

Spring Core

DEVinHouse

Parcerias para desenvolver a sua carreira

SENAI

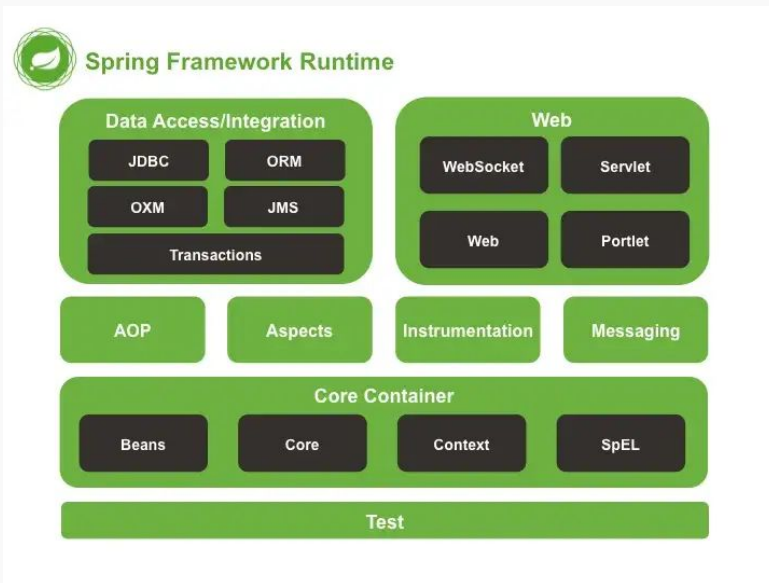
<LAB365>

AGENDA

- Core Container
- Anotação

Core Container

Dentro da arquitetura Spring, o Core Container é dividido em componentes. Os seus quatro componentes fundamentais são: Core, Beans, Expression Language (SpEL) e Context. Esses componentes são a base de todo o Core Container.



No Core se encontra toda a infraestrutura necessária que permite a implementação do container. São encontrados facilitadores no gerenciamento de classloaders, reflexão, tratamento de recursos, strings, etc.

Um bean é um objeto que é criado, gerenciado e destruído pelo container do spring, o framework é totalmente responsável por este objeto, criando, injetando suas propriedades (Injeção de dependência).

Usando como base o Core temos o módulo Beans, onde encontramos o BeanFactory, que implementa o factory. O factory é um padrão de projetos que permite às classes delegar para suas subclasses decidirem quais objetos criar. Este padrão de projeto que torna desnecessária a codificação manual de singletons (Padrão que garante a existência de instância de apenas uma classe) e nos permite o desacoplamento total do gerenciamento de dependências e ciclo de vida dos beans (Objeto de negócio) do código fonte através do mecanismo de configuração implementado neste módulo.

Expression Language (SpEL)

O componente de Expression Language é uma linguagem baseada em expressões que é utilizado como modo de configuração do Spring. Esta linguagem nos permite definir valores e comportamentos aos nossos beans definidos usando XML.

Finalmente, temos o módulo Context, onde se encontra a implementação mais avançada e comumente usada do container que é o ApplicationContext. O ApplicationContext é a interface central em um aplicativo Spring usado para fornecer informações de configuração ao aplicativo. Ele nos oferece recursos poderosos como por exemplo internacionalização, propagação de eventos, AOP, gerenciamento de recursos e acesso a funcionalidades básicas da plataforma Java EE.

@Component: Anotação genérica para qualquer componente gerenciado pelo Spring. Esta anotação faz com que o bean registrado no Spring possa ser utilizado em qualquer bean, seja ele um serviço, um DAO, um controller, etc. No nosso exemplo, ele será responsável por um Bean que representa uma entidade.

@Repository: Anotação que serve para definir uma classe como pertencente à camada de persistência.

@Service: Anotação que serve para definir uma classe como pertencente à camada de Serviço da aplicação.

@Autowired: A anotação @ Autowired fornece controle sobre onde e como a ligação entre os beans deve ser realizada. Pode ser usado para em métodos setter, no construtor, em uma propriedade ou métodos com nomes arbitrários e / ou vários argumentos.

AVALIAÇÃO DOCENTE

O que você está achando das minhas aulas neste conteúdo?

[Clique aqui](#) ou escaneie o QRCode ao lado para avaliar minha aula.

Sinta-se à vontade para fornecer uma avaliação sempre que achar necessário.





DEVinHouse

Parcerias para desenvolver a sua carreira

OBRIGADO!



<LAB365>