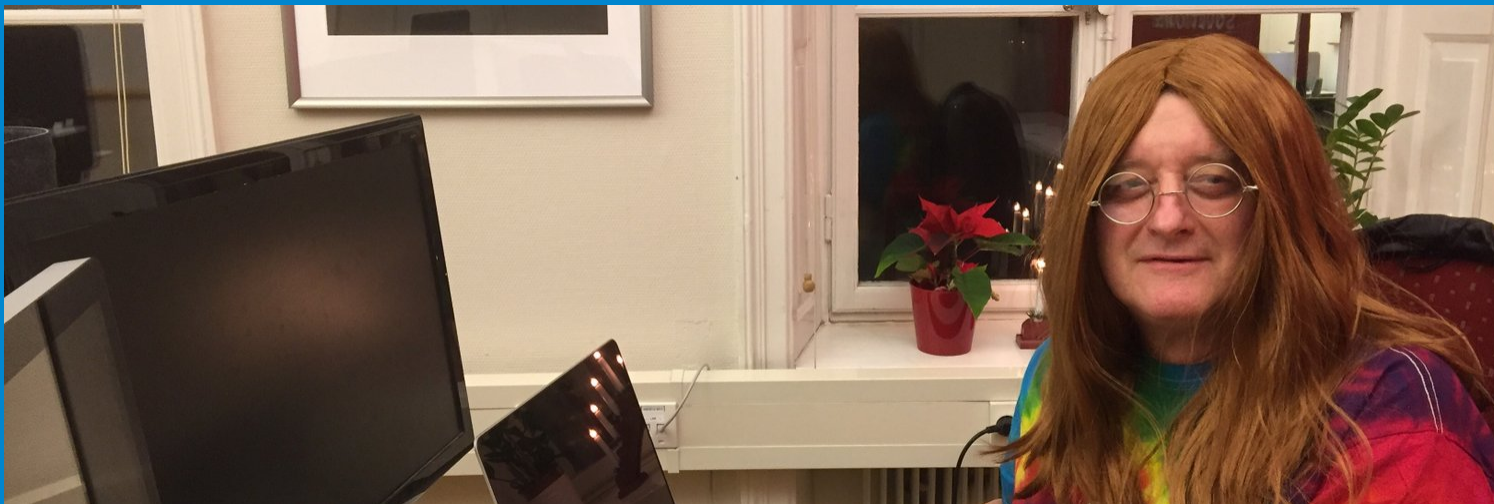# Introducción a LFE

Todo lo que necesitas para ser un crack 🐯

`(lisp (flavoured (erlang)))`

LFE es un dialecto de LISP creado por _Robert Virding_ sobre la Erlang VM (beam).

Es un Lisp-2, LFE tiene diferentes namespaces. Podes tener una fun `help` y una var `help`

¿Por qué Erlang?

- Es una técnologia probada con +**3** **0** años en la ~~trinchera~~ industria

- Manejo de concurrencia de forma sana ❗

- Soft-real time server side

- Capacidad de Tolerancia a fallos

- OTP: Patrones de diseño reales (~~Gang of Four~~)

- La VM es más un OS que un interprete del bytecode

No tenés que saber OTP !

Si sabés un poco de Lisp, podes aprender el ecosistema sobre la marcha

Números

Los enteron pueden ser tan grandes como quieras o te quedés sin memoria, lo que suceda primero 😅

```
lfe> ( + 1000000000  9999999999999999999999999999999)
1000000000000000000000000099999999
```

También hay de punto flotante, pero a nadie le importa **(?)**

```
lfe> ( /  1000000444  991)
1009082.1836528758
```

Cadenas

las cadenas en Erlang son **listas** 🍏 🎩 ...
pero de enteros 😉

```
lfe> (++ "Ceci n'est pas une " (99 104 97 238 110 101))
"Ceci n'est pas une chaîne"
```

```
lfe> (lfe_io:format "Maximale ascii est: ~c. "
        (list (lists:max "Ceci n'est pas une chaîne")))
Maximale ascii est: î. ok
```

```
erlang> io:format("Maximale ascii est: ~c. ",
        [lists:max("Ceci n'est pas une chaîne")]).
Maximale ascii est: î. ok
```

Átomos

son **enums** que se representan así mismos, los átomos empiezan con comilla simple **'**.

```
lfe> (erlang:is_atom 'desinflamante)
true
```

```
erlang> erlang:is_atom(desinflamante).
true
```

- **'true** tiene un valor `truthy` y el **'false** `falsy`

- No hay **null**🚫, pero podés definir el átomo **'null** , **'undefined**, **'none**, **'nothing**, **'lol** , **'ahre**

Links

Create links to any external resource—like [a website](http://www.decksetapp.com)—by wrapping link text in square brackets, followed immediately by a set of regular parentheses containing the URL where you want the link to point:

`'[a website](http://www.decksetapp.com)'`

Your links will be clickable in exported PDFs as well!

Code samples

Wrap your code with three backticks and specify the language for automatic syntax highlighting.

```
UIView *someView = [[UIView alloc] init];
NSString *string = @"some string that is fairly long, with
```

We scale the text dynamically so it always looks great. You can also use a single indent to switch to a monospace font.

Images

The simplest way to get images into your presentation is to drop a local or web image onto the Deckset window — the markdown is automatically created and copied to your clipboard.

You can also add images with the following syntax:

`'![](your image.jpg)'`

If you use text and images together, the image is filtered so the text is always readable.

Take a look at the 'Working with images' example presentation for a complete overview of what you can do with images in Deckset.

right

Videos can be included too, either as local files or YouTube links.

autoplay

Quotes

All slides containing a single quote have special formatting for extra impact.

Add `>` in front of every quote line
And add `--` for the quote author reference

" The best way to predict the future is to invent it
  -- Alan Kay
                                                                              "

Make sure the two directives start on the *first line* of your markdown file, and ensure there are *no empty lines* between the two.

More control with a little HTML

If you really must tweak line breaks, you can use the `<br/>` tag to split any line of text.