

Exceptions OCaml

Ce sont des erreurs, lorsqu'une exception est lancée, cela interrompt tout le programme.

- `failwith` lance une exception
- `Not_found`, `Invalid_argument`, `Failure...` sont des exceptions

On peut définir ses propres exceptions :

```
exception Nadine;;
```

Ce sont des constructeurs !

On peut lancer une exception avec `raise` :

```
raise Nadine;;
```

On peut donner des arguments à nos exceptions :

```
exception Nadine of int;;
```

```
raise (Nadine 12);;
```

`Failure` et `Invalid_argument` prennent en argument une *string*.

En fait, la fonction `failwith` est codée de cette manière :

```
let failwith S = raise (Failure S);;
```

On peut rattraper les exceptions :

```
try:  
  (* instructions renvoyant du 'a *)  
  foo ();  
  ....  
with  
  (* Ce pattern matching renvoie également du 'a *)  
  | Failure s -> bar ()  
  | Nadine s -> nad ()
```