**Accessibility/visibility/override ability modifiers**

final:

– prevents inheritance from the unit

**final unit** A

**end**

**unit** B **extend** A // CTE: Validity error

**end**

* prevents overriding the unit member

**unit** A

**final** foo **do end**

**final** goo: **as this**

**end**

**unit** B **extend** A

**override** foo // CTE: Validity error

**do end**

**end**

|  |  |  |
| --- | --- | --- |
|  | **Accessible for descendants** | |
| **Accessible to clients** | yes | no |
| Yes | public |  |
| No | protected | private |
|  |  |  |
|  | **Accessible for descendants** | |
| **Accessible to clients** | yes | no |
| Yes | {A, B, C} or {Any} |  |
| No | {} | {this} |

protected - own: {}

* prevents access to this unit member from clients of the unit, descendants have full access

**unit** A

{} foo () **do end**

{} goo: **as this**

too **do**

foo ()

goo := **this**

**end**

**end**

**unit** B **extend** A

rtn () **do**

foo ()

goo := **this**

**end**

**end**

a **is** A

a.foo () // validity error!

private - hidden: {this}

* prevents access to this unit members from all descendants of this unit and all its clients. This has full access. The caveat is descendants must call parent(s) initializer to ensure private attributes will get right values.

**unit** A

**{this}** foo **do end**

**{this}** goo: **as this**

too **do**

foo ()

goo := **this**

**end**

**end**

**unit** B **extend** A

rtn **do**

foo () // CTE: Validity error!

goo := **this** // CTE: Validity error!

**end**

**end**

a **is** A

a.foo () // CTE: Validity error!