

Contents

1 Module Foo : The first special comment of the file is the comment associated with the whole module.

Special comments can be placed between elements and are kept by the OCaml doc tool, but are not associated to any element. @-tags in these comments are ignored.

Comments like the one above, with more than two asterisks, are ignored.

```
val f : int -> int -> int
```

The comment for function f.

The continuation of the comment for function f.

```
exception My_exception of (int -> int) * int
```

Comment for exception My_exception, even with a simple comment between the special comment and the exception.

```
type weather =
```

```
| Rain of int
```

The comment for constructor Rain

```
| Sun
```

The comment for constructor Sun

Comment for type weather

```
type weather2 =
```

```
| Rain of int
```

The comment for constructor Rain

```
| Sun
```

The comment for constructor Sun

Comment for type weather2

I can continue the comment for type weather2 here because there is already a comment associated to the last constructor.

```
type my_record = {
```

```
  foo : int ;
```

Comment for field foo

```
  bar : string ;
```

Comment for field bar

```
}
```

The comment for type my_record

Continuation of comment for type my_record

```
val foo : string
    Comment for foo
    This comment is associated to foo and not to bar.
```

```
val bar : string
    This comment is associated to bar.
```

```
class my_class :
    object
```

1.0.1 A comment to describe inheritance from cl

```
val mutable tutu : string
    The comment for attribute tutu
```

```
val toto : int
    The comment for attribute toto.
```

1.0.2 This comment is not attached to titi since there is a blank line before titi, but is kept as a comment in the class.

```
val titi : string
method toto : string
    Comment for method toto
```

```
method m : float -> int
    Comment for method m
```

```
end
```

```
    The comment for class my_class
```

```
class type my_class_type =
    object
```

```
    val mutable x : int
        The comment for variable x.
```

```
    method m : int -> int
        The comment for method m.
```

```
end
```

The comment for the class type `my_class_type`

```
module type Foo =  
  sig  
    val x : int
```

The comment for x

A special comment that is kept but not associated to any element

```
end
```

The comment for module Foo

```
module type my_module_type =  
  sig  
    val x : int
```

The comment for value x.

```
module type M =  
  sig
```

```
    val y : int
```

The comment for value y.

```
end
```

The comment for module M.

```
end
```

The comment for module type `my_module_type`.