

Vitor Amadeu Souza

Introdução ao

Kotlin

Parte I

© 2025 by Cerne Tecnologia e Treinamento Ltda.

© 2025 by Vitor Amadeu Souza

Nenhuma parte desta publicação poderá ser reproduzida sem autorização prévia e escrita de **Cerne Tecnologia e Treinamento Ltda.** Este livro publica nomes comerciais e marcas registradas de produtos pertencentes a diversas companhias. O editor utiliza as marcas somente para fins editoriais e em benefício dos proprietários das marcas, sem nenhuma intenção de atingir seus direitos.

Fevereiro de 2025

Direitos reservados por:

Cerne Tecnologia e Treinamento Ltda

Produção: Cerne Tecnologia e Treinamento

E-mail da Empresa: cerne@cerne-tec.com.br

Home Page: www.cerne-tec.com.br.com.br

Atendimento ao Consumidor: sac@cerne-tec.com.br

Contato com o Autor: vitor@cerne-tec.com.br



FEITO NO BRASIL

“Se quer ir rápido, vá sozinho. Se quer ir longe, vá acompanhado.”

Provérbio africano

Cerne Tecnologia

A Cerne Tecnologia tem uma equipe preparada para desenvolvimento de projetos eletrônicos em diversas áreas: Médica, Entretenimento, Industrial, Robótica, Científica, Automobilística, Aeronáutica, etc. Trabalhamos com tecnologia microcontrolada usando o PIC, ARM, AVR, 8051, dsPIC, PIC24, PIC32 além do Arduino, Raspberry, Beaglebone etc. Desenvolvemos o projeto desde sua concepção até a entrega do produto final, passando pelas etapas de esquema elétrico, protótipo e desenvolvimento de circuito impresso.

Desenvolvemos aplicativos para smartphones/tablets Android, iOS, Blackberry, Windows Phone e no desenvolvimento de softwares a nível PC para plataforma Windows, usando ferramentas como o Visual Basic, C# e C++.

Atuamos na parte de montagem de placas, onde podemos fornecer ambos os serviços de desenvolvimento de projetos e produção ou apenas um destes.

Desenvolvemos esquemas elétricos e layout de PCI, tanto em tecnologia convencional como SMD.

Temos a flexibilidade de customizar um de nossos produtos, de modo a atender a uma necessidade específica do cliente, tornando o custo de desenvolvimento menor se comparado a construção de um projeto desde a sua fase inicial.

Desenvolvemos e fornecemos kits didáticos para diversos microcontroladores além de apostilas, livros e e-books.

Na hora de desenvolver um projeto ou equipar seu laboratório não hesite em nos contatar. Entre em contato conosco através do endereço cerne-tec.com.br para obter mais informações.



Sumário

Capítulo I – Metodologia de desenvolvimento	6
1. Introdução	6
 Capítulo II – Programação em Kotlin	 7
1. Introdução	7
2. Ambientes de desenvolvimento	10
3. Olá, Mundo!	12
4. Variáveis mutáveis e imutáveis	13
5. Tipos de Dados	14
6. Condicional If-Else	16
7. Operador Ternário (If simplificado)	18
8. When (Switch do Kotlin)	19

Capítulo I

Metodologia de desenvolvimento

1. Introdução

Este livro tem como objetivo ensinar a programação utilizando Kotlin, explorando seus recursos de forma prática e aplicada. Ao longo desta série, diversas aplicações serão desenvolvidas, permitindo uma compreensão aprofundada da linguagem e de suas principais funcionalidades. Cada volume abordará temas específicos, conforme detalhado no sumário, proporcionando um aprendizado progressivo e estruturado.

Capítulo II

Programação em Kotlin

1. Introdução

Kotlin é uma linguagem de programação moderna, concisa e segura, desenvolvida pela JetBrains em 2011. Desde então, ganhou popularidade rapidamente, especialmente após ser adotada como uma linguagem oficial para o desenvolvimento de aplicativos Android em 2017. Combinando características de linguagens orientadas a objetos e funcionais, Kotlin oferece uma experiência de desenvolvimento eficiente e agradável, tornando-se uma escolha popular para desenvolvedores em diversos cenários.

Uma das principais vantagens do Kotlin é sua interoperabilidade com Java. Como Kotlin é totalmente compatível com o ecossistema Java, os desenvolvedores podem integrar facilmente bibliotecas e frameworks existentes em seus projetos Kotlin. Isso permite uma transição suave para quem já está familiarizado com Java, além de possibilitar a migração gradual de

projetos legados para Kotlin. Essa interoperabilidade é um dos fatores que contribuíram para a rápida adoção da linguagem.

A sintaxe do Kotlin é concisa e expressiva, o que reduz a quantidade de código repetitivo necessário para realizar tarefas comuns. Por exemplo, em Kotlin, a declaração de variáveis é simplificada com o uso das palavras-chave `val` (para variáveis imutáveis) e `var` (para variáveis mutáveis). Além disso, a linguagem oferece recursos como `data classes`, que permitem a criação de classes para armazenar dados com menos código, e funções de extensão, que possibilitam adicionar funcionalidades a classes existentes sem modificar seu código-fonte.

Kotlin também é uma linguagem multiparadigma, suportando programação orientada a objetos, funcional e procedural. Isso permite que os desenvolvedores escolham o estilo de programação mais adequado para cada situação. Recursos como lambdas, funções de alta ordem e imutabilidade facilitam a escrita de código funcional, enquanto classes, herança e polimorfismo atendem às necessidades da programação orientada a objetos.

No contexto do desenvolvimento Android, Kotlin se tornou uma alternativa preferida em relação a Java devido à sua

produtividade e segurança. Ferramentas como o Android Studio oferecem suporte nativo para Kotlin, incluindo recursos como conversão automática de código Java para Kotlin. Além disso, a comunidade Kotlin é ativa e crescente, com uma vasta gama de bibliotecas e frameworks disponíveis para simplificar o desenvolvimento de aplicativos.

Fora do ecossistema Android, Kotlin também é utilizado em outras áreas, como desenvolvimento web (com frameworks como Ktor), backend (com Spring Boot) e até mesmo em ciência de dados. A linguagem é versátil e pode ser aplicada em diversos domínios, graças à sua flexibilidade e desempenho.

Em resumo, Kotlin é uma linguagem poderosa e moderna que combina produtividade, segurança e interoperabilidade. Sua adoção crescente no mercado de desenvolvimento de software reflete sua capacidade de atender às necessidades dos desenvolvedores, seja em projetos novos ou na modernização de sistemas legados. Para quem busca uma linguagem que une simplicidade e robustez, Kotlin é uma escolha excelente e promissora para o futuro da programação.

2. Ambientes de desenvolvimento

Kotlin pode ser desenvolvido em diversos ambientes, dependendo da necessidade do projeto. O IntelliJ IDEA, criado pela JetBrains, é a IDE oficial e oferece suporte completo à linguagem, com funcionalidades como autocompletar, refatoração inteligente e depuração avançada. Outra opção popular é o Android Studio, amplamente utilizado para desenvolvimento mobile, já que o Kotlin é a linguagem recomendada pelo Google para Android. Para quem busca um ambiente mais leve, o Visual Studio Code pode ser usado com extensões específicas que adicionam suporte à linguagem. Além das IDEs tradicionais, existem ambientes online como o Kotlin Playground, ideal para testar trechos de código sem necessidade de instalação, e o Replit, que permite rodar projetos Kotlin diretamente no navegador. Também é possível programar em Kotlin via linha de comando, utilizando o compilador Kotlin (kotlinc) para compilar e executar arquivos .kt. Com tantas opções disponíveis, o desenvolvedor pode escolher o ambiente que melhor se adapta ao seu fluxo de trabalho, garantindo produtividade e eficiência no desenvolvimento de aplicações.