**Dependency Injection (DI)** is a software design pattern that aims to decouple objects from their dependencies by injecting or passing those dependencies into a dependent object. It promotes flexibility, testability, and maintainability.

Here are **five free resources** where you can learn more about Dependency Injection:

1. [**GeeksforGeeks**: Their article provides an in-depth explanation of Dependency Injection, examples, advantages, and disadvantages1](https://www.geeksforgeeks.org/dependency-injectiondi-design-pattern/).
2. [**Java Design Patterns**: This resource specifically focuses on the Java implementation of Dependency Injection, emphasizing its benefits and usage](https://www.geeksforgeeks.org/dependency-injectiondi-design-pattern/)[2](https://java-design-patterns.com/patterns/dependency-injection/).
3. [**Fuad Efendi**: The article on this website delves into the concept of Dependency Injection and its role in managing dependencies between objects](https://www.geeksforgeeks.org/dependency-injectiondi-design-pattern/)[3](https://fuadefendi.com/docs/design-patterns/creational/dependency-injection/).
4. [**Medium**: Learn about Dependency Injection’s technique of supplying object dependencies and how it enhances code flexibility and testability](https://www.geeksforgeeks.org/dependency-injectiondi-design-pattern/)[4](https://medium.com/@sardar.khan299/understanding-dependency-injection-a-powerful-design-pattern-for-flexible-and-testable-code-5e1161dd37dd).
5. [**Stackify**: Their explanation covers Dependency Injection as an alternative to the service locator pattern, with practical insights](https://www.geeksforgeeks.org/dependency-injectiondi-design-pattern/)[5](https://stackify.com/dependency-injection/).

Feel free to explore these resources to deepen your understanding of Dependency Injection! 🚀