

PyGraph Code

Generated by Yash Kanani

October 31, 2024

1 Hierarchical Index	1
1.1 Class Hierarchy	1
2 Class Index	3
2.1 Class List	3
3 File Index	5
3.1 File List	5
4 Class Documentation	7
4.1 JSONDataHandler::_JSONContainerHeader Struct Reference	7
4.2 AbstractElement Class Reference	7
4.3 BuilderContainer Class Reference	9
4.4 BuilderControlPanel Class Reference	9
4.5 BuilderControlsButtons Class Reference	10
4.6 CodeText Class Reference	10
4.7 ConditionalElement Class Reference	11
4.7.1 Member Function Documentation	11
4.7.1.1 clone()	11
4.7.1.2 getText()	11
4.7.1.3 getUserInput()	12
4.7.1.4 getViewWidget()	12
4.7.1.5 setUserInput()	12
4.7.1.6 updateParameterWidgets()	12
4.8 ConstantDecimalElement Class Reference	12
4.8.1 Member Function Documentation	13
4.8.1.1 clone()	13
4.8.1.2 getText()	13
4.8.1.3 getUserInput()	13
4.8.1.4 getViewWidget()	13
4.8.1.5 setUserInput()	14
4.8.1.6 updateParameterWidgets()	14
4.9 ConstantTextElement Class Reference	14
4.9.1 Member Function Documentation	14
4.9.1.1 clone()	15
4.9.1.2 getText()	15
4.9.1.3 getUserInput()	15
4.9.1.4 getViewWidget()	15
4.9.1.5 setUserInput()	15
4.9.1.6 updateParameterWidgets()	15
4.10 ContainerInformation Struct Reference	16
4.11 CustomElementInputDialog Class Reference	16
4.12 DeleteItem Class Reference	16

4.13 ElementDragEventHandler Class Reference	17
4.14 ElementManager Class Reference	17
4.15 ElementsControlButtons Class Reference	18
4.16 ElementsCreator Class Reference	18
4.17 ElementsLeftNavigator Class Reference	19
4.18 ElementsListWidget Class Reference	19
4.19 ElementUserInputs Class Reference	20
4.20 ForLoopElement Class Reference	20
4.20.1 Member Function Documentation	21
4.20.1.1 clone()	21
4.20.1.2 getText()	21
4.20.1.3 getUserInput()	21
4.20.1.4 getViewWidget()	21
4.20.1.5 setUserInput()	21
4.20.1.6 updateParameterWidgets()	22
4.21 IfLoopElement Class Reference	22
4.21.1 Member Function Documentation	22
4.21.1.1 clone()	22
4.21.1.2 getText()	23
4.21.1.3 getUserInput()	23
4.21.1.4 getViewWidget()	23
4.21.1.5 setUserInput()	23
4.21.1.6 updateParameterWidgets()	23
4.22 ImageButton Class Reference	24
4.23 InputLabelDialog Class Reference	24
4.24 InternalDragEventHandler Class Reference	24
4.25 JSONDataHandler Class Reference	25
4.26 MainAppWindow Class Reference	25
4.27 MainWindowUI Class Reference	26
4.28 OperationElement Class Reference	26
4.28.1 Member Function Documentation	27
4.28.1.1 clone()	27
4.28.1.2 getText()	27
4.28.1.3 getUserInput()	27
4.28.1.4 getViewWidget()	27
4.28.1.5 setUserInput()	27
4.28.1.6 updateParameterWidgets()	28
4.29 ParameterUIBuilder Class Reference	28
4.30 ReadVariableElement Class Reference	28
4.30.1 Member Function Documentation	29
4.30.1.1 clone()	29
4.30.1.2 getText()	29

4.30.1.3 getUserInput()	29
4.30.1.4 getViewWidget()	29
4.30.1.5 setUserInput()	30
4.30.1.6 updateParameterWidgets()	30
4.31 ResultedTextView Class Reference	30
4.32 SearchBarWidget Class Reference	31
4.33 SettingsManager Class Reference	31
4.34 UserDefinedElement Class Reference	32
4.34.1 Member Function Documentation	32
4.34.1.1 clone()	32
4.34.1.2 getText()	32
4.34.1.3 getUserInput()	33
4.34.1.4 getViewWidget()	33
4.34.1.5 setUserInput()	33
4.34.1.6 updateParameterWidgets()	33
4.35 WriteVariableElement Class Reference	33
4.35.1 Member Function Documentation	34
4.35.1.1 clone()	34
4.35.1.2 getText()	34
4.35.1.3 getUserInput()	34
4.35.1.4 getViewWidget()	34
4.35.1.5 setUserInput()	35
4.35.1.6 updateParameterWidgets()	35
5 File Documentation	37
5.1 AbstractElement.h	37
5.2 BasicElementType.h	37
5.3 BuilderContainer.h	38
5.4 BuilderControlPanel.h	38
5.5 BuilderControlsButtons.h	39
5.6 CodeText.h	39
5.7 ConditionalElement.h	40
5.8 ConstantDecimalElement.h	40
5.9 ConstantTextElement.h	40
5.10 ContainerInformation.h	41
5.11 CustomElementInputDialog.h	41
5.12 DeleteItem.h	41
5.13 ElementDragEventHandler.h	42
5.14 ElementManager.h	42
5.15 ElementsControlButtons.h	43
5.16 ElementsCreator.h	43
5.17 ElementsLeftNavigator.h	43

5.18 ElementsListWidget.h	44
5.19 ElementUserInputs.h	44
5.20 ForLoopElement.h	45
5.21 IfLoopElement.h	45
5.22 ImageButton.h	46
5.23 InputLabelDialog.h	46
5.24 InternalDragEventHandler.h	46
5.25 JSONDataHandler.h	46
5.26 MainWindow.h	47
5.27 MainWindowUI.h	47
5.28 OperationElement.h	47
5.29 ParameterUIBuilder.h	48
5.30 ReadVariableElement.h	48
5.31 ResultedTextView.h	49
5.32 SearchBarWidget.h	49
5.33 SettingsManager.h	49
5.34 UserDefinedElement.h	50
5.35 WriteVariableElement.h	50
Index	51

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

JSONDataHandler::_JSONContainerHeader	7
CodeText	10
ContainerInformation	16
ElementManager	17
ElementUserInputs	20
JSONDataHandler	25
QDialog	
CustomElementInputDialog	16
InputLabelDialog	24
QFrame	
BuilderContainer	9
SearchBarWidget	31
QMainWindow	
MainAppWindow	25
QObject	
AbstractElement	7
ConditionalElement	11
ConstantDecimalElement	12
ConstantTextElement	14
ForLoopElement	20
IfLoopElement	22
OperationElement	26
ReadVariableElement	28
UserDefinedElement	32
WriteVariableElement	33
ElementDragEventHandler	17
InternalDragEventHandler	24
MainWindowUI	26
SettingsManager	31
QWidget	
BuilderControlPanel	9
BuilderControlsButtons	10
DeleteItem	16
ElementsControlButtons	18
ElementsCreator	18

ElementsLeftNavigator	19
ElementsListWidget	19
ImageButton	24
ParameterUIBuilder	28
ResultedTextView	30

Chapter 2

Class Index

2.1 Class List

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

JSONDataHandler::_JSONContainerHeader	7
AbstractElement	7
BuilderContainer	9
BuilderControlPanel	9
BuilderControlsButtons	10
CodeText	10
ConditionalElement	11
ConstantDecimalElement	12
ConstantTextElement	14
ContainerInformation	16
CustomElementInputDialog	16
DeleteItem	16
ElementDragEventHandler	17
ElementManager	17
ElementsControlButtons	18
ElementsCreator	18
ElementsLeftNavigator	19
ElementsListWidget	19
ElementUserInputs	20
ForLoopElement	20
IfLoopElement	22
ImageButton	24
InputLabelDialog	24
InternalDragEventHandler	24
JSONDataHandler	25
MainAppWindow	25
MainWindowUI	26
OperationElement	26
ParameterUIBuilder	28
ReadVariableElement	28
ResultedTextView	30
SearchBarWidget	31
SettingsManager	31
UserDefinedElement	32
WriteVariableElement	33

Chapter 3

File Index

3.1 File List

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

AbstractElement.h	37
BasicElementType.h	37
BuilderContainer.h	38
BuilderControlPanel.h	38
BuilderControlsButtons.h	39
CodeText.h	39
ConditionalElement.h	40
ConstantDecimalElement.h	40
ConstantTextElement.h	40
ContainerInformation.h	41
CustomElementInputDialog.h	41
DeleteItem.h	41
ElementDragEventHandler.h	42
ElementManager.h	42
ElementsControlButtons.h	43
ElementsCreator.h	43
ElementsLeftNavigator.h	43
ElementsListWidget.h	44
ElementUserInputs.h	44
ForLoopElement.h	45
IfLoopElement.h	45
ImageButton.h	46
InputLabelDialog.h	46
InternalDragEventHandler.h	46
JSONDataHandler.h	46
MainWindow.h	47
MainWindowUI.h	47
OperationElement.h	47
ParameterUIBuilder.h	48
ReadVariableElement.h	48
ResultedTextView.h	49
SearchBarWidget.h	49
SettingsManager.h	49
UserDefinedElement.h	50
WriteVariableElement.h	50

Chapter 4

Class Documentation

4.1 JSONDataHandler::_JSONContainerHeader Struct Reference

Public Attributes

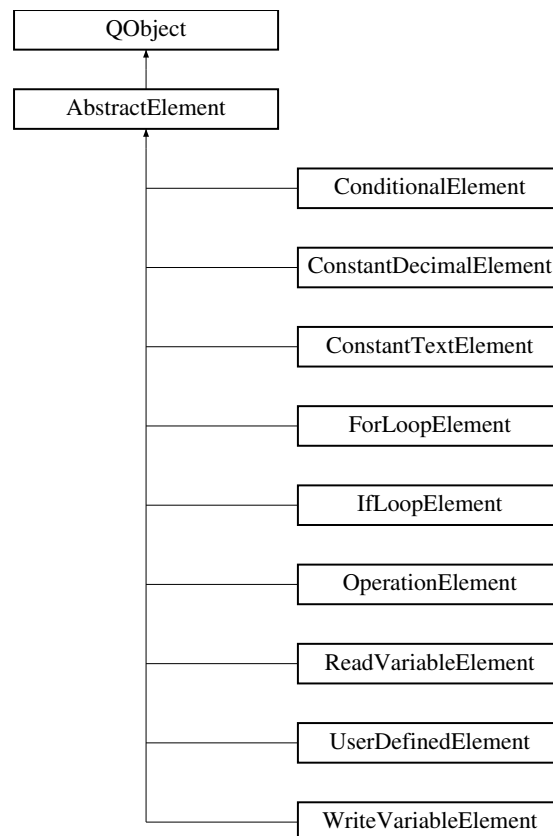
- QString **displayName**
- ContainerInformationList **containerinformationList**

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

- JSONDataHandler.h

4.2 AbstractElement Class Reference

Inheritance diagram for AbstractElement:



Signals

- void **childValueChanged** ()
- void **notifyToParameterWidgets** ()

Public Member Functions

- virtual std::shared_ptr< [AbstractElement](#) > **clone** () const =0
- QString **getName** () const
- BasicElementType **getType** () const
- QPixmap **getImage** () const
- void **setName** (const QString &name)
- virtual std::shared_ptr< [CodeText](#) > **getText** (int indentLevel) const =0
- virtual QWidget * **getViewWidget** (QWidget *parent=nullptr)=0
- virtual void **setUserInput** (std::shared_ptr< [ElementUserInputs](#) > userInput)=0
- virtual std::shared_ptr< [ElementUserInputs](#) > **getUserInput** ()=0
- virtual void **updateParameterWidgets** ([ParameterUIBuilder](#) *const parameterUIBuilder)=0

Protected Attributes

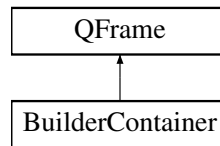
- BasicElementType **type**
- QString **name**
- QPixmap **image**

The documentation for this class was generated from the following files:

- AbstractElement.h
- AbstractElement.cpp

4.3 BuilderContainer Class Reference

Inheritance diagram for BuilderContainer:



Signals

- void **updateResultedTextView** ()
- void **EnsureVisible** (int xPosition, int yPosition, int xmargin=50, int ymargin=50)
- void **notifyToParameterWidgets** ()

Public Member Functions

- **BuilderContainer** (QWidget *parent=nullptr, bool isSubContainer=false)
- std::shared_ptr< [CodeText](#) > **getText** (int indentLevel) const
- void **setAcceptedTypes** (const QList< BasicElementType > &acceptedTypes)
- void **setMaxElements** (int maxElements)
- void **removeElementFromContainerInformation** (const [AbstractElement](#) *element)
- const ContainerInformationList & **getContainerInformation** () const
- void **appendContainerInformationList** (const ContainerInformationList &informationList, int insertIndex=0)
- void **updateParameterWidgets** ([ParameterUIBuilder](#) *const parameterUIBuilder)

Protected Member Functions

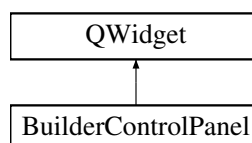
- void **dragEnterEvent** (QDragEnterEvent *event) override
- void **dragMoveEvent** (QDragMoveEvent *event) override
- void **dropEvent** (QDropEvent *event) override
- void **dragLeaveEvent** (QDragLeaveEvent *event) override
- void **paintEvent** (QPaintEvent *event) override
- void **mousePressEvent** (QMouseEvent *event) override

The documentation for this class was generated from the following files:

- BuilderContainer.h
- BuilderContainer.cpp

4.4 BuilderControlPanel Class Reference

Inheritance diagram for BuilderControlPanel:



Public Member Functions

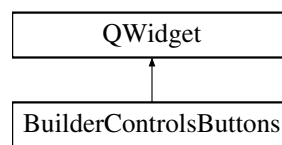
- **BuilderControlPanel** ([ElementsListWidget](#) *customElementWidget, QWidget *parent=nullptr)

The documentation for this class was generated from the following files:

- BuilderControlPanel.h
- BuilderControlPanel.cpp

4.5 BuilderControlsButtons Class Reference

Inheritance diagram for BuilderControlsButtons:



Public Member Functions

- **BuilderControlsButtons** ([BuilderContainer](#) *builderContainer, [ElementsListWidget](#) *customElementWidget, QWidget *parent=nullptr)

The documentation for this class was generated from the following files:

- BuilderControlsButtons.h
- BuilderControlsButtons.cpp

4.6 CodeText Class Reference

Public Member Functions

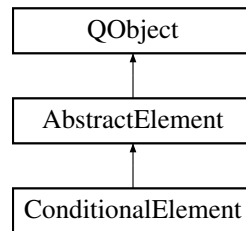
- **CodeText** (int indentLevel)
- void **addToHeader** (const QString &content)
- void **addToBody** (const QString &content)
- void **increaseIndentOfBody** ()
- void **removeIndentOfBody** ()
- int **getCurrentIndentOfBody** () const
- void **append** (const [CodeText](#) &other)
- QString **getResult** () const
- QStringList **getHeader** () const
- QString **getBody** () const

The documentation for this class was generated from the following files:

- CodeText.h
- CodeText.cpp

4.7 ConditionalElement Class Reference

Inheritance diagram for ConditionalElement:



Public Member Functions

- `std::shared_ptr< AbstractElement > clone () const` override
- `std::shared_ptr< CodeText > getText (int indentLevel) const` override
- `QWidget * getViewWidget (QWidget *parent=nullptr)` override
- `void setUserInput (std::shared_ptr< ElementUserInputs > userInput)` override
- `std::shared_ptr< ElementUserInputs > getUserInput ()` override
- `void updateParameterWidgets (ParameterUIBuilder *const parameterUIBuilder)` override

Additional Inherited Members

4.7.1 Member Function Documentation

4.7.1.1 clone()

```
std::shared_ptr< AbstractElement > ConditionalElement::clone ( ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.7.1.2 getText()

```
std::shared_ptr< CodeText > ConditionalElement::getText (
    int indentLevel ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.7.1.3 getUserInput()

```
std::shared_ptr< ElementUserInputs > ConditionalElement::getUserInput ( ) [override], [virtual]
```

Implements [AbstractElement](#).

4.7.1.4 getViewWidget()

```
QWidget * ConditionalElement::getViewWidget (
    QWidget * parent = nullptr ) [override], [virtual]
```

Implements [AbstractElement](#).

4.7.1.5 setUserInput()

```
void ConditionalElement::setUserInput (
    std::shared_ptr< ElementUserInputs > userInput ) [override], [virtual]
```

Implements [AbstractElement](#).

4.7.1.6 updateParameterWidgets()

```
void ConditionalElement::updateParameterWidgets (
    ParameterUIBuilder *const parameterUIBuilder ) [override], [virtual]
```

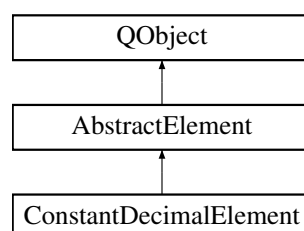
Implements [AbstractElement](#).

The documentation for this class was generated from the following files:

- ConditionalElement.h
- ConditionalElement.cpp

4.8 ConstantDecimalElement Class Reference

Inheritance diagram for ConstantDecimalElement:



Public Member Functions

- `std::shared_ptr< AbstractElement > clone ()` const override
- `std::shared_ptr< CodeText > getText (int indentLevel)` const override
- `QWidget * getViewWidget (QWidget *parent=nullptr)` override
- `void setUserInput (std::shared_ptr< ElementUserInputs > userInput)` override
- `std::shared_ptr< ElementUserInputs > getUserInput ()` override
- `void updateParameterWidgets (ParameterUIBuilder *const parameterUIBuilder)` override

Additional Inherited Members

4.8.1 Member Function Documentation

4.8.1.1 clone()

```
std::shared_ptr< AbstractElement > ConstantDecimalElement::clone ( ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.8.1.2 getText()

```
std::shared_ptr< CodeText > ConstantDecimalElement::getText (
    int indentLevel ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.8.1.3 getUserInput()

```
std::shared_ptr< ElementUserInputs > ConstantDecimalElement::getUserInput ( ) [override],
[virtual]
```

Implements [AbstractElement](#).

4.8.1.4 getViewWidget()

```
QWidget * ConstantDecimalElement::getViewWidget (
    QWidget * parent = nullptr ) [override], [virtual]
```

Implements [AbstractElement](#).

4.8.1.5 setUserInput()

```
void ConstantDecimalElement::setUserInput (
    std::shared_ptr< ElementUserInputs > userInput ) [override], [virtual]
```

Implements [AbstractElement](#).

4.8.1.6 updateParameterWidgets()

```
void ConstantDecimalElement::updateParameterWidgets (
    ParameterUIBuilder *const parameterUIBuilder ) [override], [virtual]
```

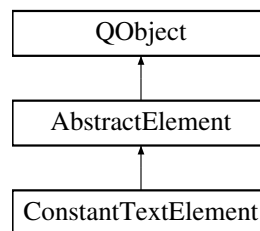
Implements [AbstractElement](#).

The documentation for this class was generated from the following files:

- ConstantDecimalElement.h
- ConstantDecimalElement.cpp

4.9 ConstantTextElement Class Reference

Inheritance diagram for ConstantTextElement:



Public Member Functions

- std::shared_ptr< [AbstractElement](#) > [clone](#) () const override
- std::shared_ptr< [CodeText](#) > [getText](#) (int indentLevel) const override
- QWidget * [getViewWidget](#) (QWidget *parent=nullptr) override
- void [setUserInput](#) (std::shared_ptr< [ElementUserInputs](#) > userInput) override
- std::shared_ptr< [ElementUserInputs](#) > [getUserInput](#) () override
- void [updateParameterWidgets](#) ([ParameterUIBuilder](#) *const parameterUIBuilder) override

Additional Inherited Members

4.9.1 Member Function Documentation

4.9.1.1 clone()

```
std::shared_ptr< AbstractElement > ConstantTextElement::clone ( ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.9.1.2 getText()

```
std::shared_ptr< CodeText > ConstantTextElement::getText (
    int indentLevel ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.9.1.3 getUserInput()

```
std::shared_ptr< ElementUserInputs > ConstantTextElement::getUserInput ( ) [override], [virtual]
```

Implements [AbstractElement](#).

4.9.1.4 getViewWidget()

```
QWidget * ConstantTextElement::getViewWidget (
    QWidget * parent = nullptr ) [override], [virtual]
```

Implements [AbstractElement](#).

4.9.1.5 setUserInput()

```
void ConstantTextElement::setUserInput (
    std::shared_ptr< ElementUserInputs > userInput ) [override], [virtual]
```

Implements [AbstractElement](#).

4.9.1.6 updateParameterWidgets()

```
void ConstantTextElement::updateParameterWidgets (
    ParameterUIBuilder *const parameterUIBuilder ) [override], [virtual]
```

Implements [AbstractElement](#).

The documentation for this class was generated from the following files:

- [ConstantTextElement.h](#)
- [ConstantTextElement.cpp](#)

4.10 ContainerInformation Struct Reference

Public Attributes

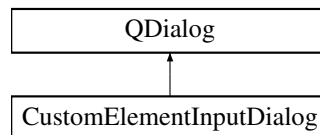
- DroppedItem **droppedItem** = DroppedItem::ELEMENT
- std::shared_ptr< [AbstractElement](#) > **elementPointer**

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

- ContainerInformation.h

4.11 CustomElementInputDialog Class Reference

Inheritance diagram for CustomElementInputDialog:



Public Member Functions

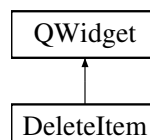
- **CustomElementInputDialog** (QWidget *parent=nullptr)
- QString **getCustomElementName** () const
- QString **getSaveLocation** () const

The documentation for this class was generated from the following files:

- CustomElementInputDialog.h
- CustomElementInputDialog.cpp

4.12 Deleteltem Class Reference

Inheritance diagram for Deleteltem:



Public Member Functions

- **Deleteltem** (const QString &closeImage, const QString &openImage, QWidget *parent=nullptr)

Protected Member Functions

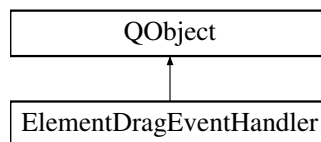
- void **dragEnterEvent** (QDragEnterEvent *event) override
- void **dragMoveEvent** (QDragMoveEvent *event) override
- void **dropEvent** (QDropEvent *event) override
- void **dragLeaveEvent** (QDragLeaveEvent *event) override
- void **paintEvent** (QPaintEvent *event) override

The documentation for this class was generated from the following files:

- Deleteltem.h
- Deleteltem.cpp

4.13 ElementDragEventHandler Class Reference

Inheritance diagram for ElementDragEventHandler:



Public Member Functions

- **ElementDragEventHandler** (QObject *parent=nullptr)

Protected Member Functions

- bool **eventFilter** (QObject *watched, QEvent *event) override

The documentation for this class was generated from the following files:

- ElementDragEventHandler.h
- ElementDragEventHandler.cpp

4.14 ElementManager Class Reference

Public Member Functions

- std::vector< [AbstractElement](#) * > **getAllElements** ()
- std::shared_ptr< [AbstractElement](#) > **createElementFromName** (const QString &elementName)
- std::shared_ptr< [AbstractElement](#) > **createCustomElement** (const QString &elementName, const ContainerInformationList &informationList)

Static Public Member Functions

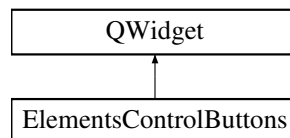
- static [ElementManager](#) & `getInstance ()`

The documentation for this class was generated from the following files:

- `ElementManager.h`
- `ElementManager.cpp`

4.15 ElementsControlButtons Class Reference

Inheritance diagram for ElementsControlButtons:



Signals

- void `addButtonClicked ()`

Public Member Functions

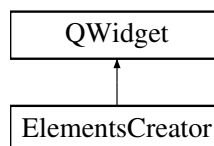
- **ElementsControlButtons** (`QWidget *parent=nullptr`)

The documentation for this class was generated from the following files:

- `ElementsControlButtons.h`
- `ElementsControlButtons.cpp`

4.16 ElementsCreator Class Reference

Inheritance diagram for ElementsCreator:



Public Member Functions

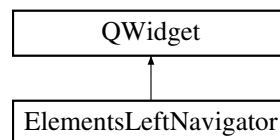
- **ElementsCreator** (QWidget *parent=nullptr)

The documentation for this class was generated from the following files:

- ElementsCreator.h
- ElementsCreator.cpp

4.17 ElementsLeftNavigator Class Reference

Inheritance diagram for ElementsLeftNavigator:



Public Member Functions

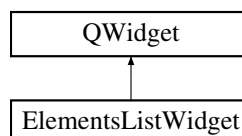
- **ElementsLeftNavigator** ([ElementsListWidget](#) *builderContainer, QWidget *parent=nullptr)

The documentation for this class was generated from the following files:

- ElementsLeftNavigator.h
- ElementsLeftNavigator.cpp

4.18 ElementsListWidget Class Reference

Inheritance diagram for ElementsListWidget:



Public Slots

- void **serachElement** (const QString &text)

Public Member Functions

- **ElementsListWidget** (QWidget *parent=nullptr)
- void **addElement** ([AbstractElement](#) *element)
- void **setElementsPerRow** (int numberOfElementsPerRow)

The documentation for this class was generated from the following files:

- ElementsListWidget.h
- ElementsListWidget.cpp

4.19 ElementUserInputs Class Reference

Public Member Functions

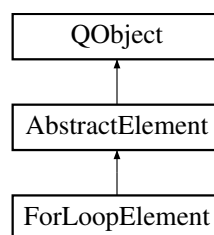
- void **addContainer** (const std::string &key, std::shared_ptr< [BuilderContainer](#) > container)
- void **addString** (const std::string &key, const QString &value)
- std::shared_ptr< [BuilderContainer](#) > **getContainer** (const std::string &key) const
- QString **getString** (const std::string &key) const
- const std::map< std::string, std::variant< std::shared_ptr< [BuilderContainer](#) >, QString > > & **getAllUserInputs** () const

The documentation for this class was generated from the following files:

- ElementUserInputs.h
- ElementUserInputs.cpp

4.20 ForLoopElement Class Reference

Inheritance diagram for ForLoopElement:



Public Member Functions

- std::shared_ptr< [AbstractElement](#) > **clone** () const override
- std::shared_ptr< [CodeText](#) > **getText** (int indentLevel) const override
- QWidget * **getViewWidget** (QWidget *parent=nullptr) override
- void **setUserInput** (std::shared_ptr< [ElementUserInputs](#) > userInput) override
- std::shared_ptr< [ElementUserInputs](#) > **getUserInput** () override
- void **updateParameterWidgets** ([ParameterUIBuilder](#) *const parameterUIBuilder) override

Additional Inherited Members

4.20.1 Member Function Documentation

4.20.1.1 clone()

```
std::shared_ptr< AbstractElement > ForLoopElement::clone ( ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.20.1.2 getText()

```
std::shared_ptr< CodeText > ForLoopElement::getText (
    int indentLevel ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.20.1.3 getUserInput()

```
std::shared_ptr< ElementUserInputs > ForLoopElement::getUserInput ( ) [override], [virtual]
```

Implements [AbstractElement](#).

4.20.1.4 getViewWidget()

```
QWidget * ForLoopElement::getViewWidget (
    QWidget * parent = nullptr ) [override], [virtual]
```

Implements [AbstractElement](#).

4.20.1.5 setUserInput()

```
void ForLoopElement::setUserInput (
    std::shared_ptr< ElementUserInputs > userInput ) [override], [virtual]
```

Implements [AbstractElement](#).

4.20.1.6 updateParameterWidgets()

```
void ForLoopElement::updateParameterWidgets (
    ParameterUIBuilder *const parameterUIBuilder ) [override], [virtual]
```

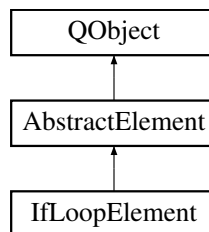
Implements [AbstractElement](#).

The documentation for this class was generated from the following files:

- ForLoopElement.h
- ForLoopElement.cpp

4.21 IfLoopElement Class Reference

Inheritance diagram for IfLoopElement:



Public Member Functions

- `std::shared_ptr< AbstractElement > clone ()` const override
- `std::shared_ptr< CodeText > getText (int indentLevel)` const override
- `QWidget * getViewWidget (QWidget *parent=nullptr)` override
- `void setUserInput (std::shared_ptr< ElementUserInputs > userInput)` override
- `std::shared_ptr< ElementUserInputs > getUserInput ()` override
- `void updateParameterWidgets (ParameterUIBuilder *const parameterUIBuilder)` override

Additional Inherited Members

4.21.1 Member Function Documentation

4.21.1.1 clone()

```
std::shared_ptr< AbstractElement > IfLoopElement::clone ( ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.21.1.2 getText()

```
std::shared_ptr< CodeText > IfLoopElement::getText (
    int indentLevel ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.21.1.3 getUserInput()

```
std::shared_ptr< ElementUserInputs > IfLoopElement::getUserInput ( ) [override], [virtual]
```

Implements [AbstractElement](#).

4.21.1.4 getViewWidget()

```
QWidget * IfLoopElement::getViewWidget (
    QWidget * parent = nullptr ) [override], [virtual]
```

Implements [AbstractElement](#).

4.21.1.5 setUserInput()

```
void IfLoopElement::setUserInput (
    std::shared_ptr< ElementUserInputs > userInput ) [override], [virtual]
```

Implements [AbstractElement](#).

4.21.1.6 updateParameterWidgets()

```
void IfLoopElement::updateParameterWidgets (
    ParameterUIBuilder *const parameterUIBuilder ) [override], [virtual]
```

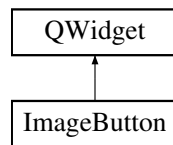
Implements [AbstractElement](#).

The documentation for this class was generated from the following files:

- IfLoopElement.h
- IfLoopElement.cpp

4.22 ImageButton Class Reference

Inheritance diagram for ImageButton:



Public Member Functions

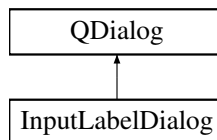
- **ImageButton** (QWidget *customWidget, const QString &text, const QString &labelObjectName, QWidget *parent=nullptr)

The documentation for this class was generated from the following files:

- ImageButton.h
- ImageButton.cpp

4.23 InputLabelDialog Class Reference

Inheritance diagram for InputLabelDialog:



Public Member Functions

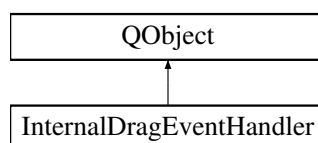
- **InputLabelDialog** (const QString &labelText, QWidget *parent=nullptr)
- QString **getText** () const

The documentation for this class was generated from the following files:

- InputLabelDialog.h
- InputLabelDialog.cpp

4.24 InternalDragEventHandler Class Reference

Inheritance diagram for InternalDragEventHandler:



Public Member Functions

- **InternalDragEventHandler** (QObject *parent=nullptr)

Protected Member Functions

- bool **eventFilter** (QObject *watched, QEvent *event) override

The documentation for this class was generated from the following files:

- InternalDragEventHandler.h
- InternalDragEventHandler.cpp

4.25 JSONDataHandler Class Reference

Classes

- struct [_JSONContainerHeader](#)

Public Types

- typedef struct [JSONDataHandler::_JSONContainerHeader](#) **JSONContainerHeader**

Public Member Functions

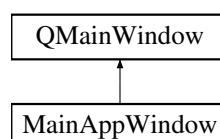
- [JSONContainerHeader](#) **readContainerInformationListFromJsonFile** (const QString &filePath) const
- bool **saveContainerInformationListToJsonFile** (const [JSONContainerHeader](#) &containerData, const QString &filePath) const

The documentation for this class was generated from the following files:

- JSONDataHandler.h
- JSONDataHandler.cpp

4.26 MainAppWindow Class Reference

Inheritance diagram for MainAppWindow:



Public Member Functions

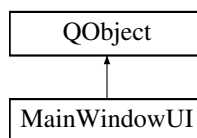
- **MainAppWindow** (QWidget *parent=nullptr)

The documentation for this class was generated from the following files:

- MainWindow.h
- MainWindow.cpp

4.27 MainWindowUI Class Reference

Inheritance diagram for MainWindowUI:



Public Member Functions

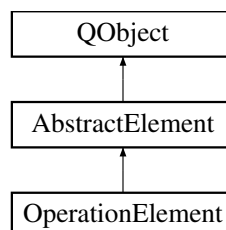
- **MainWindowUI** ([MainAppWindow](#) *main)

The documentation for this class was generated from the following files:

- MainWindowUI.h
- MainWindowUI.cpp

4.28 OperationElement Class Reference

Inheritance diagram for OperationElement:



Public Member Functions

- std::shared_ptr< [AbstractElement](#) > [clone](#) () const override
- std::shared_ptr< [CodeText](#) > [getText](#) (int indentLevel) const override
- QWidget * [getViewWidget](#) (QWidget *parent=nullptr) override
- void [setUserInput](#) (std::shared_ptr< [ElementUserInputs](#) > userInput) override
- std::shared_ptr< [ElementUserInputs](#) > [getUserInput](#) () override
- void [updateParameterWidgets](#) ([ParameterUIBuilder](#) *const parameterUIBuilder) override

Additional Inherited Members

4.28.1 Member Function Documentation

4.28.1.1 clone()

```
std::shared_ptr< AbstractElement > OperationElement::clone ( ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.28.1.2 getText()

```
std::shared_ptr< CodeText > OperationElement::getText (
    int indentLevel ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.28.1.3 getUserInput()

```
std::shared_ptr< ElementUserInputs > OperationElement::getUserInput ( ) [override], [virtual]
```

Implements [AbstractElement](#).

4.28.1.4 getViewWidget()

```
QWidget * OperationElement::getViewWidget (
    QWidget * parent = nullptr ) [override], [virtual]
```

Implements [AbstractElement](#).

4.28.1.5 setUserInput()

```
void OperationElement::setUserInput (
    std::shared_ptr< ElementUserInputs > userInput ) [override], [virtual]
```

Implements [AbstractElement](#).

4.28.1.6 updateParameterWidgets()

```
void OperationElement::updateParameterWidgets (
    ParameterUIBuilder *const parameterUIBuilder ) [override], [virtual]
```

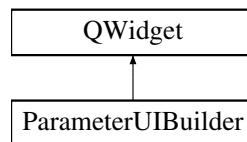
Implements [AbstractElement](#).

The documentation for this class was generated from the following files:

- OperationElement.h
- OperationElement.cpp

4.29 ParameterUIBuilder Class Reference

Inheritance diagram for ParameterUIBuilder:



Public Member Functions

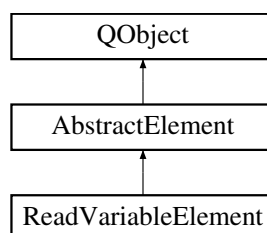
- **ParameterUIBuilder** ([BuilderContainer](#) *builderContainer, QWidget *parent=nullptr)
- void **addToUI** (QString label, QWidget *widget)
- void **removeFromUI** (QWidget *widget)

The documentation for this class was generated from the following files:

- ParameterUIBuilder.h
- ParameterUIBuilder.cpp

4.30 ReadVariableElement Class Reference

Inheritance diagram for ReadVariableElement:



Public Member Functions

- `std::shared_ptr< AbstractElement > clone ()` const override
- `std::shared_ptr< CodeText > getText (int indentLevel)` const override
- `QWidget * getViewWidget (QWidget *parent=nullptr)` override
- `void setUserInput (std::shared_ptr< ElementUserInputs > userInput)` override
- `std::shared_ptr< ElementUserInputs > getUserInput ()` override
- `void updateParameterWidgets (ParameterUIBuilder *const parameterUIBuilder)` override

Additional Inherited Members

4.30.1 Member Function Documentation

4.30.1.1 clone()

```
std::shared_ptr< AbstractElement > ReadVariableElement::clone ( ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.30.1.2 getText()

```
std::shared_ptr< CodeText > ReadVariableElement::getText (
    int indentLevel ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.30.1.3 getUserInput()

```
std::shared_ptr< ElementUserInputs > ReadVariableElement::getUserInput ( ) [override], [virtual]
```

Implements [AbstractElement](#).

4.30.1.4 getViewWidget()

```
QWidget * ReadVariableElement::getViewWidget (
    QWidget * parent = nullptr ) [override], [virtual]
```

Implements [AbstractElement](#).

4.30.1.5 setUserInput()

```
void ReadVariableElement::setUserInput (
    std::shared_ptr< ElementUserInputs > userInput )  [override], [virtual]
```

Implements [AbstractElement](#).

4.30.1.6 updateParameterWidgets()

```
void ReadVariableElement::updateParameterWidgets (
    ParameterUIBuilder *const parameterUIBuilder )  [override], [virtual]
```

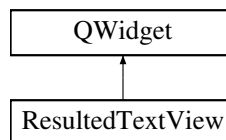
Implements [AbstractElement](#).

The documentation for this class was generated from the following files:

- ReadVariableElement.h
- ReadVariableElement.cpp

4.31 ResultedTextView Class Reference

Inheritance diagram for ResultedTextView:



Signals

- void **updateResultedTextView** ()

Public Member Functions

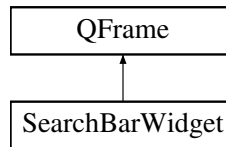
- **ResultedTextView** (QWidget *parent=nullptr, [BuilderContainer](#) *builderContainer=nullptr)

The documentation for this class was generated from the following files:

- ResultedTextView.h
- ResultedTextView.cpp

4.32 SearchBarWidget Class Reference

Inheritance diagram for SearchBarWidget:



Signals

- void **searchTextChanged** (const QString &text)

Public Member Functions

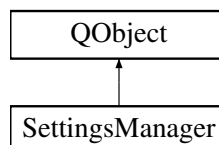
- **SearchBarWidget** (QWidget *parent=nullptr)

The documentation for this class was generated from the following files:

- SearchBarWidget.h
- SearchBarWidget.cpp

4.33 SettingsManager Class Reference

Inheritance diagram for SettingsManager:



Public Member Functions

- void **setValue** (const QString &key, const QVariant &value)
- QVariant **getValue** (const QString &key, const QVariant &defaultValue=QVariant()) const

Static Public Member Functions

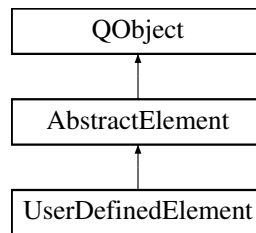
- static [SettingsManager](#) * **instance** ()

The documentation for this class was generated from the following files:

- SettingsManager.h
- SettingsManager.cpp

4.34 UserDefinedElement Class Reference

Inheritance diagram for UserDefinedElement:



Public Member Functions

- `std::shared_ptr< AbstractElement > clone ()` const override
- `std::shared_ptr< CodeText > getText (int indentLevel)` const override
- `QWidget * getViewWidget (QWidget *parent=nullptr)` override
- `void setUserInput (std::shared_ptr< ElementUserInputs > userInput)` override
- `std::shared_ptr< ElementUserInputs > getUserInput ()` override
- `void updateParameterWidgets (ParameterUIBuilder *const parameterUIBuilder)` override
- `std::string getUserInputkey ()`

Additional Inherited Members

4.34.1 Member Function Documentation

4.34.1.1 clone()

```
std::shared_ptr< AbstractElement > UserDefinedElement::clone ( ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.34.1.2 getText()

```
std::shared_ptr< CodeText > UserDefinedElement::getText (
    int indentLevel ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.34.1.3 getUserInput()

```
std::shared_ptr< ElementUserInputs > UserDefinedElement::getUserInput ( ) [override], [virtual]
```

Implements [AbstractElement](#).

4.34.1.4 getViewWidget()

```
QWidget * UserDefinedElement::getViewWidget (
    QWidget * parent = nullptr ) [override], [virtual]
```

Implements [AbstractElement](#).

4.34.1.5 setUserInput()

```
void UserDefinedElement::setUserInput (
    std::shared_ptr< ElementUserInputs > userInput ) [override], [virtual]
```

Implements [AbstractElement](#).

4.34.1.6 updateParameterWidgets()

```
void UserDefinedElement::updateParameterWidgets (
    ParameterUIBuilder *const parameterUIBuilder ) [override], [virtual]
```

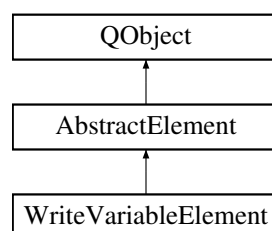
Implements [AbstractElement](#).

The documentation for this class was generated from the following files:

- UserDefinedElement.h
- UserDefinedElement.cpp

4.35 WriteVariableElement Class Reference

Inheritance diagram for WriteVariableElement:



Public Member Functions

- `std::shared_ptr< AbstractElement > clone ()` const override
- `std::shared_ptr< CodeText > getText (int indentLevel)` const override
- `QWidget * getViewWidget (QWidget *parent=nullptr)` override
- `void setUserInput (std::shared_ptr< ElementUserInputs > userInput)` override
- `std::shared_ptr< ElementUserInputs > getUserInput ()` override
- `void updateParameterWidgets (ParameterUIBuilder *const parameterUIBuilder)` override

Additional Inherited Members

4.35.1 Member Function Documentation

4.35.1.1 clone()

```
std::shared_ptr< AbstractElement > WriteVariableElement::clone ( ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.35.1.2 getText()

```
std::shared_ptr< CodeText > WriteVariableElement::getText (
    int indentLevel ) const [override], [virtual]
```

Implements [AbstractElement](#).

4.35.1.3 getUserInput()

```
std::shared_ptr< ElementUserInputs > WriteVariableElement::getUserInput ( ) [override], [virtual]
```

Implements [AbstractElement](#).

4.35.1.4 getViewWidget()

```
QWidget * WriteVariableElement::getViewWidget (
    QWidget * parent = nullptr ) [override], [virtual]
```

Implements [AbstractElement](#).

4.35.1.5 setUserInput()

```
void WriteVariableElement::setUserInput (
    std::shared_ptr< ElementUserInputs > userInput ) [override], [virtual]
```

Implements [AbstractElement](#).

4.35.1.6 updateParameterWidgets()

```
void WriteVariableElement::updateParameterWidgets (
    ParameterUIBuilder *const parameterUIBuilder ) [override], [virtual]
```

Implements [AbstractElement](#).

The documentation for this class was generated from the following files:

- WriteVariableElement.h
- WriteVariableElement.cpp

Chapter 5

File Documentation

5.1 AbstractElement.h

```
1 #ifndef ABSTRACT_ELEMENT_H
2 #define ABSTRACT_ELEMENT_H
3
4 #include <QPixmap>
5 #include <QString>
6 #include <qmetatype.h>
7 #include <qobject.h>
8
9 #include "BasicElementType.h"
10
11 class ElementUserInputs;
12 class CodeText;
13 class ParameterUIBuilder;
14 class AbstractElement : public QObject {
15     Q_OBJECT
16 public:
17     AbstractElement();
18     virtual ~AbstractElement() { } // Virtual destructor
19     virtual std::shared_ptr<AbstractElement> clone() const = 0;
20     QString getName() const;
21     BasicElementType getType() const;
22     QPixmap getImage() const;
23     void setName(const QString& name);
24     virtual std::shared_ptr<CodeText> getText(int indentLevel) const = 0;
25     virtual QWidget* getViewWidget(QWidget* parent = nullptr) = 0;
26     virtual void setUserInput(std::shared_ptr<ElementUserInputs> userInput) = 0;
27     virtual std::shared_ptr<ElementUserInputs> getUserInput() = 0;
28     virtual void updateParameterWidgets(ParameterUIBuilder* const parameterUIBuilder) = 0;
29 signals:
30     void childValueChanged();
31     void notifyToParameterWidgets();
32
33 protected:
34     BasicElementType type;
35     QString name;
36     QPixmap image;
37 };
38 Q_DECLARE_METATYPE(AbstractElement*);
39 #endif // ABSTRACT_ELEMENT_H
```

5.2 BasicElementType.h

```
1 #ifndef BASICELEMENTTYPE_H
2 #define BASICELEMENTTYPE_H
3
4 enum class BasicElementType {
5     UNKNOWN,
6     WRITE_VARIABLE,
7     READ_VARIABLE,
8     CONDITIONS,
9     OPERATOR,
10    CONSTANT_DECIMAL,
11    IF_LOOP,
12    FOR_LOOP,
13 }
```

```

13     CONSTANT_TEXT,
14     USER_DEFINED
15 };
16
17 #endif // BASICELEMENTTYPE_H

```

5.3 BuilderContainer.h

```

1 #ifndef BUILDERCONTAINER_H
2 #define BUILDERCONTAINER_H
3
4 #include <QString>
5 #include <QVBoxLayout>
6 #include <QWidget>
7
8 #include "BasicElementType.h"
9 #include "ContainerInformation.h"
10 #include <qframe.h>
11
12 class ParameterUIBuilder;
13 class CodeText;
14 class BuilderContainer : public QFrame {
15     Q_OBJECT
16
17 public:
18     explicit BuilderContainer(QWidget* parent = nullptr, bool isSubContainer = false);
19     std::shared_ptr<CodeText> getText(int indentLevel) const;
20     void setAcceptedTypes(const QList<BasicElementType>& acceptedTypes);
21     void setMaxElements(int maxElements);
22     void removeElementFromContainerInformation(const AbstractElement* element);
23     const ContainerInformationList& getContainerInformation() const;
24     void appendContainerInformationList(const ContainerInformationList& informationList, int insertIndex
    = 0);
25     void updateParameterWidgets(ParameterUIBuilder* const parameterUIBuilder);
26
27
28 protected:
29     void dragEnterEvent(QDragEnterEvent* event) override;
30     void dragMoveEvent(QDragMoveEvent* event) override;
31     void dropEvent(QDropEvent* event) override;
32     void dragLeaveEvent(QDragLeaveEvent* event) override;
33     void paintEvent(QPaintEvent* event) override;
34     void mousePressEvent(QMouseEvent* event) override;
35
36 signals:
37     void updateResultedTextView();
38     void EnsureVisible(int xPosition, int yPosition, int xmargin = 50, int ymargin = 50);
39     void notifyToParameterWidgets();
40
41 private:
42     ContainerInformationList containerInformationList;
43
44     QRect dropIndicatorRect;
45     bool isDropIndicatorVisible = false;
46     void hideDropIndicator();
47     void drawArrows();
48     void updateDropIndicator(int insertIndex);
49
50     QRect calculateDropIndicatorRect(int insertIndex) const;
51     int findInsertIndex(QDropEvent* event);
52
53     bool isDropAccepted(const BasicElementType& elementType) const;
54     bool isMaxElementsReached() const;
55     void addElementWidget(QWidget* elementWidget, int insertIndex);
56     void addInformationAndView(std::shared_ptr<AbstractElement> element, int insertIndex);
57
58     QVBoxLayout* builderContainerlayout;
59     QList<BasicElementType> acceptedTypes;
60     int maxElements;
61 };
62
63 #endif // BUILDERCONTAINER_H

```

5.4 BuilderControlPanel.h

```

1 #ifndef BUILDERCONTROL_PANEL_H
2 #define BUILDERCONTROL_PANEL_H
3
4 #include <QWidget>

```

```

5 #include <QGridLayout>
6 #include <qtablewidget.h>
7
8 class ElementsListWidget;
9 class BuilderContainer;
10
11 class BuilderControlPanel : public QWidget {
12     Q_OBJECT
13 public:
14     explicit BuilderControlPanel(ElementsListWidget* customElementWidget, QWidget* parent = nullptr);
15
16 private:
17     QTabWidget* createTabWidget(BuilderContainer* builderContainer);
18
19     QWidget* getResultedTextViewWidget(QWidget*, BuilderContainer* builderContainer);
20     QWidget* getElementsCreatorWidget(QWidget* parent);
21
22     QWidget* getButtonWidget(BuilderContainer* builderContainer, ElementsListWidget* customElementWidget,
23                             QWidget* parent);
24
25     QGridLayout* m_layout;
26 };
27
28 #endif // BUILDERCONTROL_PANEL_H

```

5.5 BuilderControlsButtons.h

```

1 #ifndef BUILDERCONTROLSBUTTONS_H
2 #define BUILDERCONTROLSBUTTONS_H
3
4 #include <QWidget>
5
6 // Forward declaration of BuilderContainer
7 class BuilderContainer;
8 class ElementsListWidget;
9 class AbstractElement;
10 class BuilderControlsButtons : public QWidget {
11     Q_OBJECT
12
13 public:
14     BuilderControlsButtons(BuilderContainer* builderContainer, ElementsListWidget* customElementWidget,
15                           QWidget* parent = nullptr);
16
17 private slots:
18     void handleSaveButtonClicked();
19     void handleLoadButtonClicked();
20
21 private:
22     BuilderContainer* m_builderContainer;
23     ElementsListWidget* m_customElementWidget;
24
25 };
26
27 #endif // BUILDERCONTROLSBUTTONS_H

```

5.6 CodeText.h

```

1 #ifndef CODETEXT_H
2 #define CODETEXT_H
3
4 #include <QString>
5 #include <QStringList>
6
7 class CodeText {
8 public:
9     CodeText(int indentLevel); // Default constructor
10    CodeText(); // Default constructor
11    void addToHeader(const QString& content);
12    void addToBody(const QString& content);
13    void increaseIndentOfBody();
14    void removeIndentOfBody();
15    int getCurrentIndentOfBody() const;
16    void append(const CodeText& other);
17    QString getResult() const;
18    QStringList getHeader() const;
19    QString getBody() const;
20

```

```

21 private:
22     QStringList header;
23     QString body;
24     int indentLevel;
25     static const int space = 4;
26 };
27
28 #endif // CODETEXT_H

```

5.7 ConditionalElement.h

```

1 #ifndef CONDITIONAL_ELEMENT_H
2 #define CONDITIONAL_ELEMENT_H
3
4 #include "AbstractElement.h"
5
6 class BuilderContainer;
7 class QComboBox;
8 class ConditionalElement : public AbstractElement {
9 public:
10     ConditionalElement();
11     std::shared_ptr<AbstractElement> clone() const override;
12     std::shared_ptr<CodeText> getText(int indentLevel) const override;
13     QWidget* getViewWidget(QWidget* parent = nullptr) override;
14     void setUserInput(std::shared_ptr<ElementUserInputs> userInput) override;
15     std::shared_ptr<ElementUserInputs> getUserInput() override;
16     void updateParameterWidgets(ParameterUIBuilder* const parameterUIBuilder) override;
17
18 private:
19     std::shared_ptr<BuilderContainer> firstVariableContainer;
20
21     std::shared_ptr<BuilderContainer> secondVariableContainer;
22
23     QString comboSelection;
24 };
25
26 #endif // CONDITIONAL_ELEMENT_H

```

5.8 ConstantDecimalElement.h

```

1 #ifndef CONSTANT_DECIMAL_ELEMENT_H
2 #define CONSTANT_DECIMAL_ELEMENT_H
3
4 #include "AbstractElement.h"
5
6
7
8 class ConstantDecimalElement : public AbstractElement {
9 public:
10     ConstantDecimalElement();
11     std::shared_ptr<AbstractElement> clone() const override;
12
13     std::shared_ptr<CodeText> getText(int indentLevel) const override;
14     QWidget* getViewWidget(QWidget* parent = nullptr) override;
15     void setUserInput(std::shared_ptr<ElementUserInputs> userInput) override;
16     std::shared_ptr<ElementUserInputs> getUserInput() override;
17     void updateParameterWidgets(ParameterUIBuilder* const parameterUIBuilder) override;
18
19 private:
20     QWidget* tempLineHolder;
21     QString lineEditValue;
22     bool isAddedToUI;
23     QString labelForUI;
24 };
25
26
27 #endif // CONSTANT_DECIMAL_ELEMENT_H

```

5.9 ConstantTextElement.h

```

1 #ifndef CONSTANT_TEXT_ELEMENT_H
2 #define CONSTANT_TEXT_ELEMENT_H
3
4 #include "AbstractElement.h"
5

```

```

6 class QLineEdit;
7
8 class ConstantTextElement : public AbstractElement {
9 public:
10     ConstantTextElement();
11     std::shared_ptr<AbstractElement> clone() const override;
12
13     std::shared_ptr<CodeText> getText(int indentLevel) const override;
14     QWidget* getViewWidget(QWidget* parent = nullptr) override;
15     void setUserInput(std::shared_ptr<ElementUserInputs> userInput) override;
16     std::shared_ptr<ElementUserInputs> getUserInput() override;
17     void updateParameterWidgets(ParameterUIBuilder* const parameterUIBuilder) override;
18
19 private:
20     QString lineEditText;
21 };
22
23
24 #endif // CONSTANT_TEXT_ELEMENT_H

```

5.10 ContainerInformation.h

```

1 #ifndef CONTAINER_INFORMATION_H
2 #define CONTAINER_INFORMATION_H
3
4 #include <QList>
5 #include <memory>
6
7 class AbstractElement;
8 class BuilderContainer;
9
10 enum class DroppedItem {
11     UNKNOWN = 0,
12     ELEMENT
13 };
14
15 typedef struct ContainerInformation {
16     DroppedItem droppedItem = DroppedItem::ELEMENT;
17     std::shared_ptr<AbstractElement> elementPointer;
18 } ContainerInformation;
19
20 typedef QList<ContainerInformation> ContainerInformationList;
21
22 #endif // CONTAINER_INFORMATION_H

```

5.11 CustomElementInputDialog.h

```

1 #ifndef CUSTOMELEMENTINPUTDIALOG_H
2 #define CUSTOMELEMENTINPUTDIALOG_H
3
4 #include <QDialog>
5 #include <QFileDialog>
6 #include <QLineEdit>
7
8 class CustomElementInputDialog : public QDialog {
9     Q_OBJECT
10
11 public:
12     explicit CustomElementInputDialog(QWidget* parent = nullptr);
13
14     QString getCustomElementName() const;
15     QString getSaveLocation() const;
16
17 private slots:
18     void handleBrowseButtonClicked();
19
20 private:
21     QLineEdit* nameLineEdit;
22     QLineEdit* saveLocationLineEdit;
23     QPushButton* browseButton;
24 };
25
26 #endif // CUSTOMELEMENTINPUTDIALOG_H

```

5.12 DeleteItem.h

```

1 #ifndef DELETEITEM_H

```

```

2 #define DELETEITEM_H
3
4 #include <QWidget>
5 #include <QMimeData>
6 #include <QVariant>
7 #include <QPixmap>
8
9 class DeleteItem : public QWidget {
10     Q_OBJECT
11
12 public:
13     explicit DeleteItem(const QString& closeImage, const QString& openImage, QWidget* parent = nullptr);
14
15 protected:
16     void dragEnterEvent(QDragEnterEvent* event) override;
17     void dragMoveEvent(QDragMoveEvent* event) override;
18     void dropEvent(QDropEvent* event) override;
19     void dragLeaveEvent(QDragLeaveEvent* event) override;
20     void paintEvent(QPaintEvent* event) override;
21
22
23 private:
24     QPixmap openPixmap;
25     QPixmap closePixmap;
26     bool isDustbinOpen;
27 };
28
29 #endif // DELETEITEM_H

```

5.13 ElementDragEventHandler.h

```

1 #include <qobject.h>
2
3 class ElementDragEventHandler : public QObject
4 {
5     Q_OBJECT
6
7 public:
8     explicit ElementDragEventHandler(QObject* parent = nullptr)
9         : QObject(parent)
10     {
11     }
12
13 protected:
14     bool eventFilter(QObject* watched, QEvent* event) override;
15 };
16

```

5.14 ElementManager.h

```

1 #ifndef ELEMENT_MANAGER_H
2 #define ELEMENT_MANAGER_H
3
4 #include <vector>
5 #include <memory>
6 #include "ContainerInformation.h"
7 #include "BasicElementType.h"
8
9 class AbstractElement;
10 class ElementManager {
11 public:
12     static ElementManager& getInstance();
13
14     std::vector<AbstractElement*> getAllElements();
15
16
17     std::shared_ptr<AbstractElement> createElementFromName(const QString& elementName);
18
19     std::shared_ptr<AbstractElement> createCustomElement(const QString& elementName, const
        ContainerInformationList& informationList);
20
21 private:
22     ElementManager();
23
24     ElementManager(const ElementManager&) = delete;
25     ElementManager& operator=(const ElementManager&) = delete;
26
27     void createElements();
28

```



```

29     std::vector<std::shared_ptr<AbstractElement>> elements;
30     std::vector<std::shared_ptr<AbstractElement>> customElementsList;
31
32     std::shared_ptr<AbstractElement> findElementFromType(const BasicElementType& elementType);
33     std::shared_ptr<AbstractElement> findElementFromName(const QString& elementName);
34     std::shared_ptr<AbstractElement> createElementFromType(const BasicElementType& elementType);
35     void addCustomElement(std::shared_ptr<AbstractElement>);
36 };
37
38 #endif // ELEMENT_MANAGER_H

```

5.15 ElementsControlButtons.h

```

1 #ifndef ELEMENTSCONTROLBUTTONS_H
2 #define ELEMENTSCONTROLBUTTONS_H
3
4 #include <QWidget>
5
6 class QPushButton;
7
8 class ElementsControlButtons : public QWidget
9 {
10     Q_OBJECT
11
12 public:
13     ElementsControlButtons(QWidget* parent = nullptr);
14
15 signals:
16     void addButtonClicked();
17
18 };
19
20 #endif // ELEMENTSCONTROLBUTTONS_H

```

5.16 ElementsCreator.h

```

1 #ifndef ELEMENTSCREATOR_H
2 #define ELEMENTSCREATOR_H
3
4 #include <QLineEdit>
5 #include <QPushButton>
6 #include <QWidget>
7 #include <qgridlayout.h>
8
9 class ElementsListWidget;
10 class AbstractElement;
11 class ElementsCreator : public QWidget {
12     Q_OBJECT
13
14 public:
15     explicit ElementsCreator(QWidget* parent = nullptr);
16
17 private slots:
18     void onAddVariableClicked();
19
20 private:
21     void addElement(std::shared_ptr<AbstractElement> element);
22     void addReadVariableElement(const QString& variableName);
23     void addWriteVariableElement(const QString& variableName);
24
25     ElementsListWidget* elementsListHolder;
26
27
28
29     std::vector<std::shared_ptr<AbstractElement>> customReadVariables;
30     std::vector<std::shared_ptr<AbstractElement>> customWriteVariables;
31 };
32
33 #endif // ELEMENTSCREATOR_H

```

5.17 ElementsLeftNavigator.h

```

1 #ifndef ELEMENTSLEFTNAVIGATOR_H
2 #define ELEMENTSLEFTNAVIGATOR_H
3

```

```

4 #include <QWidget>
5
6 // Forward declaration of ParameterUIBuilder (assuming it's defined elsewhere)
7 class ElementsListWidget;
8
9 class ElementsLeftNavigator : public QWidget {
10     Q_OBJECT
11 public:
12     ElementsLeftNavigator(ElementsListWidget* builderContainer, QWidget* parent = nullptr);
13
14 private:
15     ElementsListWidget* m_customElementWidget;
16
17     QWidget* getElementsWidget(QWidget* parent);
18 };
19
20
21 #endif // ELEMENTSLEFTNAVIGATOR_H

```

5.18 ElementsListWidget.h

```

1 #ifndef ELEMENTSLISTWIDGET_H
2 #define ELEMENTSLISTWIDGET_H
3
4 #include <QWidget>
5
6 class QLabel;
7 class QGridLayout;
8 class QScrollArea;
9 class QAbstractScrollArea;
10 class AbstractElement;
11 class QTreeView;
12 class QStandardItemModel;
13 class QSortFilterProxyModel;
14 class QStandardItem;
15
16
17 class ElementsListWidget : public QWidget {
18     Q_OBJECT
19 public:
20     ElementsListWidget(QWidget* parent = nullptr);
21
22 public slots:
23     void addElement(AbstractElement* element);
24     void setElementsPerRow(int numberOfElementsPerRow);
25
26 private slots:
27     void searchElement(const QString& text);
28
29 private slots:
30
31
32 private:
33     QTreeView* elementsListLayout;
34     QStandardItemModel* sourceModel;
35
36     //QGridLayout* elementsListLayout;
37     int elementsPerRow;
38     int totalModelElements;
39
40     QVector<AbstractElement*> elementsModel;
41
42
43     void clearModel();
44     void showWidgetOnView(QWidget* elementDisplayUnit);
45     QWidget* createDisplayUnit(AbstractElement* element);
46 };
47
48 #endif // ELEMENTSLISTWIDGET_H

```

5.19 ElementUserInputs.h

```

1 #ifndef ELEMENTUSERINPUTS_H
2 #define ELEMENTUSERINPUTS_H
3
4 #include <map>
5 #include <variant>
6 #include <QString>
7 #include <memory>

```

```

8
9 class BuilderContainer;
10 class ElementUserInputs {
11 public:
12     void addContainer(const std::string& key, std::shared_ptr<BuilderContainer> container);
13     void addString(const std::string& key, const QString& value);
14     std::shared_ptr<BuilderContainer> getContainer(const std::string& key) const;
15     QString getString(const std::string& key) const;
16     const std::map<std::string, std::variant<std::shared_ptr<BuilderContainer>, QString>&
        getAllUserInputs() const;
17
18 private:
19     std::map<std::string, std::variant<std::shared_ptr<BuilderContainer>, QString>& dataMap;
20 };
21
22 #endif // ELEMENTUSERINPUTS_H

```

5.20 ForLoopElement.h

```

1 #ifndef FOR_LOOP_ELEMENT_H
2 #define FOR_LOOP_ELEMENT_H
3
4 #include "AbstractElement.h"
5
6 #include <QVBoxLayout>
7 #include <QComboBox>
8 #include <QLineEdit>
9
10 class BuilderContainer;
11 class ForLoopElement : public AbstractElement {
12 public:
13     ForLoopElement();
14     std::shared_ptr<AbstractElement> clone() const override;
15
16     std::shared_ptr<CodeText> getText(int indentLevel) const override;
17     QWidget* getViewWidget(QWidget* parent = nullptr) override;
18     void setUserInput(std::shared_ptr<ElementUserInputs> userInput) override;
19     std::shared_ptr<ElementUserInputs> getUserInput() override;
20     void updateParameterWidgets(ParameterUIBuilder* const parameterUIBuilder) override;
21
22 private:
23     std::shared_ptr<BuilderContainer> variableContainer;
24     QString comboSelection;
25     std::shared_ptr<BuilderContainer> startContainer;
26     std::shared_ptr<BuilderContainer> endContainer;
27     std::shared_ptr<BuilderContainer> incrementContainer;
28     std::shared_ptr<BuilderContainer> bodyContainer;
29
30 };
31
32 #endif // FOR_LOOP_ELEMENT_H

```

5.21 IfLoopElement.h

```

1 #ifndef IF_LOOP_ELEMENT_H
2 #define IF_LOOP_ELEMENT_H
3
4 #include "AbstractElement.h"
5 class BuilderContainer;
6 class IfLoopElement : public AbstractElement {
7 public:
8     IfLoopElement();
9     std::shared_ptr<AbstractElement> clone() const override;
10     std::shared_ptr<CodeText> getText(int indentLevel) const override;
11     QWidget* getViewWidget(QWidget* parent = nullptr) override;
12     void setUserInput(std::shared_ptr<ElementUserInputs> userInput) override;
13     std::shared_ptr<ElementUserInputs> getUserInput() override;
14     void updateParameterWidgets(ParameterUIBuilder* const parameterUIBuilder) override;
15
16 private:
17     std::shared_ptr<BuilderContainer> conditionsContainer;
18     std::shared_ptr<BuilderContainer> onTrueContainer;
19     std::shared_ptr<BuilderContainer> onFalseContainer;
20 };
21
22
23 #endif // IF_LOOP_ELEMENT_H

```

5.22 ImageButton.h

```

1 #ifndef IMAGEBUTTON_H
2 #define IMAGEBUTTON_H
3
4 #include <QWidget>
5
6
7 class ImageButton : public QWidget {
8     Q_OBJECT
9 public:
10     ImageButton(QWidget* customWidget, const QString& text, const QString& labelObjectName, QWidget*
        parent = nullptr);
11
12 private:
13     QWidget* m_customWidget;
14 };
15
16 #endif // IMAGEBUTTON_H

```

5.23 InputLabelDialog.h

```

1 #ifndef INPUTLABELDIALOG_H
2 #define INPUTLABELDIALOG_H
3
4 #include <QDialog>
5 #include <QLineEdit>
6
7 class InputLabelDialog : public QDialog
8 {
9     Q_OBJECT
10
11 public:
12     explicit InputLabelDialog(const QString& labelText, QWidget* parent = nullptr);
13     QString getText() const;
14
15 private:
16     QLineEdit* labelLineEdit;
17 };
18
19 #endif // INPUTLABELDIALOG_H

```

5.24 InternalDragEventHandler.h

```

1 #ifndef INTERNALDRAGEVENTHANDLER_H
2 #define INTERNALDRAGEVENTHANDLER_H
3
4 #include <QObject>
5 #include <QEvent>
6
7 class InternalDragEventHandler : public QObject
8 {
9     Q_OBJECT
10
11 public:
12     explicit InternalDragEventHandler(QObject* parent = nullptr);
13
14 protected:
15     bool eventFilter(QObject* watched, QEvent* event) override;
16
17 };
18
19 #endif // INTERNALDRAGEVENTHANDLER_H

```

5.25 JSONDataHandler.h

```

1 #ifndef JSON_DATA_HANDLER_H
2 #define JSON_DATA_HANDLER_H
3
4 #include "ContainerInformation.h"
5 #include <QJsonObject>
6
7 class JSONDataHandler {
8 public:
9     typedef struct _JSONContainerHeader{

```

```

10     QString displayName;
11     ContainerInformationList containerinformationList;
12 } JSONContainerHeader;
13
14 JSONContainerHeader readContainerInformationListFromJsonFile(const QString& filePath) const;
15 bool saveContainerInformationListToJsonFile(const JSONContainerHeader& containerData, const QString&
    filePath) const;
16
17 private:
18     QJsonObject convertContainerInformationListToJson(const ContainerInformationList& containerList)
    const;
19     ContainerInformationList convertJsonToContainerInformationList(const QJsonObject& jsonObject) const;
20 };
21
22 #endif // JSON_DATA_HANDLER_H

```

5.26 MainWindow.h

```

1
2 #ifndef MainWindow_H
3 #define MainWindow_H
4
5 #include <QMainWindow>
6
7 class QWidget;
8 class MainWindowUI;
9
10 class MainAppWindow : public QMainWindow {
11
12 public:
13     explicit MainAppWindow(QWidget* parent = nullptr);
14     ~MainAppWindow();
15
16 private:
17     void applyStyleSheet();
18     std::unique_ptr<MainWindowUI> mainUI;
19 };
20 #endif

```

5.27 MainWindowUI.h

```

1
2 #ifndef MainWindowUI_H
3 #define MainWindowUI_H
4
5 #include <qobject.h>
6 class MainAppWindow;
7 class MainWindowUI : public QObject {
8     Q_OBJECT
9 public:
10     explicit MainWindowUI(MainAppWindow* main);
11     ~MainWindowUI();
12
13 private slots:
14     void onSearchTextChanged(const QString& searchText);
15
16 private:
17     void createCentralWidget();
18     //QWidget* getElementsWidget(QWidget* parent);
19
20     MainAppWindow* d_mainWindow;
21 };
22 #endif

```

5.28 OperationElement.h

```

1 #ifndef OPERATIONELEMENT_H
2 #define OPERATIONELEMENT_H
3
4 #include "AbstractElement.h"
5 class QComboBox;
6 class BuilderContainer;
7 class OperationElement : public AbstractElement {
8 public:
9     OperationElement();

```

```

10
11     std::shared_ptr<AbstractElement> clone() const override;
12
13     std::shared_ptr<CodeText> getText(int indentLevel) const override;
14     QWidget* getViewWidget(QWidget* parent = nullptr) override;
15     void setUserInput(std::shared_ptr<ElementUserInputs> userInput) override;
16     std::shared_ptr<ElementUserInputs> getUserInput() override;
17     void updateParameterWidgets(ParameterUIBuilder* const parameterUIBuilder) override;
18
19 private:
20     std::shared_ptr<BuilderContainer> firstVariableContainer;
21     std::shared_ptr<BuilderContainer> secondVariableContainer;
22
23     QString comboSelection;
24
25
26 };
27
28 #endif // OPERATIONELEMENT_H

```

5.29 ParameterUIBuilder.h

```

1 #ifndef PARAMETERUIBUILDER_H
2 #define PARAMETERUIBUILDER_H
3
4 #include <QHash>
5 #include <QVBoxLayout>
6 #include <QWidget>
7
8 class BuilderContainer;
9 class QLabel;
10
11 class ParameterUIBuilder : public QWidget {
12     Q_OBJECT
13 public:
14     explicit ParameterUIBuilder(BuilderContainer* builderContainer, QWidget* parent = nullptr);
15     void addToUI(QString label, QWidget* widget);
16     void removeFromUI(QWidget* widget);
17
18
19 private slots:
20     void updateParameterList();
21
22 private:
23
24     void handleParentDestroyed(QObject* parentObj);
25     void clearWidgets();
26
27     QVBoxLayout* parameterLay;
28     QHash<QWidget*, QLabel*> widgetLabelMap;
29     QHash<QWidget*, QObject*> originalParents;
30     BuilderContainer* m_builderContainer;
31 };
32
33 #endif // PARAMETERUIBUILDER_H

```

5.30 ReadVariableElement.h

```

1 #ifndef READ_VARIABLE_ELEMENT_H
2 #define READ_VARIABLE_ELEMENT_H
3
4 #include "AbstractElement.h"
5
6
7 class ReadVariableElement : public AbstractElement {
8 public:
9     ReadVariableElement();
10    std::shared_ptr<AbstractElement> clone() const override;
11
12    std::shared_ptr<CodeText> getText(int indentLevel) const override;
13    QWidget* getViewWidget(QWidget* parent = nullptr) override;
14    void setUserInput(std::shared_ptr<ElementUserInputs> userInput) override;
15    std::shared_ptr<ElementUserInputs> getUserInput() override;
16    void updateParameterWidgets(ParameterUIBuilder* const parameterUIBuilder) override;
17 };
18
19 #endif // READ_VARIABLE_ELEMENT_H

```

5.31 ResultedTextView.h

```

1 #ifndef RESULTEDTEXTVIEW_H
2 #define RESULTEDTEXTVIEW_H
3
4 #include <QWidget>
5 #include <QScrollArea>
6 #include <QLabel>
7 #include "BuilderContainer.h"
8
9 class ResultedTextView : public QWidget {
10     Q_OBJECT
11 public:
12     ResultedTextView(QWidget* parent = nullptr, BuilderContainer* builderContainer = nullptr);
13
14 private slots:
15     void updateText();
16
17 signals:
18     void updateResultedTextView();
19
20 private:
21     BuilderContainer* m_builderContainer;
22     QLabel* m_resultedTextLabel;
23 };
24
25 #endif // RESULTEDTEXTVIEW_H

```

5.32 SearchBarWidget.h

```

1 #ifndef SEARCHBARWIDGET_H
2 #define SEARCHBARWIDGET_H
3
4 #include <QFrame>
5 #include <QHBoxLayout>
6 #include <QLabel>
7 #include <QLineEdit>
8 #include <QPushButton>
9
10 class SearchBarWidget : public QFrame {
11     Q_OBJECT
12
13 public:
14     SearchBarWidget(QWidget* parent = nullptr);
15
16 signals:
17     void searchTextChanged(const QString& text);
18
19 private:
20     QHBoxLayout* searchBarLayout;
21     QLabel* searchIconLabel;
22     QLineEdit* searchLineEdit;
23     QPushButton* searchClearButton;
24 };
25 #endif // SEARCHBARWIDGET_H

```

5.33 SettingsManager.h

```

1 #ifndef SETTINGSMANAGER_H
2 #define SETTINGSMANAGER_H
3
4 #include <QObject>
5 #include <QVariant>
6
7 class SettingsManager : public QObject
8 {
9     Q_OBJECT
10
11 public:
12     static SettingsManager* instance();
13
14     void setValue(const QString& key, const QVariant& value);
15     QVariant getValue(const QString& key, const QVariant& defaultValue = QVariant()) const;
16
17 private:
18     explicit SettingsManager(QObject* parent = nullptr);
19     ~SettingsManager();
20
21     QString getSettingsFilePath() const;

```

```

22     void readSettingsFile();
23     void writeSettingsFile() const;
24
25     QString m_settingsFilePath;
26     QMap<QString, QVariant> m_settingsMap;
27 };
28
29 #endif // SETTINGSMANAGER_H

```

5.34 UserDefinedElement.h

```

1 #ifndef USER_DEFINED_ELEMENT_H
2 #define USER_DEFINED_ELEMENT_H
3
4 #include "AbstractElement.h"
5 #include "BuilderContainer.h"
6
7 class UserDefinedElement : public AbstractElement {
8 public:
9     UserDefinedElement();
10    std::shared_ptr<AbstractElement> clone() const override;
11    std::shared_ptr<CodeText> getText(int indentLevel) const override;
12    QWidget* getViewWidget(QWidget* parent = nullptr) override;
13    void setUserInput(std::shared_ptr<ElementUserInputs> userInput) override;
14    std::shared_ptr<ElementUserInputs> getUserInput() override;
15    void updateParameterWidgets(ParameterUIBuilder* const parameterUIBuilder) override;
16    std::string getUserInputkey();
17
18 private:
19    std::shared_ptr<BuilderContainer> userContainer;
20    bool isVisible;
21 };
22
23 #endif // USER_DEFINED_ELEMENT_H

```

5.35 WriteVariableElement.h

```

1 #ifndef WRITE_VARIABLE_ELEMENT_H
2 #define WRITE_VARIABLE_ELEMENT_H
3
4 #include "AbstractElement.h"
5
6 class BuilderContainer;
7
8 class WriteVariableElement : public AbstractElement {
9 public:
10    WriteVariableElement();
11    std::shared_ptr<AbstractElement> clone() const override;
12
13    std::shared_ptr<CodeText> getText(int indentLevel) const override;
14    QWidget* getViewWidget(QWidget* parent = nullptr) override;
15    void setUserInput(std::shared_ptr<ElementUserInputs> userInput) override;
16    std::shared_ptr<ElementUserInputs> getUserInput() override;
17    void updateParameterWidgets(ParameterUIBuilder* const parameterUIBuilder) override;
18
19 private:
20    std::shared_ptr<BuilderContainer> dynamicValueContainer;
21 };
22
23
24 #endif // WRITE_VARIABLE_ELEMENT_H

```


Index

AbstractElement, 7
AbstractElement.h, 37

BasicElementType.h, 37
BuilderContainer, 9
BuilderContainer.h, 38
BuilderControlPanel, 9
BuilderControlPanel.h, 38
BuilderControlsButtons, 10
BuilderControlsButtons.h, 39

clone
 ConditionalElement, 11
 ConstantDecimalElement, 13
 ConstantTextElement, 14
 ForLoopElement, 21
 IfLoopElement, 22
 OperationElement, 27
 ReadVariableElement, 29
 UserDefinedElement, 32
 WriteVariableElement, 34

CodeText, 10
CodeText.h, 39
ConditionalElement, 11
 clone, 11
 getText, 11
 getUserInput, 11
 getViewWidget, 12
 setUserInput, 12
 updateParameterWidgets, 12
ConditionalElement.h, 40
ConstantDecimalElement, 12
 clone, 13
 getText, 13
 getUserInput, 13
 getViewWidget, 13
 setUserInput, 13
 updateParameterWidgets, 14
ConstantDecimalElement.h, 40
ConstantTextElement, 14
 clone, 14
 getText, 15
 getUserInput, 15
 getViewWidget, 15
 setUserInput, 15
 updateParameterWidgets, 15
ConstantTextElement.h, 40
ContainerInformation, 16
ContainerInformation.h, 41
CustomElementInputDialog, 16
CustomElementInputDialog.h, 41

DeleteItem, 16
DeleteItem.h, 41

ElementDragEventHandler, 17
ElementDragEventHandler.h, 42
ElementManager, 17
ElementManager.h, 42
ElementsControlButtons, 18
ElementsControlButtons.h, 43
ElementsCreator, 18
ElementsCreator.h, 43
ElementsLeftNavigator, 19
ElementsLeftNavigator.h, 43
ElementsListWidget, 19
ElementsListWidget.h, 44
ElementUserInputs, 20
ElementUserInputs.h, 44

ForLoopElement, 20
 clone, 21
 getText, 21
 getUserInput, 21
 getViewWidget, 21
 setUserInput, 21
 updateParameterWidgets, 21
ForLoopElement.h, 45

getText
 ConditionalElement, 11
 ConstantDecimalElement, 13
 ConstantTextElement, 15
 ForLoopElement, 21
 IfLoopElement, 22
 OperationElement, 27
 ReadVariableElement, 29
 UserDefinedElement, 32
 WriteVariableElement, 34

getUserInput
 ConditionalElement, 11
 ConstantDecimalElement, 13
 ConstantTextElement, 15
 ForLoopElement, 21
 IfLoopElement, 23
 OperationElement, 27
 ReadVariableElement, 29
 UserDefinedElement, 32
 WriteVariableElement, 34

getViewWidget

- ConditionalElement, 12
- ConstantDecimalElement, 13
- ConstantTextElement, 15
- ForLoopElement, 21
- IfLoopElement, 23
- OperationElement, 27
- ReadVariableElement, 29
- UserDefinedElement, 33
- WriteVariableElement, 34
- IfLoopElement, 22
 - clone, 22
 - getText, 22
 - getUserInput, 23
 - getViewWidget, 23
 - setUserInput, 23
 - updateParameterWidgets, 23
- IfLoopElement.h, 45
- ImageButton, 24
- ImageButton.h, 46
- InputLabelDialog, 24
- InputLabelDialog.h, 46
- InternalDragEventHandler, 24
- InternalDragEventHandler.h, 46
- JSONDataHandler, 25
- JSONDataHandler.h, 46
- JSONDataHandler::_JSONContainerHeader, 7
- MainAppWindow, 25
- MainWindow.h, 47
- MainWindowUI, 26
- MainWindowUI.h, 47
- OperationElement, 26
 - clone, 27
 - getText, 27
 - getUserInput, 27
 - getViewWidget, 27
 - setUserInput, 27
 - updateParameterWidgets, 27
- OperationElement.h, 47
- ParameterUIBuilder, 28
- ParameterUIBuilder.h, 48
- ReadVariableElement, 28
 - clone, 29
 - getText, 29
 - getUserInput, 29
 - getViewWidget, 29
 - setUserInput, 29
 - updateParameterWidgets, 30
- ReadVariableElement.h, 48
- ResultedTextView, 30
- ResultedTextView.h, 49
- SearchBarWidget, 31
- SearchBarWidget.h, 49
- SettingsManager, 31
- SettingsManager.h, 49
- setUserInput
 - ConditionalElement, 12
 - ConstantDecimalElement, 13
 - ConstantTextElement, 15
 - ForLoopElement, 21
 - IfLoopElement, 23
 - OperationElement, 27
 - ReadVariableElement, 29
 - UserDefinedElement, 33
 - WriteVariableElement, 34
- updateParameterWidgets
 - ConditionalElement, 12
 - ConstantDecimalElement, 14
 - ConstantTextElement, 15
 - ForLoopElement, 21
 - IfLoopElement, 23
 - OperationElement, 27
 - ReadVariableElement, 30
 - UserDefinedElement, 33
 - WriteVariableElement, 35
- UserDefinedElement, 32
 - clone, 32
 - getText, 32
 - getUserInput, 32
 - getViewWidget, 33
 - setUserInput, 33
 - updateParameterWidgets, 33
- UserDefinedElement.h, 50
- WriteVariableElement, 33
 - clone, 34
 - getText, 34
 - getUserInput, 34
 - getViewWidget, 34
 - setUserInput, 34
 - updateParameterWidgets, 35
- WriteVariableElement.h, 50