

Vitor Amadeu Souza

Introdução ao

TypeScript

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

“Conhecimento é poder.”

Francis Bacon

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 TypeScript	 7
1. Introdução	7
2. Ambientes de desenvolvimento	9
3. Declaração de Variáveis	12
4. Tipos Primitivos	14
5. Arrays	15
6. Enum	17
7. Tuplas	18
8. Função Simples	20

Capítulo I

Metodologia de desenvolvimento

1. Introdução

Este livro tem como objetivo ensinar o desenvolvimento web utilizando TypeScript, 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 TypeScript

1. Introdução

TypeScript é uma linguagem de programação que estende o JavaScript, adicionando tipagem estática opcional e outros recursos avançados para facilitar o desenvolvimento de aplicações robustas e escaláveis. Criado pela Microsoft, o TypeScript é um superconjunto de JavaScript, o que significa que todo código JavaScript válido também é válido em TypeScript, mas com a adição de um sistema de tipos que permite detectar erros antes mesmo da execução do código. Essa característica reduz a ocorrência de falhas inesperadas, tornando o desenvolvimento mais seguro e previsível.

Uma das principais vantagens do TypeScript é a detecção precoce de erros. Como a linguagem permite definir explicitamente os tipos das variáveis, funções e objetos, o compilador pode alertar sobre possíveis incompatibilidades antes mesmo que o código seja executado. Isso evita problemas comuns de JavaScript, como erros relacionados a tipos inesperados ou referências a propriedades

inexistentes. Além disso, a ferramenta de autocompletar e sugestão de código, oferecida por editores como o Visual Studio Code, melhora significativamente a produtividade dos desenvolvedores, pois reduz a necessidade de consultas constantes à documentação.

Outro grande benefício do TypeScript é a melhor organização do código, especialmente em projetos grandes. Com recursos como interfaces, classes e módulos, o desenvolvimento se torna mais estruturado e modular, facilitando a manutenção e reutilização de código. O suporte a programação orientada a objetos e a compatibilidade com padrões modernos de JavaScript fazem com que a transição de equipes para TypeScript seja natural e sem grandes obstáculos. Além disso, a compatibilidade total com JavaScript permite a adoção gradual da linguagem, tornando possível a migração de projetos existentes sem a necessidade de reescrevê-los completamente.

Por fim, TypeScript melhora a colaboração entre desenvolvedores, pois a tipagem explícita e a documentação integrada facilitam a compreensão do código, mesmo para aqueles que não o escreveram originalmente. Como resultado, a adoção de TypeScript vem crescendo rapidamente em empresas e comunidades

de desenvolvimento, consolidando-se como uma das escolhas mais eficientes para o desenvolvimento de aplicações web e backend modernas.

2. Ambientes de desenvolvimento

Existem diversos ambientes de desenvolvimento que suportam TypeScript, cada um adequado a diferentes necessidades e níveis de complexidade. Desde editores online para testes rápidos até IDEs completas para desenvolvimento profissional, a escolha do ambiente pode impactar diretamente a produtividade e a experiência do desenvolvedor.

[1] TypeScript Playground – Ambiente oficial e online do TypeScript, ideal para testes rápidos e aprendizado, sem necessidade de instalação. Permite escrever e compilar código TypeScript diretamente no navegador, mostrando erros e sugestões em tempo real.

[2] Visual Studio Code (VS Code) – Editor de código leve e altamente popular, com suporte nativo a TypeScript. Oferece autocompletar, depuração e integração com NPM, Webpack e outros bundlers, sendo amplamente utilizado para desenvolvimento profissional.

[3] StackBlitz e CodeSandbox – Ambientes online para desenvolvimento completo de aplicações, com suporte a TypeScript, frameworks como Angular, React e Vue, além de integração com GitHub e pacotes NPM.

[4] Replit – Plataforma colaborativa que permite rodar TypeScript diretamente no navegador, sendo útil para projetos full-stack combinando backend e frontend.

[5] Node.js com TypeScript – Para desenvolvimento backend, é comum configurar TypeScript no Node.js usando ts-node e tsconfig.json, permitindo a execução direta do código sem precisar transpilar manualmente.

Nesta literatura, será utilizado o TypeScript Playground como ferramenta principal para demonstrações e exemplos de código. Sua interface simplificada permite que o leitor experimente diretamente os conceitos apresentados sem precisar configurar um ambiente local. Isso facilita a aprendizagem inicial e torna os exemplos mais acessíveis, garantindo que o foco esteja na lógica e na estrutura do TypeScript, sem distrações com instalação ou configuração de ferramentas. Para acessar tal ferramenta, visite o link abaixo: