

Lorem Ipsum Calculator

Generováno programem Doxygen 1.8.13

Obsah

1	Rejstřík hierarchie tříd	1
1.1	Hierarchie tříd	1
2	Rejstřík tříd	3
2.1	Seznam tříd	3
3	Dokumentace tříd	5
3.1	Dokumentace třídy <code>pckg.Calc_Lib</code>	5
3.1.1	Detailní popis	5
3.1.2	Dokumentace k metodám	5
3.1.2.1	<code>add()</code>	5
3.1.2.2	<code>divide()</code>	6
3.1.2.3	<code>factorial()</code>	6
3.1.2.4	<code>mult()</code>	7
3.1.2.5	<code>power()</code>	7
3.1.2.6	<code>root()</code>	8
3.1.2.7	<code>st_Dev()</code>	8
3.1.2.8	<code>sub()</code>	9
3.2	Dokumentace třídy <code>pckg.Calc_LibTest</code>	9
3.3	Dokumentace třídy <code>pckg.CalcActionListener</code>	10
3.3.1	Detailní popis	10
3.3.2	Dokumentace konstruktoru a destrukturu	10
3.3.2.1	<code>CalcActionListener()</code>	10
3.3.3	Dokumentace k metodám	11
3.3.3.1	<code>actionPerformed()</code>	11
3.4	Dokumentace třídy <code>pckg.Main_Window.ComponentContainer</code>	11
3.4.1	Detailní popis	12
3.5	Dokumentace třídy <code>pckg.Main_Window</code>	12
3.5.1	Detailní popis	13
3.5.2	Dokumentace konstruktoru a destrukturu	13
3.5.2.1	<code>Main_Window()</code>	13
3.5.3	Dokumentace k metodám	13
3.5.3.1	<code>infoBox()</code>	13
3.5.3.2	<code>main()</code>	13

Rejstřík	15
----------	----

Kapitola 1

Rejstřík hierarchie tříd

1.1 Hierarchie tříd

Zde naleznete seznam, vyjadřující vztah dědičnosti tříd. Je seřazen přibližně (ale ne úplně) podle abecedy:

AbstractAction	
pkg.CalcActionListener	10
pkg.Calc_Lib	5
pkg.Calc_LibTest	9
pkg.Main_Window.ComponentContainer	11
JFrame	
pkg.Main_Window	12

Kapitola 2

Rejstřík tříd

2.1 Seznam tříd

Následující seznam obsahuje především identifikace tříd, ale nacházejí se zde i další netriviální prvky, jako jsou struktury (struct), unie (union) a rozhraní (interface). V seznamu jsou uvedeny jejich stručné popisy:

pkg.Calc_Lib	5
pkg.Calc_LibTest	9
pkg.CalcActionListener Třída pro zpracování signálů	10
pkg.Main_Window.ComponentContainer Třída pro předání tlačítek action listeneru	11
pkg.Main_Window Třída pro GUI	12

Kapitola 3

Dokumentace tříd

3.1 Dokumentace třídy pkg.Calc_Lib

Statické veřejné metody

- static double `add` (double x, double y) throws Exception
- static double `sub` (double x, double y) throws Exception
- static double `mult` (double x, double y) throws Exception
- static double `divide` (double x, double y) throws Exception
- static long `factorial` (long x) throws Exception
- static double `power` (double x, double n) throws Exception
- static double `root` (double x, double n) throws Exception
- static double `st_Dev` (String filename) throws Exception

3.1.1 Detailní popis

Matematická knihovna obsahující funkce pro výpočet: součtu, rozdílu, součinu, podílu, faktoriálu, mocniny, odmocniny a směrodatné odchylky. V případě chyby vyvolá každá z funkcí výjimku.

3.1.2 Dokumentace k metodám

3.1.2.1 `add()`

```
static double pkg.Calc_Lib.add (  
    double x,  
    double y ) throws Exception [static]
```

Provádí součet dvou reálných čísel. V případě přetečení vyvolá výjimku.

Parametry

<i>x</i>	double
<i>y</i>	double

Návratová hodnota

Výsledek po sečtení vstupních parametru.

Výjimky

<i>Exception</i>	(Add overflow)
------------------	----------------

3.1.2.2 divide()

```
static double pkg.Calc_Lib.divide (  
    double x,  
    double y ) throws Exception [static]
```

Provádí výpočet podílu dvou čísel. Pokud je dělitel 0 vyvolá výjimku

Parametry

<i>x</i>	- dělenec
<i>y</i>	- dělitel

Návratová hodnota

výsledek po násobení

Výjimky

<i>Exception</i>	(Dividing by zero)
------------------	--------------------

3.1.2.3 factorial()

```
static long pkg.Calc_Lib.factorial (  
    long x ) throws Exception [static]
```

Provádí výpočet faktoriálu. Pokud je číslo 'x' záporné nebo dojde k přetecení vyvolá výjimku.

Parametry

<i>x</i>	- číslo pro výpočet faktoriálu
----------	--------------------------------

Návratová hodnota

hodnotu faktoriálu

Výjimky

<i>Exception</i>	(Factorial from negative number) (Factorial overflow)
------------------	--

3.1.2.4 `mult()`

```
static double pckg.Calc_Lib.mult (
    double x,
    double y ) throws Exception [static]
```

Provádí násobení dvou reálných čísel. V případě přetečení vyvolá výjimku.

Parametry

<i>x</i>	double
<i>y</i>	double

Návratová hodnota

Výsledek po násobení dvou vstupních parametru.

Výjimky

<i>Exception</i>	(Mult overflow)
------------------	-----------------

3.1.2.5 `power()`

```
static double pckg.Calc_Lib.power (
    double x,
    double n ) throws Exception [static]
```

Provádí výpočet mocniny pomocí Taylorova polynomu. Pokud je základ záporné číslo, provede se kontrola, jestli je exponent sudé nebo liché číslo. Pokud je exponent sudý, základ je vynásoben -1 a následně je proveden výpočet. Pokud je exponent liché aplikuje se stejný postup a na závěr je výsledek vynásoben -1. Pokud exponent není celé číslo, je výsledek roven hodnotě NaN.

Parametry

<i>x</i>	- základ
<i>n</i>	- exponent

Návratová hodnota

umocnene cislo

Výjimky

<i>Exception</i>	(Power overflow)
------------------	------------------

3.1.2.6 root()

```
static double pkg.Calc_Lib.root (
    double x,
    double n ) throws Exception [static]
```

Provádí výpočet odmocniny $n^{\sqrt{x}}$. Jestliže je $x < 0$ nebo $n == 0$ vyvolá výjimku.

Parametry

<i>x</i>	- základ
<i>n</i>	- stupeň odmocniny

Návratová hodnota

výsledek po odmocnování

Výjimky

<i>Exception</i>	(Root from negative number) (Can't count zero root)
------------------	--

3.1.2.7 st_Dev()

```
static double pkg.Calc_Lib.st_Dev (
    String filename ) throws Exception [static]
```

Vypočítá směrodatnou odchylku z hodnot, které se nacházejí v souboru zadaném cestou 'filename'.

Čísla uvnitř souboru jsou ve formátu: číslo1 'whitespace chars' číslo2 'whitespace chars'... číslo n

Směrodatnou odchylku počítá podle vzorečku: $((\sum_{i=0}^n (x_i - \bar{x})^2) / (n-1))^{0.5}$

Parametry

<i>filename</i>	- Cesta k souboru s čísly
-----------------	---------------------------

Návratová hodnota

hodnota směrodatné odchylky

Výjimky

<i>Exception</i>	
------------------	--

3.1.2.8 sub()

```
static double pckg.Calc_Lib.sub (  
    double x,  
    double y ) throws Exception [static]
```

Provádí rozdíl dvou reálných čísel. V případě přetečení vyvolá výjimku.

Parametry

<i>x</i>	double
<i>y</i>	double

Návratová hodnota

Výsledný rozdíl dvou vstupních parametrů.

Výjimky

<i>Exception</i>	(Sub overflow)
------------------	----------------

Dokumentace pro tuto třídu byla generována z následujícího souboru:

- pckg/Calc_Lib.java

3.2 Dokumentace třídy pckg.Calc_LibTest

Veřejné metody

- void **add** () throws Exception
- void **sub** () throws Exception
- void **mult** () throws Exception
- void **divide** () throws Exception
- void **factorial** () throws Exception
- void **power** () throws Exception
- void **root** () throws Exception
- void **st_Dev** () throws Exception

Veřejné atributy

- `ExpectedException thrown = ExpectedException.none()`

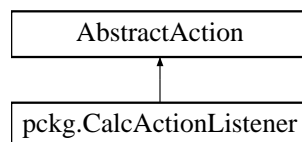
Dokumentace pro tuto třídu byla generována z následujícího souboru:

- `pckg/Calc_LibTest.java`

3.3 Dokumentace třídy `pckg.CalcActionListener`

Třída pro zpracování signálů

Diagram dědičnosti pro třídu `pckg.CalcActionListener`



Veřejné metody

- `CalcActionListener` (`Main_Window.ComponentContainer container`)
- `void actionPerformed` (`ActionEvent e`)

3.3.1 Detailní popis

Třída pro zpracování signálů

Zpracovává signály od GUI a volá patřičné metody.

3.3.2 Dokumentace konstruktoru a destruktoru

3.3.2.1 `CalcActionListener()`

```
pckg.CalcActionListener.CalcActionListener (
    Main_Window.ComponentContainer container )
```

Konstruktor

Parametry

<code>Main_Window.ComponentContainer</code>	předán container s graf. prvky
---	--------------------------------

3.3.3 Dokumentace k metodám

3.3.3.1 actionPerformed()

```
void pkg.CalcActionListener.actionPerformed (
   (ActionEvent e )
```

Odchytávač signálů volá příslušné funkce tlačítek

Parametry

<i>ActionEvent</i>	předán ActionEvent
--------------------	--------------------

Dokumentace pro tuto třídu byla generována z následujícího souboru:

- pkg/CalcActionListener.java

3.4 Dokumentace třídy pkg.Main_Window.ComponentContainer

Třída pro předání tlačítek action listeneru.

Veřejné atributy

- JButton **btnrov**
- JButton **btnplus**
- JButton **btn0**
- JButton **btncar**
- JButton **btnodch**
- JButton **btnminus**
- JButton **btnC**
- JButton **btn2**
- JButton **btn1**
- JButton **btn3**
- JButton **btn6**
- JButton **btn5**
- JButton **btn4**
- JButton **btn7**
- JButton **btn8**
- JButton **btn9**
- JButton **btnDel**
- JButton **btnodm2**
- JButton **btnmoc2**
- JButton **btnmocn**
- JButton **btnodmocn**
- JButton **btnAns**
- JButton **btnfakt**

- JButton **btndeleno**
- JButton **btnkrat**
- JButton **btnplmi**
- JTextField **priklad**
- JTextField **vysledek**
- JFileChooser **fc**
- JButton **btnHelp**

3.4.1 Detailní popis

Třída pro předání tlačítek action listeneru.

Rato třída slouží pouze jako container pro předání.

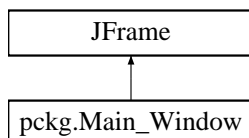
Dokumentace pro tuto třídu byla generována z následujícího souboru:

- pckg/Main_Window.java

3.5 Dokumentace třídy pckg.Main_Window

Třída pro GUI.

Diagram dědičnosti pro třídu pckg.Main_Window



Třídy

- class [ComponentContainer](#)

Třída pro předání tlačítek action listeneru.

Veřejné metody

- [Main_Window](#) ()

Statické veřejné metody

- static void [main](#) (String[] args)
- static void [infoBox](#) (String zprava)

3.5.1 Detailní popis

Třída pro GUI.

Třída řídící GUI a jeho signály.

3.5.2 Dokumentace konstruktoru a destruktoru

3.5.2.1 `Main_Window()`

```
pckg.Main_Window.Main_Window ( )
```

Vytvoření frame a inicializace signálů

3.5.3 Dokumentace k metodám

3.5.3.1 `infoBox()`

```
static void pckg.Main_Window.infoBox (
    String zprava ) [static]
```

Funkce vytvářející Dialog okno pro nápovědu

Parametry

<code>zprava</code>	zprava která bude zobrazena v nápovědě
---------------------	--

3.5.3.2 `main()`

```
static void pckg.Main_Window.main (
    String [] args ) [static]
```

Spuštění aplikace

Dokumentace pro tuto třídu byla generována z následujícího souboru:

- `pckg/Main_Window.java`

Rejstřík

actionPerformed
 pckg::CalcActionListener, [11](#)
add
 pckg::Calc_Lib, [5](#)

CalcActionListener
 pckg::CalcActionListener, [10](#)

divide
 pckg::Calc_Lib, [6](#)

factorial
 pckg::Calc_Lib, [6](#)

infoBox
 pckg::Main_Window, [13](#)

main
 pckg::Main_Window, [13](#)

Main_Window
 pckg::Main_Window, [13](#)

mult
 pckg::Calc_Lib, [7](#)

pckg.Calc_Lib, [5](#)
pckg.Calc_LibTest, [9](#)
pckg.CalcActionListener, [10](#)
pckg.Main_Window, [12](#)
pckg.Main_Window.ComponentContainer, [11](#)
pckg::Calc_Lib
 add, [5](#)
 divide, [6](#)
 factorial, [6](#)
 mult, [7](#)
 power, [7](#)
 root, [8](#)
 st_Dev, [8](#)
 sub, [9](#)

pckg::CalcActionListener
 actionPerformed, [11](#)
 CalcActionListener, [10](#)

pckg::Main_Window
 infoBox, [13](#)
 main, [13](#)
 Main_Window, [13](#)

power
 pckg::Calc_Lib, [7](#)

root
 pckg::Calc_Lib, [8](#)

st_Dev
 pckg::Calc_Lib, [8](#)
sub
 pckg::Calc_Lib, [9](#)