



Class ArgumentSetter

Table of contents

Class Documentation

- Defined in [File argument_setter.hpp](#)

Class Documentation

class ArgumentSetter

Class to set arguments for components.

This class is used to set arguments for components (including [Operator](#), [Condition](#), and [Resource](#)) from [Arg](#) or [ArgList](#).

Public Types

```
using SetterFunc = std::function<void(ParameterWrapper&, Arg&)>
```

Function type for setting an argument to the parameter.

Public Functions

```
inline SetterFunc &get_argument_setter(std::type_index index)
```

Get the argument setter function object.

Parameters

index – The type index of the parameter.

Returns

The reference to the [SetterFunc](#) object.

```
template<typename typeT>  
inline void add_argument_setter(SetterFunc func)
```

Add the [SetterFunc](#) for the type.

Template Parameters

typeT – typeT The type of the parameter.

Parameters

func – The SetterFunc object.

```
inline void add_argument_setter(std::type_index index, SetterFunc func)
```

Add the SetterFunc for the type.

Parameters

- **index** – The type index of the parameter.
- **func** – The SetterFunc object.

```
template<typename typeT>  
inline void add_argument_setter()
```

Add the SetterFunc for the type.

Template Parameters

typeT – The type of the parameter.

Public Static Functions

```
static ArgumentSetter &get_instance()
```

Get the instance object.

Returns

The reference to the ArgumentSetter instance.

```
static inline void set_param(ParameterWrapper &param_wrap, Arg &arg)
```

Set the param object.

Parameters

- **param_wrap** – The ParameterWrapper object.

- **arg** – The [Arg](#) object to set.

```
template<typename typeT>  
static inline void ensure_type()
```

Register the SetterFunc for the type.

Template Parameters

typeT – The type of the parameter.

Public Static Attributes

```
static SetterFunc none_argument_setter =
```

```
[](ParameterWrapper& param_wrap, Arg& arg) {(void)param_wrap;  
(void)arg;HOLOSCAN_LOG_ERROR("Unable to handle parameter: {}", arg.name());}
```

Default [SetterFunc](#) for [Arg](#).

© Copyright 2022-2024, NVIDIA.. PDF Generated on 06/06/2024