

RE

```
regex =  
    Re.one_or_more(Re.Chars.any_digit)  
    |> Re.compile  
"oh hi mark" =~ regex      # false  
"42" =~ regex              # true
```

```
defmacro raw(expr) do
  eager [expr] do
    quote generated: true do
      case unquote(expr) do
        %Regex{} = val -> {:re_expr, Regex.source(val)}
        val -> {:re_expr, val}
      end
    end
  end
end
end
```

```
defmacro eager(params, do: block) do
  quote generated: true do
    if Macro.quoted_literal?(unquote(params)) do
      {term, _} = unquote(block) |> Code.eval_quoted()
      Macro.escape(term)
    else
      unquote(block)
    end
  end
end
end
```

```
defmacro raw(expr) do
  expr = Macro.expand(expr, __ENV__)
  eager [expr] do
    // ...
  end
end
```

GITHUB.COM/ORSINIUM-LABS/RE

ORSINIUM.DEV