Injeção de dependências (ou Dependency Injection – DI) é um tipo de inversão de controle (ou Inversion of Control – IoC) que dá nome ao processo de prover instâncias de classes que um objeto precisa para funcionar. A grande vantagem desse conceito é que nós conseguimos programar voltados para interfaces e, com isso, manter o baixo acoplamento entre as classes de um mesmo projeto. Com certeza, essa característica é uma grande vantagem para a arquitetura do seu sistema, assim como é para o próprio Spring.

Essa funcionalidade é a base de todo o ecossistema Spring. É difícil pensar em Spring sem a injeção de dependências. Sendo assim, esse é um conceito que merece atenção.

Na prática, ao invés de você fazer isso:

public class ServicoCliente {

Nesse exemplo, supondo que "RepositorioCliente" é uma interface. Mas poderia ser uma classe abstrata ou mesmo uma classe concreta.

private RepositorioCliente repositorio = new RepositorioClienteImpl ( )

}

Faça isso:

public class ServicoCliente {

@Autowired

private RepositorioCliente repositorio;

...

}

A anotação @Autowired avisa ao Spring Framework para injetar uma instância de alguma implementação da interface RepositorioCliente na propriedade repositorio.

Ou seja é um padrão de projeto que nos ajuda a desacoplar o código, melhora a interpretação e melhora a distribuição de responsabilidade entre as classes e também facilita a manutenção do código.