

Os pilares de Gleam

Os 4 fundamentos que guiam a linguagem

Segura

Gleam une a segurança de um robusto sistema de tipos estáticos com a confiabilidade da BEAM

Amigável

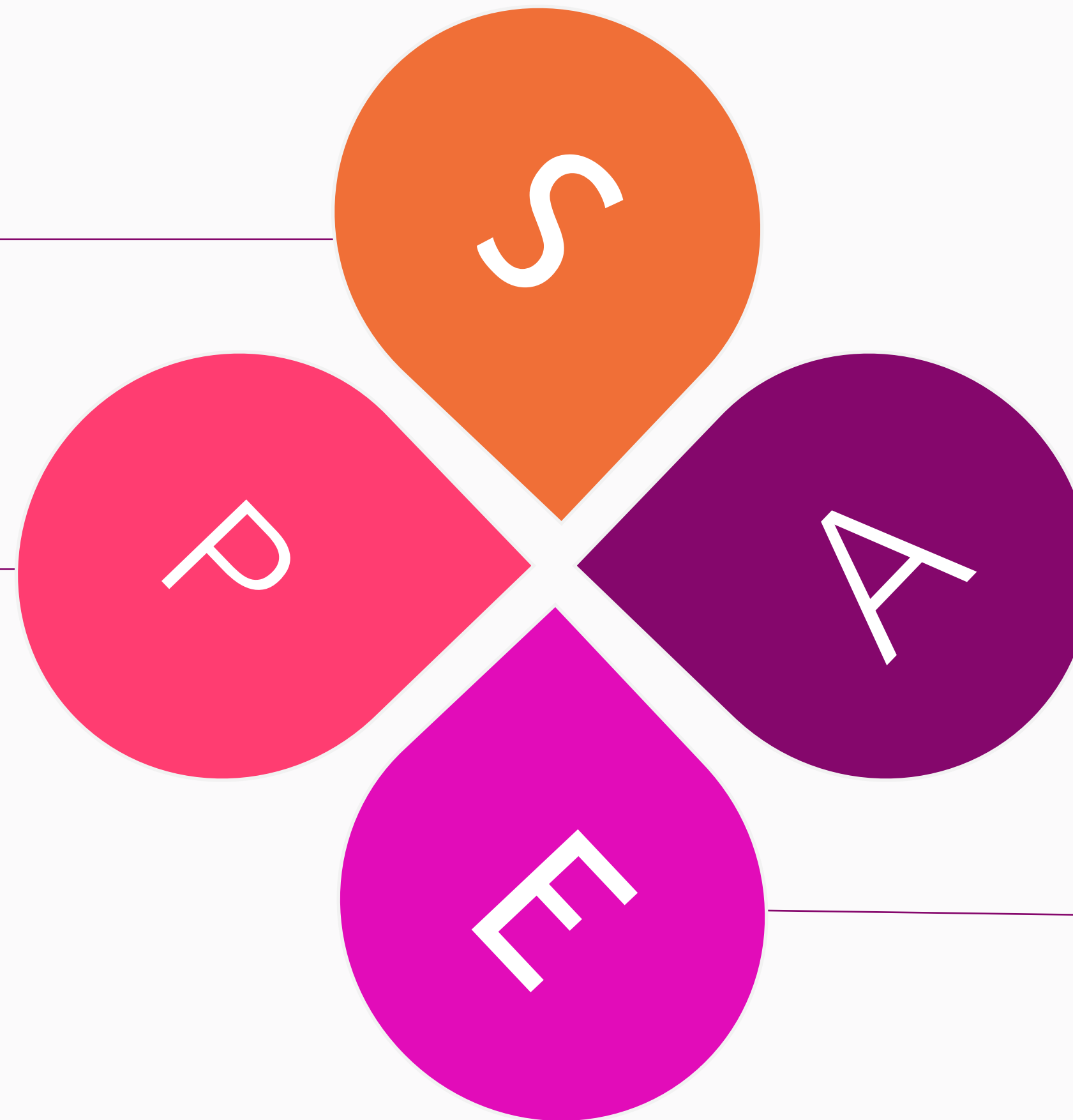
O compilador de Gleam tem como objetivo oferecer feedbacks claros e uteis sobre qualquer tipo de problema encontrado

Performática

Gleam foi construída em cima da máquina virtual do Erlang e trás consigo toda a otimização oferecida por ela

Erlang

Gleam compila para Erlang e tem como objetivo tornar o uso de códigos escritos em outras linguagens BEAM o mais fácil possível



Olá mundo!

O Olá mundo em Gleam

```
import gleam/io

pub fn main() {
  io.println("Hello, world!")
}
```

Em Gleam, declaramos uma função nomeada usando a keyword `fn`, seguida pelo nome da função, seus parâmetros e por fim, seu corpo delimitado por chaves.

Já funções anônimas são criadas com a sintaxe `fn(argumentos) {corpo da função}`

