

Домашнее задание по конфигурационному управлению №3  
Документация для проекта

Создано системой Doxygen 1.12.0

# Глава 1

## Алфавитный указатель пространств имен

### 1.1 Пространства имен

Полный список пространств имен.

<code>main</code>	.....	??
-------------------	-------	----

## Глава 2

# Иерархический список классов

### 2.1 Иерархия классов

Иерархия классов.

Exception	
main.SyntaxError	...
	??

## Глава 3

# Алфавитный указатель классов

### 3.1 Классы

Классы с их кратким описанием.

`main.SyntaxError` . . . . . ??

## Глава 4

# Список файлов

### 4.1 Файлы

Полный список файлов.

[main.py](#) . . . . . ??

## Глава 5

# Пространства имен

### 5.1 Пространство имен main

#### Классы

- class [SyntaxError](#)

#### Функции

- [validate\\_name](#) (name)
- [evaluate\\_postfix](#) (expression, constants)
- [convert\\_toml\\_to\\_custom](#) (data)
- [main](#) ()

#### 5.1.1 Функции

##### 5.1.1.1 `convert_toml_to_custom()`

```
main.convert_toml_to_custom (  
    data)
```

Преобразует TOML-данные в пользовательский конфигурационный язык.

Args:

data (dict): Словарь с данными TOML.

Returns:

str: Строка с преобразованными данными.

Raises:

SyntaxError: Если встречены некорректные данные.

#### 5.1.1.2 evaluate\_postfix()

```
main.evaluate_postfix (
    expression,
    constants)
```

Вычисляет значение постфиксного выражения.

Args:  
expression (list): Список токенов постфиксного выражения.  
constants (dict): Словарь доступных констант.

Returns:  
float: Результат вычисления выражения.

Raises:  
SyntaxError: Если встречен некорректный токен.

#### 5.1.1.3 main()

```
main.main ()
```

Основная функция скрипта. Обрабатывает входной файл TOML и записывает результат в выходной файл.

Args:  
-i (str): Путь к входному файлу TOML.  
-o (str): Путь к выходному файлу.

Raises:  
SyntaxError: Если входные данные содержат ошибки синтаксиса.  
Exception: Для любых других ошибок.

#### 5.1.1.4 validate\_name()

```
main.validate_name (
    name)
```

Проверяет корректность имени в соответствии с правилами.

Args:  
name (str): Имя для проверки.

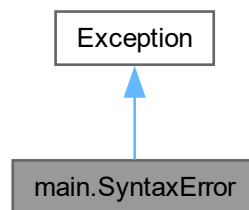
Raises:  
SyntaxError: Если имя некорректно.

## Глава 6

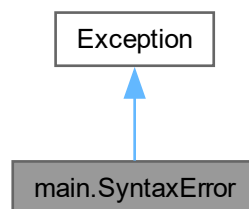
# Классы

### 6.1 Класс `main.SyntaxError`

Граф наследования: `main.SyntaxError`:



Граф связей класса `main.SyntaxError`:





### 6.1.1 Подробное описание

Исключение, вызываемое при синтаксических ошибках в данных.

Args:

`message (str)`: Сообщение об ошибке.

Объявления и описания членов класса находятся в файле:

- [main.py](#)

# Глава 7

## Файлы

### 7.1 Файл `main.py`

#### Классы

- class `main.SyntaxError`

#### Пространства имен

- namespace `main`

#### Функции

- `main.validate_name` (name)
- `main.evaluate_postfix` (expression, constants)
- `main.convert_toml_to_custom` (data)
- `main.main` ()