

Team_Not_Found_Calc

Generated by Doxygen 1.13.2

1 Namespace Index	1
1.1 Namespace List	1
2 File Index	3
2.1 File List	3
3 Namespace Documentation	5
3.1 calc_lib Namespace Reference	5
3.1.1 Detailed Description	5
3.1.2 Function Documentation	5
3.1.2.1 absolute()	5
3.1.2.2 add()	6
3.1.2.3 div()	6
3.1.2.4 expon()	6
3.1.2.5 factorial()	6
3.1.2.6 fib()	6
3.1.2.7 mul()	7
3.1.2.8 sqr()	7
3.1.2.9 sub()	7
3.2 profiling Namespace Reference	7
3.2.1 Detailed Description	8
3.2.2 Variable Documentation	8
3.2.2.1 count	8
3.2.2.2 numCount	8
3.2.2.3 nums	8
3.2.2.4 prum	8
3.2.2.5 prumExp	8
4 File Documentation	9
4.1 calc_lib.py File Reference	9
4.2 profiling.py File Reference	9
Index	11

Chapter 1

Namespace Index

1.1 Namespace List

Here is a list of all namespaces with brief descriptions:

calc_lib	5
profiling	7

Chapter 2

File Index

2.1 File List

Here is a list of all files with brief descriptions:

calc_lib.py	9
profiling.py	9

Chapter 3

Namespace Documentation

3.1 calc_lib Namespace Reference

Functions

- [add](#) (num1, num2)
- [sub](#) (num1, num2)
- [mul](#) (num1, num2)
- [div](#) (num1, num2)
- [factorial](#) (num1)
- [expon](#) (num1, num2)
- [sqr](#) (num1, num2)
- [absolute](#) (num1)
- [fib](#) (num1)

3.1.1 Detailed Description

```
@file calc_lib.py
@brief knihovna matematických funkcí
@details jsou volány funkce které načítají jedno nebo dvě čísla a ověřuje jejich integritu s tím že vrací
@author Adam Kadlec
```

3.1.2 Function Documentation

3.1.2.1 absolute()

```
calc_lib.absolute (
    num1)
```

```
@brief Vrábí absolutní hodnotu čísla.
@param num1 Číslo (int nebo float).
@return Vrací absolutní hodnotu, nebo chybovou hlášku (float nebo str).
```

3.1.2.2 add()

```
calc_lib.add (  
    num1,  
    num2)  
  
@brief Sčítá dvě čísla.  
@param num1 První číslo (int nebo float).  
@param num2 Druhé číslo (int nebo float).  
@return Vrací součet dvou čísel, nebo chybovou hlášku (int, float nebo str).
```

3.1.2.3 div()

```
calc_lib.div (  
    num1,  
    num2)  
  
@brief Dělí dvě čísla.  
@param num1 Dělenec (int nebo float).  
@param num2 Dělitel (int nebo float).  
@return Vrací podíl dvou čísel, nebo chybovou hlášku (float nebo str).
```

3.1.2.4 expon()

```
calc_lib.expon (  
    num1,  
    num2)  
  
@brief Umocní číslo.  
@param num1 Základ (int nebo float).  
@param num2 Exponent (int nebo float).  
@return Vrací výsledek umocnění, nebo chybovou hlášku (float nebo str).
```

3.1.2.5 factorial()

```
calc_lib.factorial (  
    num1)  
  
@brief Vypočítá faktoriál čísla.  
@param num1 Celé číslo (int).  
@return Vrací faktoriál čísla, nebo chybovou hlášku (int nebo str).
```

3.1.2.6 fib()

```
calc_lib.fib (  
    num1)  
  
@brief Vypočítá Fibonacciho číslo na dané pozici.  
@param num1 Pořadí v posloupnosti (int).  
@return Vrací číslo z Fibonacciho posloupnosti, nebo chybovou hlášku (int nebo str).
```

3.1.2.7 mul()

```
calc_lib.mul (  
    num1,  
    num2)  
  
@brief Násobí dvě čísla.  
@param num1 První číslo (int nebo float).  
@param num2 Druhé číslo (int nebo float).  
@return Vrací součin dvou čísel, nebo chybovou hlášku (int, float nebo str).
```

3.1.2.8 sqr()

```
calc_lib.sqr (  
    num1,  
    num2)  
  
@brief Vypočítá odmocninu čísla.  
@param num1 Odmocňované číslo (int nebo float).  
@param num2 Kolikátá odmocnina (int nebo float).  
@return Vrací výsledek odmocnění, nebo chybovou hlášku (float nebo str).
```

3.1.2.9 sub()

```
calc_lib.sub (  
    num1,  
    num2)  
  
@brief Odečítá dvě čísla.  
@param num1 První číslo (int nebo float).  
@param num2 Druhé číslo (int nebo float).  
@return Vrací rozdíl dvou čísel, nebo chybovou hlášku (int, float nebo str).
```

3.2 profiling Namespace Reference

Variables

- list `nums` = []
- int `numCount` = 0
- int `count` = 0
- `prum` = `div(numCount,count)`
- int `prumExp` = 0

3.2.1 Detailed Description

```
@file profiling.py
@author Adam Kadlec
@brief Výpočet výběrové směrodatné odchylky ze standardního vstupu pomocí vlastní matematické knihovny.
@details Program načítá čísla ze standardního vstupu, vypočítá aritmetický průměr a následně výběrovou směrodatnou odchylku.

@section usage Usage
Spuštění programu s čísly zadanými na standardním vstupu. Program načítá čísla, počítá průměr a následně výběrovou směrodatnou odchylku.

@param nums Seznam čísel, která se načítají ze vstupu.
@param numCount Součet všech čísel načtených ze vstupu.
@param count Počet načtených čísel.

@return Vrací výběrovou směrodatnou odchylku pomocí výpisu na stdout.
```

3.2.2 Variable Documentation

3.2.2.1 count

```
profiling.count = 0
```

3.2.2.2 numCount

```
profiling.numCount = 0
```

3.2.2.3 nums

```
list profiling.nums = []
```

3.2.2.4 prum

```
profiling.prum = div(numCount, count)
```

3.2.2.5 prumExp

```
profiling.prumExp = 0
```


Chapter 4

File Documentation

4.1 `calc_lib.py` File Reference

Namespaces

- namespace `calc_lib`

Functions

- `calc_lib.add` (num1, num2)
- `calc_lib.sub` (num1, num2)
- `calc_lib.mul` (num1, num2)
- `calc_lib.div` (num1, num2)
- `calc_lib.factorial` (num1)
- `calc_lib.expon` (num1, num2)
- `calc_lib.sqr` (num1, num2)
- `calc_lib.absolute` (num1)
- `calc_lib.fib` (num1)

4.2 `profiling.py` File Reference

Namespaces

- namespace `profiling`

Variables

- list `profiling.nums` = []
- int `profiling.numCount` = 0
- int `profiling.count` = 0
- `profiling.prum` = `div(numCount, count)`
- int `profiling.prumExp` = 0

Index

- absolute
 - [calc_lib](#), [5](#)
- add
 - [calc_lib](#), [5](#)
- [calc_lib](#), [5](#)
 - [absolute](#), [5](#)
 - [add](#), [5](#)
 - [div](#), [6](#)
 - [expon](#), [6](#)
 - [factorial](#), [6](#)
 - [fib](#), [6](#)
 - [mul](#), [6](#)
 - [sqr](#), [7](#)
 - [sub](#), [7](#)
- [calc_lib.py](#), [9](#)
- count
 - [profiling](#), [8](#)
- div
 - [calc_lib](#), [6](#)
- expon
 - [calc_lib](#), [6](#)
- factorial
 - [calc_lib](#), [6](#)
- fib
 - [calc_lib](#), [6](#)
- mul
 - [calc_lib](#), [6](#)
- numCount
 - [profiling](#), [8](#)
- nums
 - [profiling](#), [8](#)
- [profiling](#), [7](#)
 - [count](#), [8](#)
 - [numCount](#), [8](#)
 - [nums](#), [8](#)
 - [prum](#), [8](#)
 - [prumExp](#), [8](#)
- [profiling.py](#), [9](#)
- prum
 - [profiling](#), [8](#)
- prumExp
 - [profiling](#), [8](#)
- sqr

- [calc_lib](#), [7](#)
- sub
 - [calc_lib](#), [7](#)