

phrase/2

Description

```
phrase(NonTerminal, Input)
```

True if the list `Input` can be parsed using the specified non-terminal `NonTerminal`. This method also accepts grammar rule bodies in the first argument (with the limitation that cuts are only supported in local calls).

Template and modes

```
phrase(+callable, ?list)
```

Errors

NonTerminal is a variable:

```
instantiation_error
```

NonTerminal is neither a variable nor a callable term:

```
type_error(callable, NonTerminal)
```

Input is neither a partial list nor a list:

```
type_error(list, Input)
```

The grammar rule non-terminal NonTerminal is private:

```
permission_error(access, private_non_terminal, NonTerminal)
```

The grammar rule non-terminal NonTerminal is protected:

```
permission_error(access, protected_non_terminal, NonTerminal)
```

The grammar rule non-terminal NonTerminal is not declared:

```
existence_error(non_terminal_declaration, NonTerminal)
```

Examples

```
| ?- sentence::phrase(sentence, [the, girl, likes, the, boy]).
```

```
yes
```

```
| ?- calculator::phrase(expr(Value), "1+2-3*4").
```

```
Value = -9
```

```
yes
```