

# Haki do Código

## Desperte o Angular Supremo



# Bruno Wilson

# 1

## Sob as Estrelas do Angular

Introdução ao framework



# Introdução ao framework

**Angular é um framework desenvolvido pelo Google.**

**Para criar aplicações web SPA (Single Page Applications).**

**Utiliza TypeScript como linguagem principal e oferece uma arquitetura baseada em componentes.**

# 2

## Preparando o Navio

Configuração do ambiente e  
ferramentas



# Configuração do ambiente

**Instale o Node.js:**

Baixe de <https://nodejs.org>  
e instale a versão LTS.

**Instale o Angular CLI:**

```
npm install -g @angular/cli
```

codesnap.dev

# Configuração do ambiente

**Crie seu primeiro projeto:**

```
ng new haki-angular  
cd haki-angular  
ng serve
```

**Abra <http://localhost:4200>**

**Para ver sua aplicação  
navegando pela primeira vez.**






# 3

## Cartografia Ágil

Comandos básicos do Angular  
CLI

# Comandos básicos do Angular

**Gerar um componente:**



```
ng generate component tarefa
```

**Gerar um serviço:**



```
ng generate service tarefa
```

**Rodar o servidor de desenvolvimento:**



```
ng serve
```



# 4

## Construindo o Navio

Componentes do Angular

# Componentes do Angular

**Exemplo de componente:**

```
ng generate component cabecalho
```

**No arquivo cabecalho.component.html:**

```
<h1>Meu Painel de Tarefas</h1>
```

**No app.component.html:**

```
<app-cabecalho></app-cabecalho>
```

**Isso adiciona o componente Cabecalho à página principal.**





# 5

## Mares dos Módulos

### Módulos do Angular

# Módulos do Angular

Crie um novo módulo:

```
ng generate module tarefas
```

codesnap.dev

Importe em app.module.ts:

```
import { TarefasModule } from '../tarefas/tarefas.module';

@NgModule({
  imports: [TarefasModule]
})
```

codesnap.dev

**Dividir a aplicação em módulos melhora a organização e performance.**





# Guardiões do Código

Serviços e Injeção de  
Dependência

# Serviços e Injeção de Dependência

Crie um serviço de tarefas:



```
ng generate service tarefa
```

No serviço (tarefa.service.ts):



```
@Injectable({ providedIn: 'root' })  
export class TarefaService {  
  private tarefas: string[] = ['Estudar Angular'];  
  
  getTarefas() {  
    return this.tarefas;  
  }  
  
  adicionarTarefa(tarefa: string) {  
    this.tarefas.push(tarefa);  
  }  
}
```



# Serviços e Injeção de Dependência

Uso no componente:



```
constructor(private tarefaService: TarefaService) {}
```



# 7

## Mapa do Tesouro

### Roteamento Angular



# Roteamento Angular

Habilitar roteamento ao criar projeto:

```
ng new haki-angular --routing
```

Definir rotas:

```
const routes: Routes = [  
  { path: 'home', component: HomeComponent },  
  { path: 'tarefas', component: TarefaComponent },  
];
```

# Roteamento Angular

No HTML:

```
<a routerLink="/tarefas">Ver Tarefas</a>  
<router-outlet></router-outlet>
```



# 8

## Vínculos de Haki

Ligação de Dados (Data Binding)


# Ligacao de Dados

## Interpolacao:



```
<p>{{ titulo }}</p>
```

## Property Binding:



```
<img [src]="imageUrl" />
```



# Ligacao de Dados

## Event Binding:



```
<button (click)="executar()">Executar</button>
```

## Two-Way Binding:



```
<input [(ngModel)]="nome" />  
<p>Olá {{ nome }}</p>
```



# Mares Revoltos

Diretivas Angular



# Diretivas Angular

ngIf:

```
<p *ngIf="mostrar">Mostrando algo</p>
```

ngFor:

```
<li *ngFor="let t of tarefas">{{ t }}</li>
```

ngClass:

```
<p [ngClass]="{importante: tarefa.importante}">{  
  { tarefa.nome }}</p>
```

# 10

## Alquimia do Código

### Pipes Angular



# Pipes Angular

## Uso de pipe embutido:

```
<p>{{ hoje | date }}</p>  
<p>{{ valor | currency:'BRL' }}</p>
```

## Criando pipe personalizado:

```
ng generate pipe primeiraLetraMaiuscula
```

# Pipes Angular

No pipe gerado:

```
transform(valor: string): string {  
  return valor.charAt(0).toUpperCase() + valor.slice(1);  
}
```

Uso:

```
<p>{{ 'angular' | primeiraLetraMaiuscula }}</p>
```



# AGRADECIMENTOS



# Obrigado por chegar ate aqui

Esse Ebook foi gerado por IA, e diagramado por humano.

Esse conteúdo foi gerado com fins didáticos de construção, não foi realizado uma validação cuidadosa humana no conteúdo e pode conter erros gerados por uma IA.



<https://github.com/BrunoWil/>



#### **BrunoWil - Overview**

BrunoWil has 18 repositories available. Follow their code on GitHub.

 GitHub