



Funções de alta ordem nada mais são do que **funções** que recebem ou devolvem **outras funções** como **parâmetro**. O **let** é um exemplo de **função de alta ordem**, que justamente por receber uma **função como parâmetro** e **devolver outra** **ele** pode ser **encadeado**:

```
Endereco(logradouro = "Rua almirante", numero = 12, complemento = "casa")  
    .let { endereco -> "${endereco.logradouro} , ${endereco.complemento}".uppercase() }  
    .let (::println)  
  
enderecos.filter { endereco -> endereco.complemento.isNotEmpty() }  
    .forEach { endereco -> println(endereco.complemento.uppercase(Locale.getDefault())) }
```

Um outro exemplo de função de **alta ordem** seria **devolver uma função como resultado dela**. Por exemplo, eu posso **pedir parâmetros para checar uma senha** e como resultado **devolver uma função**, assim eu posso usar essa função retornada para **códigos específicos de acordo com o resultado**:

```
fun checarSenha(senha : String, numero: Int, resultado: (Boolean) -> Unit){  
    resultado(senha == "Senha" && numero == 1)  
}  
  
val checarSenha = checarSenha(senha: "senha", numero: 1){ it: Boolean  
    if(it){  
        println("Usuário aprovado para logar no sistema")  
    }else{  
        println("Usuario não aprovado")  
    }  
}
```