



Map é uma estrutura de chave e valor, onde num mapa **só é possível ter chaves únicas**, embora **valores podem ser repetidos**.

Adicionar **chave** e **valor** de um **mapa** no **Kotlin** pode ser feito usando o infix **"to"** ou usando um **Pair(x, x)**. O infix **"to"** não é aconselhável quando se trata de performance, então pra um mapa muito grande é melhor usar o **Pair**:

```
val pedidoValor = mapOf<Int, BigDecimal>(Pair(1, BigDecimal( val: "160")),  
2 to BigDecimal( val: "730"), 3 to BigDecimal( val: "95"), Pair(4, BigDecimal( val: "333.60")))
```

Valores do mapa são buscados pela chave, percorrer o mapa também é tranquilo com um **"for"** mas precisamos usar um **Entry** (é uma interface que oferece uma implementação para iterar pelo mapa usando nossa chave e valor especificadas **permitindo trabalhar de maneira isolada com a chave e o valor**):

```
val pedido = pedidoValor.getOrElse( key: 2) { println("Pedido inexistente") }  
  
for(pedido: Map.Entry<Int, BigDecimal> in pedidoValor){  
    val numeroPedido = pedido.key  
    val total = pedido.value  
  
    println("Numero do pedido: $numeroPedido - total: $total")  
}
```

MainKt X
"C:\Program Files\Java\jdk-15\bin\java.exe"
Numero do pedido: 1 - total: 160
Numero do pedido: 2 - total: 730
Numero do pedido: 3 - total: 95
Numero do pedido: 4 - total: 333.60

Para adicionar alterar a estrutura do **mapa** (adicionar ou remover elementos) ele precisa ser **mutável**, a partir daí é possível usar métodos como **put** e **remove**:

```
val pedidoValor : MutableMap<Int, BigDecimal> = mutableMapOf(Pair(1, BigDecimal( val: "160")),  
2 to BigDecimal( val: "730"), 3 to BigDecimal( val: "95"), Pair(4, BigDecimal( val: "333.60")))  
  
pedidoValor[5] = BigDecimal( val: "999")  
pedidoValor.put(6, BigDecimal( val: "675"))  
  
pedidoValor.remove( key: 2)  
pedidoValor.remove(1, BigDecimal( val: "160"))
```

É possível fazer **filtros** no mapa usando tanto a **chave** como o **valor**:

```
pedidoValor.filter { (chave, valor) -> chave in 1 <= .. <= 5 && valor >= BigDecimal( val: "120") }  
    .forEach{(chave, valor) -> println("Chave: $chave - Valor = $valor")}
```

Também é possível fazer operações aritméticas com mapas (**menos/minus "-"**, **mais/plus "+"** e etc..)