

HADDOCK

FERRAMENTA DE DOCUMENTAÇÃO PARA HASKELL

Rogério Pontes

October 8, 2015

Universidade do Minho

O QUE SÃO COMENTÁRIOS ?

Texto livre não interpretado pelo compilador com o propósito de **auxiliar** a perceber a funcionalidade do código.

PORQUÊ ?

- Trabalhamos em equipas.
- A complexidade de uma base de código cresce exponencialmente.
- Com o tempo esquecemos o objetivo do código.

```
-- Soma dois números  
f :: Int -> Int -> Int  
  
{--  
Subtrai dois números  
--}  
g :: Int -> Int -> Int
```

WHAT IS DIZ?

```
f _ [] = []  
f p xs = if p (head xs) then (f p (tail xs)) else xs
```

```
dropWhile :: (a -> Bool) -> [a] -> [a]
dropWhile _ [] = []
dropWhile p xs
    | p (head xs) = dropWhile p (tail xs)
    | otherwise = xs
```

DROPWHILE

```
-- | Removes elements from a list until the condition
-- | no longer holds.
--
-- > dropWhile (< 3) [1,2,3,4,5,1,2,3] == [3,4,5,1,2,3]
-- > dropWhile (< 2) [1,2,3] = [2,3]
-- > dropWhile (< 9) [1,2,3] == []
-- > dropWhile (< 0) [1,2,3] == [1,2,3]
--
```

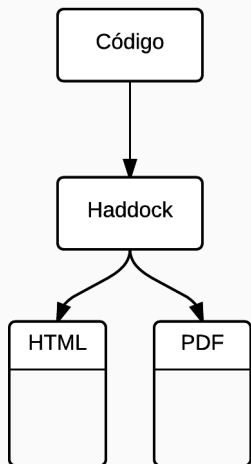
```
dropWhile :: (a -> Bool) -> [a] -> [a]
```

O QUE É O HADDOCK ?

Haddock, a tool for automatically generating documentation from annotated Haskell source code. It is primarily intended for documenting libraries, but it should be useful for any kind of Haskell code ¹.

¹Haddock

COMO FUNCIONA ?



DECLARAÇÕES

```
-- | A função 'foo'  faz um cálculo com o argumento.

{-|  A função 'foo'
retorna o resultado de x.
-}

-- Aceita um argumento do tipo Int.
foo :: Int  -- ^ Argumento Int
    -> Int  -- ^ Resultado
foo x = x * x
```

```
-- | Dois tipos de documentação
--
-- @
--     foo x = x + x
-- @
--
-- > bar x = x * 42
```

```
--  
-- >>> fib 10  
-- 55  
--  
-- >>> putStrLn "foo\nbar"  
-- foo  
-- bar
```

```
-- |  
-- = Cabeçalho nível 1 __bold__  
--  
-- == Cabeçalho nível 2  
--  
-- === Cabeçalho nível 3
```

```
> cabal init  
> cabal haddock  
> cabal haddock --haddock-option=--latex
```

- <https://www.haskell.org/haddock/>
- <https://www.youtube.com/watch?v=V0t9DBv0ZH0>
- <https://hackage.haskell.org/package/pipes-4.1.6/docs/Pipes-Tutorial.html>
- <http://blog.codinghorror.com/code-tells-you-how-comments-tell-you-why/>