Certainly! **Mockito** is a **Java-based mocking framework** used for unit testing Java applications. [It simplifies the development of tests for classes with external dependencies by creating dummy implementations (mock objects) for interfaces or classes](https://www.vogella.com/tutorials/Mockito/article.html) [1](https://www.vogella.com/tutorials/Mockito/article.html)[2](https://www.javatpoint.com/mockito).

Here are **five free reference links** where you can learn more about Mockito:

1. **Vogella Tutorial**: This comprehensive tutorial explains how to use Mockito for writing software tests with JUnit 5 in Java. [It covers both basic and advanced use cases](https://www.vogella.com/tutorials/Mockito/article.html) [1](https://www.vogella.com/tutorials/Mockito/article.html). You can find it [here](https://www.vogella.com/tutorials/Mockito/article.html).
2. **Baeldung Mockito Series**: Baeldung offers a series of tutorials focusing on Mockito, from the basics to more advanced scenarios. [It also integrates Mockito with other testing libraries like JUnit](https://www.vogella.com/tutorials/Mockito/article.html) [3](https://www.baeldung.com/mockito-series). Dive into the series [here](https://www.baeldung.com/mockito-series).
3. **Javatpoint Tutorial**: Javatpoint provides a concise tutorial on Mockito, emphasizing its role in developing testable applications. [Learn about Mockito’s features and usage under the MIT License](https://www.vogella.com/tutorials/Mockito/article.html) [2](https://www.javatpoint.com/mockito). Explore it [here](https://www.javatpoint.com/mockito).

Remember, these resources will help you master Mockito and enhance your testing skills! 🚀