




O que é Closure ?



Closures são funções anônimas (não tem o nome especificado).

São blocos de código salvos em variáveis para serem chamados num momento específico, dando mais flexibilidade ao código.

```
var closureName: (ParameterTypes) -> ReturnType
```





```
func comprarFarinha(tortaFinalizada: (_ farinha: String) -> Void) {  
}  
comprarFarinha { (farinha) in  
}
```

```
arrayCitiesFiltered = arrayCitiesFiltered.filter({ (city) -> Bool in  
    return city.pais == filterCountry  
})
```



```
present(viewDetail, animated: true) {  
    print("Complete")  
}
```



```
var jediManeiro = {(name:String) -> String in  
    return "Que a força esteja com você \(name)"  
}  
let frase = jediManeiro("Luke") //Que a força esteja com você Luke
```

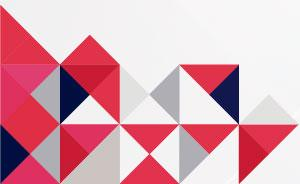




```
typealias CompletionBlock = (_ id: String) -> Void


func load(onComplete: CompletionBlock) {
    print("A")
    sleep(3)
    onComplete("B")
}

load { (id) in
    print(id)
}
```






O que é Tuple ?



Um grupo de diferentes valores, representados por um.

Como se fosse um objeto sem classe definida. Normalmente utilizada quando não há a necessidade de criação de uma classe específica.

Parênteses



```
var person = (firstName: "John", lastName: "Smith")  
var firstName = person.firstName // John  
var lastName = person.lastName // Smith
```



Parênteses



```
func getSeparated(name: String) -> (firstName: String, lastName: String) {  
    let arrayNames = name.components(separatedBy: " ")  
    if arrayNames.count > 1 {  
        return (firstName: arrayNames.first!, lastName: arrayNames.last!)  
    }  
    return (firstName: arrayNames.first!, lastName: "")  
}  
  
let separated = getSeparated(name: "Narlei Moreira")  
print(separated.firstName) // Narlei  
print(separated.lastName) // Moreira
```



Let's code!

Leia Mais

<https://docs.swift.org/swift-book/LanguageGuide/Closures.html>

https://docs.swift.org/swift-book/ReferenceManual/Types.html#grammar_tuple-type



Exercícios

Exercício 1:
Crie um código:

Crie uma função que deve receber o valor do produto e o percentual de desconto. Ela deve retornar o valor do desconto E o valor do produto com o desconto aplicado. Use tuple.

Exercício 2:
Crie um código:

Crie uma função que deve receber uma lista de nomes e o nome a ser buscado. Deve retornar quantas ocorrências havia na lista e o array com essas ocorrências. Use tuple.



Exercícios

Exercício 3:
Crie um código:

Dona Maria mandou sua filha Susan comprar farinha.
Ao mandar ela comprar farinha, ela disse que quer a Farinha da marca Nordeste.
Susan foi ao mercado, chegando lá, esqueceu a farinha,
ligou para sua mãe e pediu primeiro se era a marca Dona Benta, a mãe respondeu que não,
dai ela perguntou se era Girassol, ela respondeu que não,
então ela perguntou se era Nordeste, dai a mãe respondeu que sim, e a filha voltou pra casa.
Use closure.



Exercícios

Exercício 4:
Crie um app:

Tenha uma tela para preencher login e senha. Um botão de login e outro de cadastro.
Ao clicar em cadastro, deve informar na nova tela aberta via modal, o email e senha.
Quando clicar em cadastrar, deve retornar a tela de login, preenchendo os campos de login e senha.
Use closure.