

Hamburg-BEAMers



elixir

Meetup #2

- ▶ Language: English || German

Organisational stuff

- ▶ Language: English || German
- ▶ Naming: clone Vienna-BEAMers

Organisational stuff

- ▶ Language: English || German
- ▶ Naming: clone Vienna-BEAMers
- ▶ Intervall: 0 19 11-17 */2 1 /bin/meetup

Organisational stuff

- ▶ Language: English || German
- ▶ Naming: clone Vienna-BEAMers
- ▶ Intervall: 0 19 11-17 */2 1 /bin/meetup
- ▶ Communication: Meetup.com || Twitter

Organisational stuff

- ▶ Language: English || German
- ▶ Naming: clone Vienna-BEAMers
- ▶ Intervall: 0 19 11-17 */2 1 /bin/meetup
- ▶ Communication: Meetup.com || Twitter
- ▶ Website: *Maybe with Elixir/Phoenix?*

Organisational stuff

- ▶ Language: English || German
- ▶ Naming: clone Vienna-BEAMers
- ▶ Intervall: 0 19 11-17 */2 1 /bin/meetup
- ▶ Communication: Meetup.com || Twitter
- ▶ Website: *Maybe with Elixir/Phoenix?*
- ▶ Locations

Organisational stuff

- ▶ Language: English || German
- ▶ Naming: clone Vienna-BEAMers
- ▶ Intervall: 0 19 11-17 */2 1 /bin/meetup
- ▶ Communication: Meetup.com || Twitter
- ▶ Website: *Maybe with Elixir/Phoenix?*
- ▶ Locations
- ▶ Contribute

Organisational stuff

- ▶ Language: English || German
- ▶ Naming: clone Vienna-BEAMers
- ▶ Intervall: 0 19 11-17 */2 1 /bin/meetup
- ▶ Communication: Meetup.com || Twitter
- ▶ Website: *Maybe with Elixir/Phoenix?*
- ▶ Locations
- ▶ Contribute
- ▶ Whats your intention?

What is



elixir

?

Elixir is a functional, concurrent, general-purpose programming language that runs on the Erlang Virtual Machine (BEAM). Elixir builds on top of Erlang to provide distributed, fault-tolerant, soft real-time, non-stop applications but also extends it to support metaprogramming with macros and polymorphism via protocols.



elixir

Paradigm	multi-paradigm: functional, concurrent, process-oriented, homoiconic
Appeared in	2012
Stable release	1.0.3
Typing discipline	dynamic, strong
Platform	Erlang
License	Apache License
Filename extensions	.ex, .exs
Website	elixir-lang.org 

Influenced by

Erlang, Ruby, Clojure

Example Elixir Code

hellomodule.exs

```
defmodule HelloModule do
  def hello(name \\ "World") do
    IO.puts "Hello " <> String.capitalize name
  end
end
```

```
HelloModule.hello
#=> "Hello World"
```

```
HelloModule.hello "julius"
#=> "Hello Julius"
```

Example Elixir Code

fib.exs

```
defmodule Fib do
  def fib(0), do: 0
  def fib(1), do: 1
  def fib(n) do fib(n-1) + fib(n-2) end
end
```

```
IO.puts Fib.fib(10)
```

```
#>= 55
```

Show us your
#myelixirstatus!