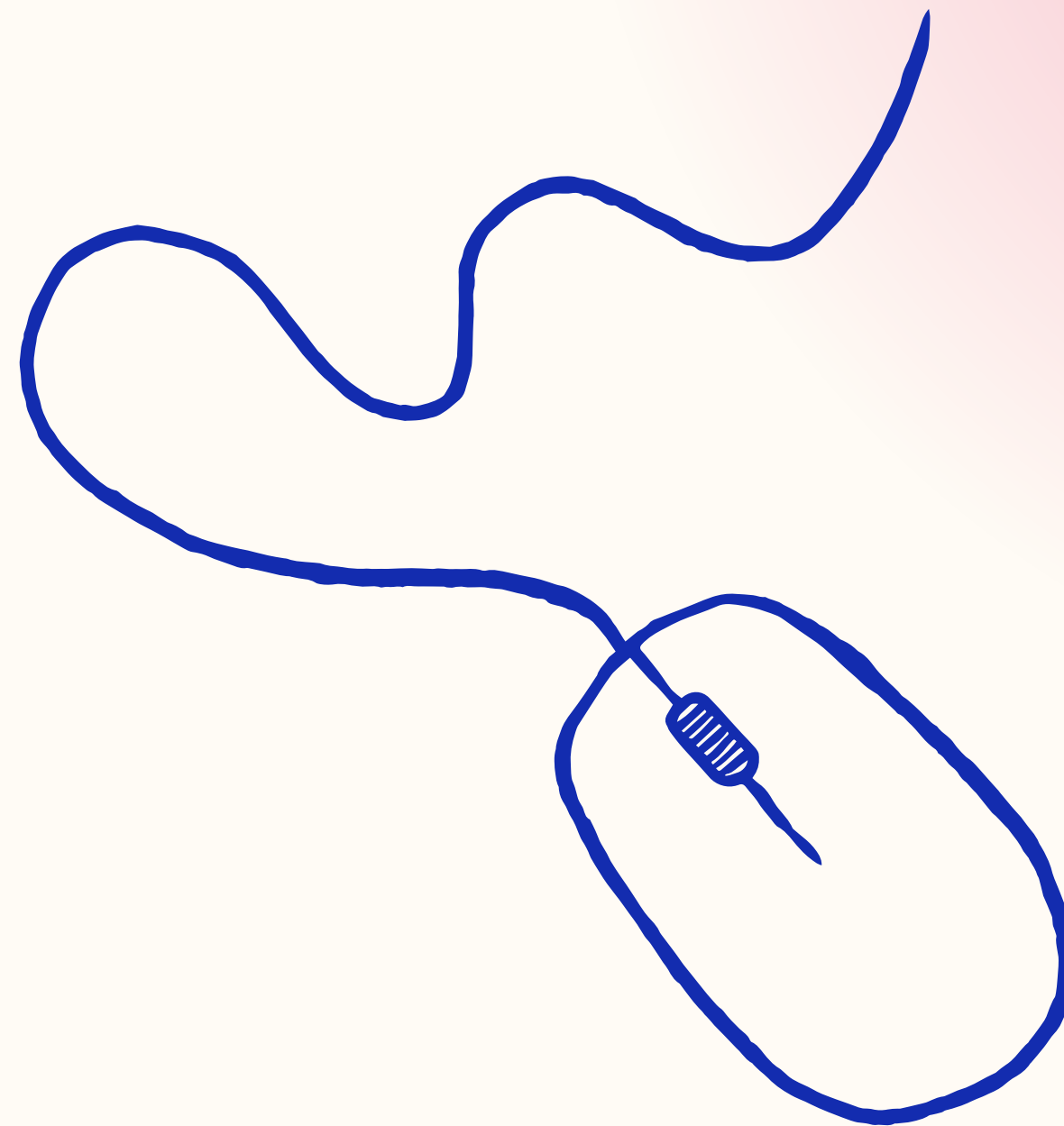


stayLang

Linguagem de
programacao



Pedro
Altobelli

- 01 – Motivação
- 02 – Características
- 03 – Curiosidades
- 04 – Exemplos



Motivação

Linguagem simples

Símbolos e nome de comandos não muito usados em outras linguagens

Operações matemáticas mais complexas

Características

Operação de potencia com ^

Criação de variavel com “variable”

Print de variaveis com “Log()”

```
variable a int = 100  
variable b int = 3  
variable power int = (45-a/4)^(b+9)  
Log(power)
```

Características

Scan de variavel com scanf()

Loop com “during ~(condicao)~ []”

```
n = scanf()  
f = 1  
during ~n < 10~ [  
    f = f + 1  
    n = n + 1  
]  
Log(f)
```

Características

Condição com “condition” e
“otherway”

```
condition a < 10 [  
    b = 10  
] otherway [  
    b = 45  
]
```

Curiosidades

A definição de variáveis na stayLang se baseia na declaração de variáveis em Go feita com “var X type”

Das 20 linguagens de programação mais famosas do mundo, nenhuma utiliza colchetes do tipo “[]” para definição de blocos (<https://www.tiobe.com/tiobe-index/>)

Exemplos

Utilizando os arquivos `bash compilerun.sh` e `FlexBisonrun.sh` no repositório da linguagem é possível fazer testes com scripts seus

Arquivos em `.stay` devem ser colocados na pasta `staylang_scripts`

Exemplos

Fibonacci.stay

```
1  variable n int = scanf()
2  variable a int = 0
3  variable b int = 1
4  variable i int = 1
5
6  condition n<19 [
7      condition (n == 1) || (n==0) [
8          condition (n==1) [
9              Log(a)
10             Log(b)
11         ] otherway [
12             Log(a)
13         ]
14     ] otherway [
15         Log(a)
16         during ~i<n~ [
17             Log(b)
18             variable a_antigo int = a
19             a = b
20             b = a_antigo+b
21             i=i+1
22         ]
23     ]
24 ] otherway [
25     Log(n)
26 ]
27
```

fermat.stay

```
1  variable n int = scanf()
2  variable i int = 3
3  variable x int = scanf()
4  variable y int = scanf()
5  variable z int = 4
6  variable true int = 1
7  variable false int = 0
8  variable left int
9  variable right int
10
11 condition n < 3 [
12     Log(n)
13 ] otherway [
14     during ~i<n~ [
15         left= x^i + y^i
16         right=z^i
17         Log(left)
18         Log(right)
19         condition left==right [
20             Log(true)
21         ]otherway[
22             Log(false)
23         ]
24         i = i+1
25     ]
26 ]
27
```

Obrigado!



Pedro Altobelli