

My Project

Generated by Doxygen 1.9.4

1 Class Index	1
1.1 Class List	1
2 File Index	3
2.1 File List	3
3 Class Documentation	5
3.1 Part Struct Reference	5
3.1.1 Detailed Description	5
4 File Documentation	7
4.1 input.h File Reference	7
4.1.1 Detailed Description	7
4.1.2 Function Documentation	7
4.1.2.1 getCount()	7
4.1.2.2 getId()	8
4.1.2.3 getName()	8
4.1.2.4 getString()	9
4.1.2.5 getUnsgnInt()	9
4.1.2.6 inputString()	9
4.1.2.7 inputStruct()	10
4.2 input.h	10
4.3 Part.h File Reference	10
4.4 Part.h	11
4.5 process.h File Reference	11
4.5.1 Detailed Description	11
4.5.2 Function Documentation	11
4.5.2.1 stringToStruct()	11
4.5.2.2 structToString()	12
4.6 process.h	12
Index	13

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

[Part](#)

This structure is designated to hold all necessary parameters for production of different parts . [5](#)

Chapter 2

File Index

2.1 File List

Here is a list of all documented files with brief descriptions:

input.h	File where input functions are defined	7
Part.h	This file contains description of the structure Part and functions of this structure	10
process.h	In this file are described functions that process given structures or strings	11

Chapter 3

Class Documentation

3.1 Part Struct Reference

This structure is designated to hold all necessary parameters for production of different parts.

```
#include <Part.h>
```

Public Attributes

- `std::string id`
- `std::string name`
- `unsigned int count {}`

3.1.1 Detailed Description

This structure is designated to hold all necessary parameters for production of different parts.

The documentation for this struct was generated from the following file:

- [Part.h](#)

Chapter 4

File Documentation

4.1 input.h File Reference

File where input functions are defined.

```
#include <iostream>
```

```
#include "Part.h"
```

Include dependency graph for input.h:

Functions

- unsigned int [input::getUnsgnInt](#) ()
Function is designated to get an unsigned integer.
- unsigned int [input::getCount](#) ()
Function is designated to get an unsigned integer for count field of structure [Part](#).
- std::string [input::getString](#) ()
Function is designated to get a string.
- std::string [input::getId](#) ()
Function is designated to get a string for id field of structure [Part](#).
- std::string [input::getName](#) ()
Function is designated to get a string for name field of structure [Part](#).
- [Part](#) [input::inputStruct](#) ()
Function is designated to get a string for id field of structure [Part](#).
- std::string [input::inputString](#) ()
Function is designated to get a .json string.

4.1.1 Detailed Description

File where input functions are defined.

4.1.2 Function Documentation

4.1.2.1 getCount()

```
unsigned int input::getCount ( )
```

Function is designated to get an unsigned integer for count field of structure [Part](#).

Exceptions

<i>std::runtime_error</i>	in case of EOF
<i>std::runtime_error</i>	in case of bad input

Returns

unsigned int

4.1.2.2 getId()

```
std::string input::getId ( )
```

Function is designated to get a string for id field of structure [Part](#).

Exceptions

<i>std::runtime_error</i>	in case of EOF
<i>std::runtime_error</i>	in case of bad input

Returns

std::string

4.1.2.3 getName()

```
std::string input::getName ( )
```

Function is designated to get a string for name field of structure [Part](#).

Exceptions

<i>std::runtime_error</i>	in case of EOF
<i>std::runtime_error</i>	in case of bad input

Returns

std::string

4.1.2.4 getString()

```
std::string input::getString ( )
```

Function is designated to get a string.

Exceptions

<i>std::runtime_error</i>	in case of EOF
<i>std::runtime_error</i>	in case of bad input

Returns

std::string

4.1.2.5 getUnsgnInt()

```
unsigned int input::getUnsgnInt ( )
```

Function is designated to get an unsigned integer.

Exceptions

<i>std::runtime_error</i>	in case of EOF
<i>std::runtime_error</i>	is case of bad input

Returns

unsinged int

4.1.2.6 inputString()

```
std::string input::inputString ( )
```

Function is designated to get a .json string.

Exceptions

<i>std::runtime_error</i>	in case of EOF
<i>std::runtime_error</i>	in case of bad input

Returns

std::string

4.1.2.7 inputStruct()

```
Part input::inputStruct ( )
```

Function is designated to get a string for id field of structure [Part](#).

Exceptions

<i>std::runtime_error</i>	in case of EOF in one of the fields of the structure
<i>std::runtime_error</i>	in case of bad input in one of the fields of the structure

Returns

[Part](#)

4.2 input.h

[Go to the documentation of this file.](#)

```
1 //
2 // Created by ionik on 07.09.2024.
3 //
4
10 #ifndef LAB1_INPUT_H
11 #define LAB1_INPUT_H
12
13 #include <iostream>
14 #include "Part.h"
15
16 namespace input {
23     unsigned int getUnsgnInt();
24
31     unsigned int getCount();
32
39     std::string getString();
40
47     std::string getId();
48
55     std::string getName();
56
63     Part inputStruct();
64
71     std::string inputString();
72 }
73
74 #endif //LAB1_INPUT_H
```

4.3 Part.h File Reference

This file contains description of the structure [Part](#) and functions of this structure.

```
#include <iostream>
```

Include dependency graph for Part.h:

4.4 Part.h

[Go to the documentation of this file.](#)

```
1 //
2 // Created by ionik on 06.09.2024.
3 //
4
10 #ifndef LAB1_PART_H
11 #define LAB1_PART_H
12
13 #include <iostream>
14
18 struct Part {
19     std::string id;
20     std::string name;
21     unsigned int count{};
22 };
23
31 Part create(std::string _id, std::string _name, unsigned int _count);
32
40 bool compare(Part& p1, Part& p2);
41
42 #endif //LAB1_PART_H
```

4.5 process.h File Reference

In this file are described functions that process given structures or strings.

```
#include <iostream>
#include "Part.h"
```

Include dependency graph for process.h:

Functions

- `std::string process::structToString` (const `Part` &part)
This function transforms structure `Part` to .json string.
- `Part process::stringToStruct` (const std::string &json)
This function transforms .json string to structure `Part`.

4.5.1 Detailed Description

In this file are described functions that process given structures or strings.

4.5.2 Function Documentation

4.5.2.1 stringToStruct()

```
Part process::stringToStruct (
    const std::string & json )
```

This function transforms .json string to structure `Part`.

Parameters

<i>part</i>	Example of .json string that will be transformed
-------------	--

Returns

std::string A tructure [Part](#) based on given .json string

4.5.2.2 structToString()

```
std::string process::structToString (
    const Part & part )
```

This function transforms structure [Part](#) to .json string.

Parameters

<i>part</i>	Example of structure Part that will be transformed
-------------	--

Returns

std::string A .json string based on given structure [Part](#)

4.6 process.h

[Go to the documentation of this file.](#)

```
1 //
2 // Created by ionik on 09.09.2024.
3 //
4
10 #ifndef LAB1_PROCESS_H
11 #define LAB1_PROCESS_H
12
13 #include <iostream>
14 #include "Part.h"
15
16 namespace process {
22     std::string structToString(const Part& part);
23
29     Part stringToStruct(const std::string& json);
30 }
31
32 #endif //LAB1_PROCESS_H
```


Index

- getCount
 - [input.h, 7](#)
- getId
 - [input.h, 8](#)
- getName
 - [input.h, 8](#)
- getString
 - [input.h, 8](#)
- getUnsgnInt
 - [input.h, 9](#)
- [input.h, 7](#)
 - [getCount, 7](#)
 - [getId, 8](#)
 - [getName, 8](#)
 - [getString, 8](#)
 - [getUnsgnInt, 9](#)
 - [inputString, 9](#)
 - [inputStruct, 10](#)
- inputString
 - [input.h, 9](#)
- inputStruct
 - [input.h, 10](#)
- [Part, 5](#)
- [Part.h, 10](#)
- [process.h, 11](#)
 - [stringToStruct, 11](#)
 - [structToString, 12](#)
- stringToStruct
 - [process.h, 11](#)
- structToString
 - [process.h, 12](#)