_18-46-34 -- projects frfx.png'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError('Path does not exist')
>           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)	tests/unit_test_fileobject.py::TestFileObjectFilenameNotInFiletagsFormat::test_doesnt_have_filetags_format_missing_fnpart_tags	0.00	

```
self = <tests.unit_test_fileobject.TestFileObjectFilenameNotInFiletagsFormat testMethod=test_doesnt_have_filetags_format_missing_fnpart_tags>

    def test_doesnt_have_filetags_format_missing_fnpart_tags(self):
>       self.fo = FileObject('2016-08-05_18-46-34 Working on PLL-monstret.png')

tests/unit_test_fileobject.py:549:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x10657a1f28>
path = '2016-08-05_18-46-34 Working on PLL-monstret.png'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError('Path does not exist')
E           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Failed (hide details)	tests/unit_test_fileobject.py::TestFileObjectFilenameNotInFiletagsFormat::test_doesnt_have_filetags_format_missing_fnpart_ts	0.00	

```
self = <tests.unit_test_fileobject.TestFileObjectFilenameNotInFiletagsFormat testMethod=test_doesnt_have_filetags_format_missing_fnpart_ts>

    def test_doesnt_have_filetags_format_missing_fnpart_ts(self):
>       self.fo = FileObject('Working on PLL-monstret.png')

tests/unit_test_fileobject.py:553:
_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _
self = <core.fileobject.FileObject object at 0x1065d7828>
path = 'Working on PLL-monstret.png'

    def __init__(self, path):
        if not os.path.exists(path):
            raise InvalidFileArgumentError('Path does not exist')
>           core.exceptions.InvalidFileArgumentError: Path does not exist

autonameow/core/fileobject.py:46: InvalidFileArgumentError
```

Skipped (hide details)	tests/unit_test_analysis.py::TestAnalysisUtilityFunctions::test_get_analyzer_classes_returns_expected_contents	0.00	

('tests/unit_test_analysis.py', 90, 'Skipped: TODO: Implement ..')

Skipped (hide details)	tests/unit_test_analysis.py::TestAnalysisUtilityFunctions::test_get_instantiated_analyzers	0.00	

('tests/unit_test_analysis.py', 94, 'Skipped: Fix/skip this test ..')

Skipped (hide details)	tests/unit_test_analyze_filesystem.py::TestFilesystemAnalyzerWithEmptyFile::test_get_datetime_from_filesystem	0.00	

('tests/unit_test_analyze_filesystem.py', 108, 'Skipped: Create and access time might have changed. Should be set to a known state as part of the tests setup.')

Skipped (hide details)	tests/unit_test_analyze_filesystem.py::TestFilesystemAnalyzerWithEmptyFile::test_get_datetime_filesystem_accessed_is_valid	0.00	

('tests/unit_test_analyze_filesystem.py', 90, 'Skipped: Access time might have changed between test runs.')

Skipped (hide details)	tests/unit_test_analyze_filesystem.py::TestFilesystemAnalyzerWithEmptyFile::test_get_datetime_filesystem_created_is_valid	0.00	

('tests/unit_test_analyze_filesystem.py', 75, 'Skipped: Create time might have changed. Should be set to a known time as part of the tests setup.')

Passed (show details)	tests/unit_test_analysis.py::TestTests::test_constant_expect_analyzer_classes	0.00	
Passed (show details)	tests/unit_test_analysis.py::TestTests::test_constant_expect_analyzer_classes_basename	0.00	
Passed (show details)	tests/unit_test_analysis.py::TestAnalysisUtilityFunctions::test_get_analysis_mime_mappings	0.00	
Passed (show details)	tests/unit_test_analysis.py::TestAnalysisUtilityFunctions::test_get_analyzer_classes_basename_returns_expected_contents	0.00	
Passed (show details)	tests/unit_test_analysis.py::TestAnalysisUtilityFunctions::test_get_analyzer_classes_basename_returns_expected_count	0.00	
Passed (show details)	tests/unit_test_analysis.py::TestAnalysisUtilityFunctions::test_get_analyzer_classes_returns_expected_count	0.00	
Passed (show details)	tests/unit_test_analysis.py::TestResults::test_add	0.00	
Passed (show details)	tests/unit_test_analysis.py::TestResults::test_add_with_invalid_analyzer_raises_exception	0.00	
Passed (show details)	tests/unit_test_analysis.py::TestResults::test_add_with_invalid_field_raises_exception	0.00	
Passed (show details)	tests/unit_test_analysis.py::TestResults::test_results_init_contains_all_valid_results_fields	0.00	
Passed (show details)	tests/unit_test_analysis.py::TestResults::test_results_init_contains_valid_results_fields	0.00	
Passed (show details)	tests/unit_test_analysis.py::TestResults::test_results_init_results_fields_are_empty	0.00	
Passed (show details)	tests/unit_test_analysis.py::TestResults::test_results_init_results_fields_are_type_dict	0.00	
Passed (show details)	tests/unit_test_analyze_filesystem.py::TestFilesystemAnalyzerWithEmptyFile::test_get_author_should_return_none	0.00	
Passed (show details)	tests/unit_test_analyze_filesystem.py::TestFilesystemAnalyzerWithEmptyFile::test_get_datetime_contains_filesystem_accessed	0.00	
Passed (show details)	tests/unit_test_analyze_filesystem.py::TestFilesystemAnalyzerWithEmptyFile::test_get_datetime_contains_filesystem_created	0.00	
Passed (show details)	tests/unit_test_analyze_filesystem.py::TestFilesystemAnalyzerWithEmptyFile::test_get_datetime_contains_filesystem_modified	0.00	
Passed (show details)	tests/unit_test_analyze_filesystem.py::TestFilesystemAnalyzerWithEmptyFile::test_get_datetime_does_not_return_none	0.00	
Passed (show details)	tests/unit_test_analyze_filesystem.py::TestFilesystemAnalyzerWithEmptyFile::test_get_datetime_filesystem_modified_is_valid	0.00	
Passed (show details)	tests/unit_test_analyze_filesystem.py::TestFilesystemAnalyzerWithEmptyFile::test_get_tags_returns_none	0.00	
Passed (show details)	tests/unit_test_analyze_filesystem.py::TestFilesystemAnalyzerWithEmptyFile::test_get_title_should_return_none	0.00	
Passed (show details)	tests/unit_test_analyze_filesystem.py::TestFilesystemAnalyzerWithEmptyFile::test_setup	0.00	
Passed (show details)	tests/unit_test_autonameow.py::TestAutonameow::test_setup	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestWriteConfig::test_load_from_dict	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestWriteConfig::test_setup	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestWriteConfig::test_write_and_verify	0.04	
Passed (show details)	tests/unit_test_configuration.py::TestWriteConfig::test_write_config	0.02	
Passed (show details)	tests/unit_test_configuration.py::TestDefaultConfig::test_default_configuration_contain_at_least_two_rules	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestDefaultConfig::test_default_configuration_contain_name_templates	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestDefaultConfig::test_default_configuration_contain_rules	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestDefaultConfig::test_default_configuration_exists	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestWriteDefaultConfig::test_load_default_config_from_dict_before_write	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestWriteDefaultConfig::test_setup	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestWriteDefaultConfig::test_write_default_config_to_disk	0.02	
Passed (show details)	tests/unit_test_configuration.py::TestWriteDefaultConfig::test_write_default_config_to_disk_and_verify	0.05	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationInit::test_configuration_parsers_in_not_none	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationInit::test_configuration_parsers_instance_of_parser	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationInit::test_configuration_parsers_subclass_of_parser	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationValidation::test_setup	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationValidation::test_validate_invalid_rule_conditions_filename_basename	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationValidation::test_validate_invalid_rule_conditions_filename_extension	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationValidation::test_validate_invalid_rule_conditions_filename_mime_type	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationValidation::test_validate_invalid_rule_conditions_filename_pathname	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationValidation::test_validate_invalid_rule_name_format	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationValidation::test_validate_valid_rule_conditions_filename_basename	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationValidation::test_validate_valid_rule_conditions_filename_extension	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationValidation::test_validate_valid_rule_conditions_filename_mime_type	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationValidation::test_validate_valid_rule_conditions_filename_pathname	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationValidation::test_validate_valid_rule_name_format	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationDataAccess::test_get_data_does_not_return_none	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationDataAccess::test_get_file_rules_does_not_return_none	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationDataAccess::test_get_file_rules_returns_expected_rule_count	0.00	
Passed (show details)	tests/unit_test_configuration.py::TestConfigurationDataAccess::test_get_file_rules_returns_expected_type	0.00	
Passed (show details)	tests/unit_test_dateandtime.py::TestDateAndTime::test_hyphenate_date	0.00	
Passed (show details)	tests/unit_test_dateandtime.py::TestMatchUnixTimestamp::test_match_unix_timestamp	0.00	
Passed (show details)	tests/unit_test_dateandtime.py::TestMatchUnixTimestamp::test_match_unix_timestamp_with_invalid_argument	0.00	
Passed (show details)	tests/unit_test_dateandtime.py::TestMatchSpecialCase::test_match_special_case_1st_variation	0.00	
Passed (show details)	tests/unit_test_dateandtime.py::TestMatchSpecialCase::test_match_special_case_2nd_variation	0.00	
Passed (show details)	tests/unit_test_dateandtime.py::TestMatchSpecialCase::test_match_special_case_3rd_variation	0.00	
Passed (show details)	tests/unit_test_dateandtime.py::TestMatchSpecialCase::test_match_special_case_4th_variation	0.00	
Passed (show details)	tests/unit_test_dateandtime.py::TestMatchSpecialCase::test_match_special_case_with_invalid_argument	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestMimeTypes::test_magic_type_lookup_arbitrarily_assert_image_is_jpeg	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestMimeTypes::test_magic_type_lookup_table_exists	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestMimeTypes::test_magic_type_lookup_table_is_instance_of_dict	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestMimeTypes::test_magic_type_lookup_table_values_are_type_list	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestSplitFileName::test_split_filename_has_multiple_extensions	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestSplitFileName::test_split_filename_has_no_extension	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestSplitFileName::test_split_filename_has_one_extension	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestSplitFileName::test_split_filename_no_name	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestFileSuffix::test_file_suffix_file_has_many_suffixes	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestFileSuffix::test_file_suffix_file_has_no_extension	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestFileSuffix::test_file_suffix_file_has_one_extension	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestFileSuffix::test_file_suffix_from_hidden_file	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestFileSuffix::test_file_suffix_from_hidden_file_many_suffixes	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestFileSuffix::test_file_suffix_no_name	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestFileBase::test_file_base_file_has_many_suffixes	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestFileBase::test_file_base_file_has_no_extension	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestFileBase::test_file_base_file_has_one_extension	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestFileBase::test_file_base_from_hidden_file	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestFileBase::test_file_base_from_hidden_file_many_suffixes	0.00	
Passed (show details)	tests/unit_test_diskutils.py::TestFileBase::test_file_base_no_name	0.00	
Passed (show details)	tests/unit_test_fileobject.py::TestFileTypeMagic::test_filetype_magic	0.06	
Passed (show details)	tests/unit_test_fileobject.py::TestFileTypeMagic::test_filetype_magic_with_invalid_args	0.00	
Passed (show details)	tests/unit_test_fileobject.py::TestFileTypeMagic::test_test_files_are_readable	0.00	

Passed (show details)	tests/unit_test_fileobject.py::TestFileTypeMagic::test_test_files_defined	0.00	Links
Passed (show details)	tests/unit_test_fileobject.py::TestFileTypeMagic::test_test_files_exist	0.00	
Passed (show details)	tests/unit_test_filerule.py::TestFileRuleInstantiation::test_init_description_is_str_or_none	0.00	
Passed (show details)	tests/unit_test_filerule.py::TestFileRuleInstantiation::test_init_exact_match_is_boolean	0.00	
Passed (show details)	tests/unit_test_filerule.py::TestFileRuleMethods::test_filerule_string	0.00	
Passed (show details)	tests/unit_test_misc_utils.py::TestUniqueIdentifier::test_unique_identifier_has_prefix_uuid	0.00	
Passed (show details)	tests/unit_test_misc_utils.py::TestUniqueIdentifier::test_unique_identifier_returns_1000_unique_values	0.00	
Passed (show details)	tests/unit_test_misc_utils.py::TestUniqueIdentifier::test_unique_identifier_returns_100_unique_values	0.00	
Passed (show details)	tests/unit_test_misc_utils.py::TestUniqueIdentifier::test_unique_identifier_returns_not_none	0.00	
Passed (show details)	tests/unit_test_misc_utils.py::TestUniqueIdentifier::test_unique_identifier_returns_string	0.00	
Passed (show details)	tests/unit_test_namebuilder.py::TestNameBuilder::test_assemble_basename_using_default_name_template	0.00	
Passed (show details)	tests/unit_test_namebuilder.py::TestNameBuilder::test_assemble_basename_using_template_1_given_all_fields	0.00	
Passed (show details)	tests/unit_test_namebuilder.py::TestNameBuilder::test_assemble_basename_using_template_1_some_fields_missing	0.00	
Passed (show details)	tests/unit_test_namebuilder.py::TestNameBuilder::test_assemble_basename_using_template_2_all_fields_missing	0.00	
Passed (show details)	tests/unit_test_namebuilder.py::TestNameBuilder::test_assemble_basename_using_template_2_given_all_fields	0.00	
Passed (show details)	tests/unit_test_options.py::TestArgumentValidators::test_arg_is_readable_file_raises_exception_arg_is_dev_null	0.00	
Passed (show details)	tests/unit_test_options.py::TestArgumentValidators::test_arg_is_readable_file_raises_exception_arg_is_directory	0.00	
Passed (show details)	tests/unit_test_options.py::TestArgumentValidators::test_arg_is_readable_file_raises_exception_empty_string_arg	0.00	
Passed (show details)	tests/unit_test_options.py::TestArgumentValidators::test_arg_is_readable_file_raises_exception_missing_file	0.00	
Passed (show details)	tests/unit_test_options.py::TestArgumentValidators::test_arg_is_readable_file_raises_exception_none_argument	0.00	
Passed (show details)	tests/unit_test_options.py::TestArgumentValidators::test_arg_is_year_passes_valid_years	0.00	
Passed (show details)	tests/unit_test_options.py::TestArgumentValidators::test_arg_is_year_raises_exception	0.00	
Passed (show details)	tests/unit_test_rulematcher.py::TestRuleMatcher::test_setup	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestRuleParserFunctions::test_get_available_parsers	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestRuleParserFunctions::test_get_available_parsers_returns_list_of_strings	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestRuleParserFunctions::test_get_instantiated_parsers_returns_arbitrary_number	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestRuleParserFunctions::test_get_instantiated_parsers_returns_class_objects	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestRuleParserFunctions::test_get_instantiated_parsers_returns_list	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestRuleParser::test_get_validation_function_should_raise_error_if_unimplemented	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestRuleParserSubclasses::test_get_validation_function_should_not_return_none	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestRuleParserSubclasses::test_get_validation_function_should_return_function	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestRuleParserSubclasses::test_setup	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestRuleParserSubclasses::test_validation_function_should_return_booleans	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestRegexRuleParser::test_validation_function_expect_fail	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestRegexRuleParser::test_validation_function_expect_pass	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestMimeTypeRuleParser::test_validation_function_expect_fail	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestMimeTypeRuleParser::test_validation_function_expect_pass	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestDateTimeRuleParser::test_validation_function_expect_fail	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestDateTimeRuleParser::test_validation_function_expect_pass	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestNameFormatRuleParser::test_validation_function_expect_fail	0.00	
Passed (show details)	tests/unit_test_ruleparsers.py::TestNameFormatRuleParser::test_validation_function_expect_pass	0.00	
Passed (show details)	tests/unit_test_unit_utils.py::TestUtilities::test_make_temp_dir	0.00	