GOOD PRACTICE

What is Code Linting ?

DEBIOTECH

Let's start with the definition of Linting:

Linting is the automated checking of your source code for programmatic and stylistic errors. This is done by using a lint tool (otherwise known as linter). A lint tool is a basic static code analyzer. (OWASP)

Advantages

The main goal of linting is to improve overall code quality, by detecting errors that could lead to bugs or security vulnerabilities but also formatting or styling issues which can make the code less readable and maintainable. Another advantage of applying code formatting automatically through a linter is to keep your code base consistent within a project (no matter which developer wrote such or such piece of code) but also across projects (i.e. same rules for every project).

Difference with advanced Static Code Analysis

Lint tools provide a very basic static code analysis and will only detect most common mistakes. Other advanced static code analysis tools that will perform a deeper analysis can be used in conjunction with linting. But we at least recommend to start with basic linting for any project.

Automated Linting

Linting should be automated in order to be efficient. Linting should be performed on the developer machine (e.g. at compile time) so that the developer can see and fix mistakes before pushing the code to the repository. Regarding code formatting, modern IDEs provide a 'Format on Save' setting which will automatically apply formatting which will save valuable time as the developer does not need to worry about the format when writing the code. Also, do not forget to add linter checks to your CI/CD pipeline to catch any non-compliant code that was wrongly committed.

Tools Examples

At Debiotech, we have been using Dart and Flutter for a couple of years. Dart provides its own linter with many rules available. You can customize the rules list to your needs and habits. For C/C++, we recommend using Clang which complies with the C and C ++ standards more strictly than GCC. Clang has become the default compiler for Android and Mac OS X. For C/C++ code formatting, clang-format can be used. It integrates natively with Visual Studio Code (through the C/C++ extension maintained by Microsoft) for auto-formatting on save. For Java, we could recommend Spotless and checkstyle. For Python, we can list pylint and flake8.

Conclusion

We really recommend putting in place a linter tool as soon as a project starts! The earlier checks are in place, the less painful and time-consuming errors correction is.

👉 **Don't hesitate to stay tuned for more software development insights!**

DEBIOTECH