

forall/2

Description

```
forall(Generator, Test)
```

For all solutions of `Generator`, `Test` is true.

Template and modes

```
forall(+callable, +callable)
```

Errors

Either `Generator` or `Test` is a variable:

```
instantiation_error
```

`Generator` is neither a variable nor a callable term:

```
type_error(callable, Generator)
```

`Test` is neither a variable nor a callable term:

```
type_error(callable, Test)
```

Examples

To call both goals in this:

```
forall(Generator, Test)
```

To call both goals in self:

```
forall(::Generator, ::Test)
```

To call both goals in an object:

```
forall(Object::Generator, Object::Test)
```

Copyright © Paulo Moura — Logtalk.org

Last updated on: October 26, 2005