Java Bytecode:

<https://docs.oracle.com/javase/specs/jvms/se7/html/jvms-6.html>

<https://www.javaworld.com/article/2077233/bytecode-basics.html>

<https://dzone.com/articles/introduction-to-java-bytecode>

Auch mit Buch: <https://www.informit.com/articles/article.aspx?p=2024315&seqNum=11>

Spotbugs:

<https://spotbugs.readthedocs.io/en/stable/introduction.html>

<https://spotbugs.readthedocs.io/en/stable/running.html> (auch VisitorPattern)

DesignPattern:

<https://de.wikipedia.org/wiki/Besucher_(Entwurfsmuster)>

<https://de.wikipedia.org/wiki/Entwurfsmuster>

<https://sourcemaking.com/design_patterns>

<https://sourcemaking.com/design_patterns/visitor>

<https://www.philipphauer.de/study/se/design-pattern.php>

<http://www.inztitut.de/blog/glossar/design-pattern/>

<https://www.tutorialspoint.com/design_pattern/builder_pattern.htm>

<https://www.baeldung.com/java-visitor-pattern>

<https://www.baeldung.com/java-decorator-pattern>

Codeanalyse:

<https://www.pmi.org/-/media/pmi/documents/public/pdf/learning/thought-leadership/pulse/pulse-of-the-profession-2017.pdf?sc_lang_temp=en>

<https://www.perforce.com/blog/sca/what-static-analysis>

<https://de.wikipedia.org/wiki/Statische_Code-Analyse>

<https://searchwindevelopment.techtarget.com/definition/static-analysis>

<https://owasp.org/www-community/controls/Static_Code_Analysis>

Dynamisch: <https://www.viva64.com/en/t/0070/>

<https://www.computerweekly.com/answer/Dynamic-code-analysis-vs-static-analysis-source-code-testing>

<https://www.veracode.com/blog/2013/12/static-testing-vs-dynamic-testing>

<https://blog.ndepend.com/difference-static-and-dynamic-code-analysis/>

<https://www.castsoftware.com/glossary/code-analysis>

<https://www.askspoke.com/blog/it/reasons-for-it-project-failure/>