

# .NET Core Documentation

## Guia Completo para Desenvolvimento

### 1. Introducao ao .NET Core

O .NET Core e uma plataforma de desenvolvimento multiplataforma, open-source e de alto desempenho para criacao de aplicacoes modernas.

#### Principais Caracteristicas:

- â€ Multiplataforma: Windows, Linux, macOS
- â€ Open Source: Codigo aberto no GitHub
- â€ Alto Desempenho: Otimizado para aplicacoes modernas
- â€ Modular: Arquitetura baseada em pacotes NuGet

#### Versoes Suportadas:

- â€ .NET Core 3.1 (LTS)
- â€ .NET 5.0+
- â€ .NET 6.0 (LTS)
- â€ .NET 7.0+

## 2. Instalacao e Configuracao

### Instalando o .NET Core SDK:

1. Acesse: <https://dotnet.microsoft.com/download>
2. Baixe o SDK para seu sistema operacional
3. Execute o instalador
4. Verifique a instalacao: `dotnet --version`

### Comandos Essenciais do CLI:

- â€¢ `dotnet new console` - Criar projeto console
- â€¢ `dotnet new webapi` - Criar Web API
- â€¢ `dotnet build` - Compilar projeto
- â€¢ `dotnet run` - Executar aplicacao
- â€¢ `dotnet test` - Executar testes
- â€¢ `dotnet publish` - Publicar aplicacao

### Estrutura de Projeto:

- â€¢ `Program.cs` - Ponto de entrada
- â€¢ `.csproj` - Arquivo de projeto
- â€¢ `appsettings.json` - Configuracoes

## 3. Conceitos Fundamentais

### Dependency Injection (DI):

O .NET Core possui um container de DI integrado que permite injeção de dependências nativa.

```
// Exemplo de registro de serviço  
builder.Services.AddScoped<IUserService, UserService>();
```

### Configuration System:

Sistema flexível de configuração que suporta múltiplas fontes:

â€¢ appsettings.json

â€¢ Environment Variables

â€¢ Command Line Arguments

â€¢ Azure Key Vault

### Logging:

Sistema de logging integrado com suporte a múltiplos providers:

â€¢ Console, Debug, EventSource

â€¢ Terceiros: Serilog, NLog

## 4. Desenvolvimento de APIs

### Minimal APIs (.NET 6+):

```
var app = WebApplication.Create(args);
app.MapGet("/", () => "Hello World!");
app.Run();
```

### Controllers Tradicionais:

```
[ApiController]
[Route("api/[controller]")]
public class UsersController : ControllerBase
{
    [HttpGet]
    public IActionResult Get() => Ok(users);
}
```

### Middleware Pipeline:

- ⌘ Request/Response processing
- ⌘ Authentication/Authorization
- ⌘ Exception Handling
- ⌘ Static Files
- ⌘ CORS