forall/2

Description

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

This predicate is true if, for all solutions of Generator, Test is true (some Prolog compilers already define this or a similar predicate).

Template and modes

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

Errors

Generator is not a callable term:
```
type_error(callable, Generator)
```
Test is not a callable term:
```
type_error(callable, Test)
```

Examples

```
| ?- forall(member(X, [1, 2, 3]), write(X)).

123
yes
```