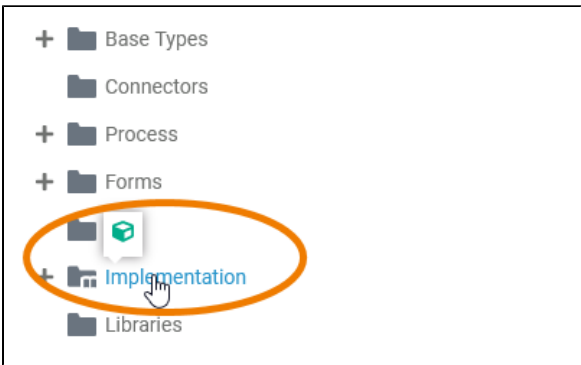


Implementation

In the service panel, you can find the folder **Implementation**. You can add your own data model to this folder. All elements inside the **Implementation** folder need to be created within packages. Add a new package via the quick action or the context menu.

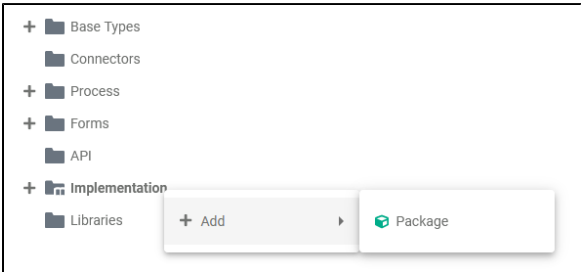
Implementation Folder Options



In a newly created service, the **Implementation** folder is empty.

If you hover over the **Implementation** folder, a quick action icon is displayed:

- [Add Package](#) 



Right click an **Implementation** folder to open its context menu.

The context menu displays all options available for the implementation folder:

- [Add Package](#)

 A package is linked directly to this

On this Page:

- [Implementation Folder Options](#)
 - [Adding Elements](#)
 - [Further Options](#)

Related Pages:

- [Service Panel](#)
 - [Managing the Service Details](#)
 - [Base Types](#)
 - [Connectors](#)
 - [Single Connector](#)
 - [Process](#)
 - [BPMN Model](#)
 - [Forms](#)
 - [Single Form](#)
 - [API](#)
 - [Implementation](#)
 - [Libraries](#)
 - [Exporting and Importing Service Panel Contents](#)
- [Implementing Your Process](#)

essential system. It is used to group expensive cuttable data model elements. Packages can have any depth of nesting: To

structure your work, you can create the packages with him packages. Also, packages defined as sort of names space

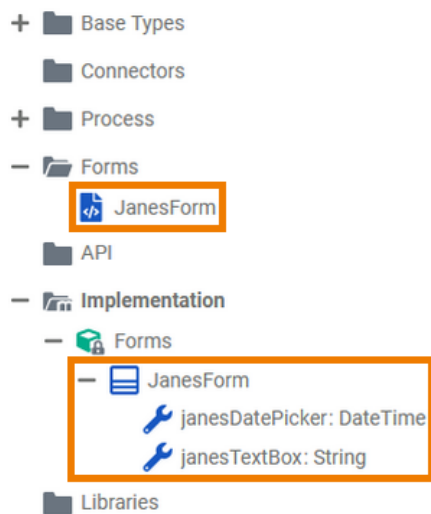
other contained elements. The name of the package is part of the element path, e.g. `Package1.Class` is different



Expert Advice

Apply the same naming conventions to all your models. This makes reading a model much easier. Refer to [Naming Conventions and Containment Tree Organization](#) in the [Builder User Guide](#) for an overview on practice-approved naming conventions.

You can add your own data model in newly created packages now. But some content is also added automatically to the **Implementation** folder:

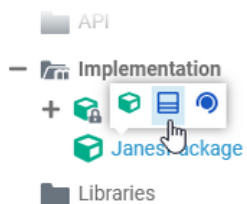


The package **Forms** is a generated package. This package is locked - you cannot change the generated form classes.

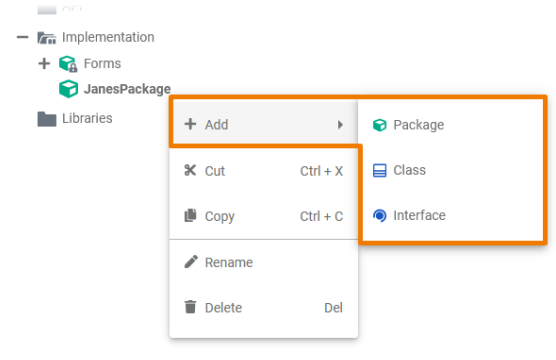
If you create a form, a corresponding class is automatically created in package **Forms**.

For each form element that you insert in the form, a corresponding property is added to the form's class.

Adding Elements



To create new elements in the implementation folder, hover over the element you want to add further content to and use one of the quick action icons that are displayed.

	Alternatively you can open the context menu and select the element you want to add.
Add Class JanesClass SaveCancel	A pop-up window opens where you need to enter a name for the element. Click Save to create the new element.  Restrictions on Element Names A BPMN model name must be unique within one service. In addition, the following name restrictions apply to all service panel elements: Element names ...

- .. must not be empty.
- .. must not contain spaces. Exception : Spaces are allowed in [operator](#) names.

• ..
· m
u
s
t
n
o
t
s
t
a
r
t
w
i
t
h
n
u
m
b
e
r
s.

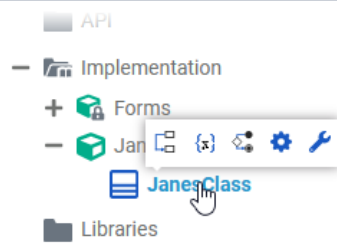
• ..
· m
u
s
t
n
o
t
e
n
d
w
i
t
h
a
p
e
r
i
o
d
(
.
).

• .. must not contain one of the following characters: <, >, :, ;, =, /, \, |, ?, *,

• Furthermore, the following strings must not be u

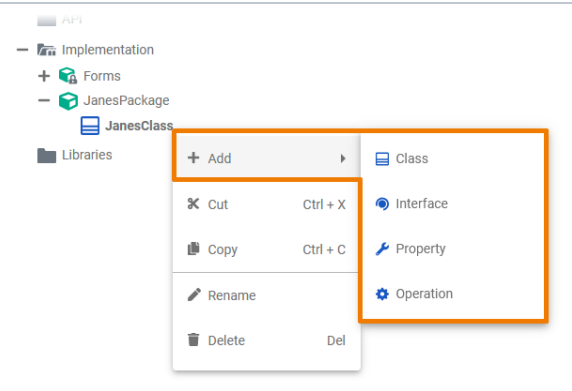
s	e	d	a	s	e	l	e	m	e	n	t	n	a	m	e	s:	C	O	N	,	P	R	N	,	A	U	X	,	N	U	L	,	C	O	M	1	,	C	O	M	2	,	C	O	M	3	,	C	O	M	4	,	C	O	M	5	,	C	O	M	6	,	C	O	M	7	,	C	O	M	8	,	C	O	M	9	,	L	P
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	----	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

T
1
,
L
P
T
2
,
L
P
T
3
,
L
P
T
4
,
L
P
T
5
,
L
P
T
6
,
L
P
T
7
,
L
P
T
8
,
L
P
T
9.



The element is added to the tree in the implementation folder.

When you hover over the element, the quick actions icons show you, which other elements you can create within this element.



These are also displayed in the context menu of each element - along with other element options such as cutting, copying, pasting, renaming and deleting.

 The possible quick actions and context menu options for all elements within an implementation folder are explained in detail on page [Modeling Data Structures](#).

 In the Implementation folder, you can undo or redo (after undo) your previous changes using the corresponding functions in the Designer editors or using the corresponding keyboard standard shortcuts (Ctrl+Z/Y).

Further Options

 For detailed explanations of the item options in the **Service** panel, such as cut, copy, rename, and delete, refer to [Service Panel](#).