



Exercícios de Kotlin - semana 04

Create by: *João Paulo - Tutor de Android*

Funções Definidas pelo Usuário:

Explicando para uma criança: Uma função é um pedaço de código que faz uma tarefa específica. Por exemplo, podemos ter uma função que soma dois números ou uma função que imprime uma mensagem na tela. Quando escrevemos um programa, às vezes precisamos de uma tarefa que ainda não existe em nenhuma função existente. É aí que entram as funções definidas pelo usuário! Elas nos permitem criar nossas próprias tarefas personalizadas e usar essas tarefas sempre que precisarmos.

- 1 - Crie uma função que receba dois números inteiros como argumentos e retorne a soma dos mesmos.
- 2 - Crie uma função que receba um número inteiro como argumento e retorne verdadeiro se o mesmo for par e falso caso contrário.
- 3 - Crie uma função que receba uma string como argumento e retorne verdadeiro se a mesma for palíndromo (ou seja, lida da esquerda para direita é igual a lida da direita para esquerda) e falso caso contrário.
- 4 - Crie uma função que receba um array de números inteiros como argumento e retorne o maior valor presente no array.
- 5 - Crie uma função que receba um array de strings como argumento e retorne uma nova lista contendo apenas as strings que possuem mais de 5 caracteres.

Tipos de argumento de função

Explicando para uma criança: Os argumentos de função são pedaços de informação que podemos passar para uma função para que ela possa fazer seu trabalho. Existem diferentes tipos de argumentos que podemos passar para uma função. Alguns argumentos são obrigatórios e precisam ser passados sempre que usamos a função, enquanto outros são opcionais e só precisam ser passados se quisermos que a função

use esse argumento. Também podemos passar um número variável de argumentos para uma função, o que nos permite criar funções que possam lidar com diferentes quantidades de informações.

1 - Crie uma função que receba um número variável de argumentos do tipo Int. e retorne a soma dos mesmos.

2 - Crie uma função que receba um array de números inteiros e um número inteiro n como argumentos e retorne uma nova lista contendo apenas os elementos do array que são maiores do que n.

3 - Crie uma função que receba uma função de transformação como argumento, um array de números inteiros e retorne uma nova lista contendo os elementos do array transformados pela função de transformação.

4 - Crie uma função que receba um array de strings e um caractere como argumentos e retorne uma nova lista contendo apenas as strings que começam com o caractere informado.

5 - Crie uma função que receba um array de números inteiros e um número inteiro n como argumentos e retorne uma nova lista contendo apenas os elementos do array que estão nas posições múltiplas de n.

Listas

Explicando para uma criança: Uma lista em Kotlin é como uma lista de compras, onde você pode adicionar, remover e modificar itens. Em Kotlin, uma lista pode conter muitos itens diferentes, como números, palavras ou até outras listas. Você pode adicionar um novo item na lista, remover um item existente ou até mesmo alterar um item que já está lá.

1 - Como criar uma lista em Kotlin com os números de 1 a 10?

2 - Como adicionar um elemento na posição 2 de uma lista já existente em Kotlin?

3 - Como remover o último elemento de uma lista em Kotlin?

Maps

Explicando para uma criança: Um Map em Kotlin é como um dicionário, onde cada palavra tem uma definição. Em um Map em Kotlin, cada "palavra" é uma "chave" que corresponde a um "valor". Por exemplo, podemos criar um Map que associa as cores

com seus respectivos códigos RGB. Então, podemos procurar uma cor pelo seu nome (chave) e obter o código RGB correspondente (valor).

- 1 - Como criar um Map em Kotlin com as cores e seus respectivos códigos RGB?
- 2 - Como verificar se um determinado elemento existe em um Map em Kotlin?
- 3 - Como remover um elemento específico de um Map em Kotlin?

Sets

Explicando para uma criança: Um Set em Kotlin é como uma caixa de brinquedos, onde cada brinquedo é único e não há brinquedos duplicados. Em um Set em Kotlin, cada elemento é único e não pode haver elementos duplicados. Você pode adicionar novos elementos ao Set e verificar se um elemento específico já está lá. E se quiser, pode remover um elemento específico do Set.

- 1 - Como criar um Set em Kotlin com os números de 1 a 5?
- 2 - Como verificar se um determinado elemento existe em um Set em Kotlin?
- 3 - Como remover um elemento específico de um Set em Kotlin?