

# Funcons-beta: Returning \*

The P<sub>L</sub>anCompS Project

Returning.cbs | PLAIN | PRETTY

---

## Returning

```
[ Datatype returning
  Funcon returned
  Funcon finalise-returning
  Funcon return
  Funcon handle-return ]
```

Meta-variables  $T <: \text{values}$

Datatype `returning` ::= `returned(_ : values)`

`returned(V?)` is a reason for abrupt termination.

```
Funcon finalise-returning( $X : \Rightarrow T$ ) :  $\Rightarrow T$  | null-type
   $\rightsquigarrow$  finalise-abrupting( $X$ )
```

`finalise-returning(X)` handles abrupt termination of  $X$  due to executing `return(V)`.

```
Funcon return( $V : T$ ) :  $\Rightarrow \text{empty-type}$ 
   $\rightsquigarrow$  abrupt(returned( $V$ ))
```

`return(V)` abruptly terminates all enclosing computations until it is handled, then giving  $V$ . Note that  $V$  may be `null-value`.

```
Funcon handle-return( $_ : \Rightarrow T$ ) :  $\Rightarrow T$ 
```

`handle-return(X)` first evaluates  $X$ . If  $X$  either terminates abruptly for reason `returned(V)`, or terminates normally with value  $V$ , it gives  $V$ .

```
Rule  $\frac{X \xrightarrow{\text{abrupted}(\_)} X'}{\text{handle-return}(X) \xrightarrow{\text{abrupted}(\_)} \text{handle-return}(X')}$ 
```

```
Rule  $\frac{X \xrightarrow{\text{abrupted}(\text{returned}(V:\text{values}))} X'}{\text{handle-return}(X) \xrightarrow{\text{abrupted}(\_)} V}$ 
```

```
Rule  $\frac{X \xrightarrow{\text{abrupted}(V':\sim\text{returning})} X'}{\text{handle-return}(X) \xrightarrow{\text{abrupted}(V')} \text{handle-return}(X')}$ 
```

```
Rule  $\text{handle-return}(V : T) \rightsquigarrow V$ 
```

---

\*Suggestions for improvement: [plancomps@gmail.com](mailto:plancomps@gmail.com).  
Reports of issues: <https://github.com/plancomps/CBS-beta/issues>.