

General Test

30 August 2018

Contents

1	SomeChapter	3
1.1	Some categories	3
1.2	SomeSection	4
1.3	Testing various kinds of documentation	4
1.4	Testing the group commands	6
1.5	Testing chunks	6
1.6	Testing code chunks	6

Chapter 1

SomeChapter

This is dummy text

Example

```
gap> S5 := SymmetricGroup(5);  
Sym( [ 1 .. 5 ] )  
gap> Size(S5);  
120
```

Some text between two examples

Example

```
gap> A5 := AlternatingGroup(5);  
Alt( [ 1 .. 5 ] )  
gap> Size(A5);  
60  
gap> # Test whether ]]> can be used safely  
gap> [[2]]>[[1]];  
true
```

And we wrap up with some dummy text

1.1 Some categories

Intro text

1.1.1 MyThings (for IsObject)

▷ MyThings(arg)

(filter)

Returns: true or false

1.1.2 MyThingsCollection

▷ MyThingsCollection(obj)

(filter)

Returns: true or false

1.1.3 MyThingsCollColl

▷ `MyThingsCollColl(obj)` (filter)

Returns: true or false

Let's wrap up with something, though.

1.2 SomeSection

Some test just inside a section.

1.2.1 SomeSubsection

This is a subsection!

1.2.2 Markdown support

We can use test some markdown features here:

- This is a list item.
 - This is a subitem
 - We can also use math mode here: $a^2 + b^2 = c^2$.
- This is *emphasized* text in a list item.
- This is also *emphasized* text in a list item.
- This is `inline code` in a list item.

All of this can *also* be *used* outside of a list. This text will only appear in the $\text{\LaTeX}$ version.

This text will only appear in the $\text{\LaTeX}$ version, too.

1.3 Testing various kinds of documentation

1.3.1 SomeCategory (for IsObject)

▷ `SomeCategory(arg)` (filter)

Returns: true or false

A category

1.3.2 SomeRepresentation (for IsAttributeStoringRep)

▷ `SomeRepresentation(arg)` (filter)

Returns: true or false

A collection category over the category we just created; A collection category over the category we just created; A collection category over the category we just created; A representation

1.3.3 SomeAttribute (for IsGroup)

▷ `SomeAttribute(arg)` (attribute)

An attribute

1.3.4 SomeProperty (for IsGroup)

▷ `SomeProperty(arg)` (property)

Returns: true or false

A property

1.3.5 SomeOperation (for IsInt, IsGroup)

▷ `SomeOperation(arg1, arg2)` (operation)

An operation

1.3.6 SomeConstructor (for IsGroup, IsInt)

▷ `SomeConstructor(arg1, arg2)` (constructor)

A cConstructor

1.3.7 SomeGlobalFunction

▷ `SomeGlobalFunction(arg)` (function)

A global function

1.3.8 SomeGlobalVariable

▷ `SomeGlobalVariable` (global variable)

A global variable

1.3.9 SomeGlobalName

▷ `SomeGlobalName` (global variable)

A global name

1.3.10 SomeFilter

▷ `SomeFilter` (filter)

A filter

1.3.11 SomeInfoClass

▷ SomeInfoClass (info class)

An info class

1.3.12 SomeKeyDependentOperation (for IsGroup, IsInt)

▷ SomeKeyDependentOperation(*arg1*, *arg2*) (operation)

A key dependent operation

1.4 Testing the group commands

1.4.1 A family of operations

▷ FirstOperation(*arg*) (operation)
 ▷ SecondOperation(*arg1*, *arg2*) (operation)
 ▷ ThirdOperation(*arg1*, *arg2*) (operation)

First sentence. Second sentence. Third sentence.

1.5 Testing chunks

This test comes after the chunk is declared, but before it is inserted. Hello, world. This line is indented!
 The text "Hello, world." is inserted right before this.

1.6 Testing code chunks

This test comes after the code chunk is declared, but before it is inserted.

Code

```

Hello, world.
x := 1 + 1;
if x = 2 then
  Print("1 + 1 = 2 holds, all is good\n");
else
  Error("1+1 <> 2");
fi;
```

The text "Hello, world." is inserted right before this.