

Cheat Sheet de Kotlin: O Guia Completo

Se você está procurando uma solução para otimizar seus códigos e melhorar sua performance de desenvolvimento de software, o Kotlin é a escolha certa. A linguagem de programação foi criada pela JetBrains em 2011 e desde então vem ganhando cada vez mais espaço no mundo da tecnologia.



by Leonardo do Amaral



Made with Gamma

O que é Kotlin?

O Kotlin é uma linguagem de programação baseada em Java que pode ser compilada e executada tanto em JVM quanto em JavaScript. O objetivo da linguagem é eliminar verbosidade e adicionar funcionalidades que ajudam o programador a evitar erros e se tornar mais produtivo.

Características do Kotlin

Concisa

O Kotlin reduz a quantidade de código necessário para uma funcionalidade, ajudando a manter a organização, legibilidade e clareza no código.

Segura

Com menos espaço para erros, a segurança do código é aumentada, já que muitos deles resultam de problemas de tipagem.

Interoperável

O Kotlin pode ser facilmente integrado com o código-fonte Java existente, tornando a migração para o Kotlin simples e fácil.

Variáveis

| Declaração | Tipo | Descrição |
|------------|----------|--|
| val | Imutável | Armazena valores que não podem ser alterados após a inicialização. |
| var | Mutável | Armazena valores que podem ser alterados a qualquer momento. |

Variáveis

| Tipo de Variável | Sintaxe | Exemplo |
|----------------------|---|-------------------------|
| Inteiro | <code>var x: Int = 5</code> | <code>x = 10</code> |
| Booleano | <code>var y: Boolean = true</code> | <code>y = false</code> |
| Cadeia de Caracteres | <code>var z: String = "Hello World!"</code> | <code>println(z)</code> |

Operadores

1 Operadores Aritméticos

São operadores matemáticos que retornam um valor numérico como resultado.

2 Operadores Lógicos

São utilizados para testar múltiplas condições lógicas e retornar um valor booleano como resultado.

3 Operadores de Atribuição

São utilizados para atribuir valores a variáveis.

4 Operadores de Comparação

São utilizados para comparar valores e retornar um valor booleano como resultado.

Operadores

1 Aritméticos

Soma (+), Subtração (-),
Divisão (/), Multiplicação
(*), e Resto da Divisão (%)

2 Lógicos

AND (&&), OR (||), e NOT (!)

3 Atribuição

=, +=, -=, *=, e /=

Classes

Classes são os blocos de construção da programação orientada a objetos. Elas encapsulam métodos e variáveis em um único objeto.

Atributos

- São as características que definem objetos.
- São declarados na classe e inicializados no construtor.
- Exemplo: nome, idade, endereço.

Métodos

- São as funções que um objeto pode realizar.
- São declarados na classe e acessados pelo objeto.
- Exemplo: correr(), falar(), dormir().

Estruturas de controle de fluxo

if/else

Verifica uma condição e executa um bloco de código se a condição for verdadeira, caso contrário ele executa o bloco do else.

for

Executa uma ação para cada elemento de uma lista, array, ou uma gama de valores.

while

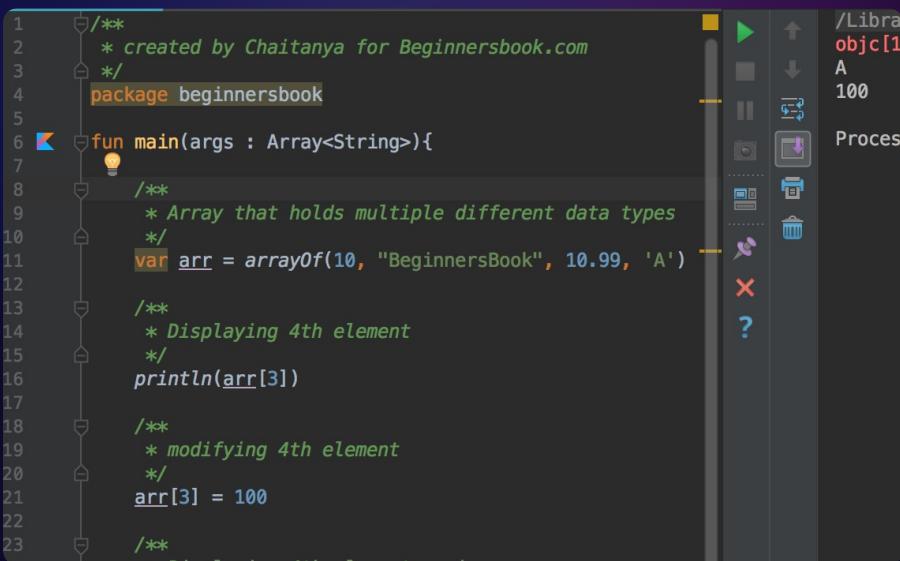
Executa um bloco de código enquanto uma condição específica for verdadeira.

Estruturas de Controle de Fluxo

| Nome | Descrição | Exemplo |
|---------|--|--|
| IF/ELSE | Testa uma condição e executa um bloco de código se ela for verdadeira. | <code>if (x > 10) {"x é maior que 10"}</code> |
| WHILE | Executa um bloco de código enquanto uma condição for verdadeira. | <code>while (x < 10) {x++}</code> |
| FOR | Executa um bloco de código para cada elemento de uma lista. | <code>for (i in lista) {println(i)}</code> |

Funções

As funções são usadas para agrupar um bloco de código em uma única unidade e executar essa unidade sempre que necessário.



```
1  /**
2   * created by Chaitanya for Beginnersbook.com
3   */
4  package beginnersbook
5
6  fun main(args : Array<String>){
7      /**
8       * Array that holds multiple different data types
9       */
10     var arr = arrayOf(10, "BeginnersBook", 10.99, 'A')
11
12     /**
13      * Displaying 4th element
14      */
15     println(arr[3])
16
17     /**
18      * modifying 4th element
19      */
20     arr[3] = 100
21
22     /**
23      * 
```

Exemplo de Funções em Kotlin

Padrões de Projeto em Kotlin

Singleton

Garante que apenas uma instância de uma classe seja criada e fornecida globalmente.

Observer

Permite que um objeto observe o estado de outro objeto e seja notificado sobre alterações.

Strategy

Permite que o algoritmo que será usado por uma classe a ser escolhido em tempo de execução.

Exemplos de código em Kotlin

Definindo uma variável

```
var nomeDoObjeto: TipoDoObjeto =  
    valorDoObjeto
```

Declarando uma função

```
fun nomeDaFuncao(parametro1: TipoDeDado1,  
    parametro2: TipoDeDado2): TipoDeRetorno {  
    /*codigo da funcao* */}
```

Usando um operador

```
resultado = valor1 + valor2
```

Criando uma Classe

```
class NomeDaClasse (propriedade1:Tipo,  
    propriedade2:Tipo){/*codigo da classe*/}
```