

Conditionals



Conditionals



The expression `if e1 then e2 else e3` evaluates to `e2` if `e1` evaluates to `true`, and to `e3` otherwise. We call `e1` the *guard* of the `if` expression.

Conditionals



In contrast to `if-then-else` statements found in imperative languages, `if-then-else` expressions within OCaml function just like any other expression. This means they are positionable wherever expressions are accepted.

```
4 + (if 'a' = 'b' then 1 else 2)
```

```
- : int = 6
```

Conditionals



If expressions can be nested in a pleasant way:

```
if e1 then e2
else if e3 then e4
else if e5 then e6
...
else en
```