

Свойства Эликсир

- ▶ Мощная система макросов
- ▶ Высокоуровневый язык
- ▶ Экосистема и сообщество

Система макросов

Работает на уровне абстрактного синтаксического дерева
Abstract Syntax Tree (**AST**).

Позволяет расширять язык и строить
Domain Specific Languages (**DSL**).

Примеры: **Ecto** и **Phoenix**.

Высокоуровневый язык

- ▶ Оператор **pipe**
- ▶ Макрос **with**
- ▶ Абстракции Эликсир
- ▶ Абстракции Эрланг

Оператор pipe

Несколько вызовов функций:

```
a = 42  
b = func1(a)  
c = func2(b)  
d = func3(c)
```

Можно собрать в цепочку:

```
d = func3(func2(func1(42)))
```

С оператором pipe сохраняется естественный порядок:

```
d = 42 |> func1() |> func2() |> func3()
```

Макрос with

Композиция функций, которые могут вернуть ошибку:

```
case func1() do
  {:ok, a} ->
    case func2() do
      {:ok, b} ->
        case func3() do
          {:ok, c} -> a + b + c
          error -> error
        end
      error -> error
    end
  error -> error
end
```

Макрос with

С макросом **with**:

```
with a = func1(),  
     b = func2(),  
     c = func3() do  
  a + b + c  
end
```

Обработка ошибок в стиле **Railway programming**
(или **do-notation**).

Абстракции Эликсир

- ▶ Коллекции **Enum**
- ▶ Ленивые коллекции **Stream**
- ▶ Строковые литералы **Sigil**
- ▶ Протоколы
- ▶ **Task** и **Agent**

Абстракции Эрланг

- ▶ Функции высшего порядка
- ▶ Конструкторы списков
- ▶ Сопоставление с образцом
- ▶ **gen_server** и **supervisor**

Экосистема и сообщество

- ▶ Большое и активное сообщество
- ▶ Много библиотек и инструментов
- ▶ Язык и экосистема быстро развиваются
- ▶ Заимствование идей и инструментов из Эликсир в Эрланг