Code Quality

Code style checkers

JabRef has three code style checkers in place:

- Checkstyle for basic checks, such as wrong import order.
- Gradle Modernizer Plugin for Java library usage checks. It ensures that "modern" Java concepts are used (e.g., one should use `Deque` instead of `Stack`).
- OpenRewrite for advanced rules. OpenRewrite can also automatically fix issues. JabRef's CI toolchain does NOT automatically rewrite the source code, but checks whether OpenRewrite would rewrite something. As developer, one can execute `./gradlew rewriteRun` to fix the issues. Note that JabRef is available on the Moderne platform, too.

In case a check fails, the CI automatically adds a comment on the pull request.

Monitoring

We monitor the general source code quality at three places:

- codacy is a hosted service to monitor code quality. It thereby combines the results of available open source code quality checkers such as Checkstyle or PMD. The code quality analysis for JabRef is available at https://app.codacy.com/gh/JabRef/jabref/dashboard, especially the list of open issues. In case a rule feels wrong, it is most likely a PMD rule.
- codecov is a solution to check code coverage of test cases. The code coverage metrics for JabRef are available at https://codecov.io/github/JabRef/jabref.
- Teamscale is a popular German product analyzing code quality. The analysis results are available at https://demo.teamscale.com/findings.html#/jabref/?.

Up to date dependencies

We believe that updated dependencies are a sign of maintained code and thus an indicator of good quality. We use GitHub's dependabot to keep our versions up to date.

Moreover, we try to test JabRef with the latest Java Development Kit (JDK) builds. Our results can be seen at the Quality Outreach page.

Statistics

- Tests passing

- codecov 40%

Background literature

We strongly recommend reading following two books on code quality:

- Java by Comparison is a book by three JabRef developers which focuses on code improvements close to single statements. It is fast to read and one gains much information from each recommendation discussed in the book.
- Effective Java is the standard book for advanced Java programming. Did you know that `enum` is the recommended way to enforce a singleton instance of a class? Did you know that one should refer to objects by their interfaces?

The principles we follow to ensure high code quality in JabRef is stated at our Development Strategy.
