

# PMD Code Analysis Locally for Salesforce Development

This guide explains how to set up PMD for Apex code analysis on your local machine using Visual Studio Code. PMD helps us enforce code quality and standards across our Salesforce projects.

*PMD is already configured for our repositories (CORE and ICARE) with a shared `ruleset.xml`, located in the root directory.*

## 1. Install PMD Locally

PMD is a Java-based tool, so you'll need Java installed on your machine.

### Prerequisites

- **Java 8 or higher** must be installed. If you don't have Java installed:
  - Download and install from [Oracle](#).
  - After installation, verify by running:

```
java -version
```

### Install PMD

1. Follow installation steps <https://pmd.github.io/>

## 2. Install PMD Extension in VSCode

To make PMD easier to use directly from your editor, we recommend using the "**PMD extension for VSCode**".

### Steps

1. Open VSCode.
2. Go to the **Extensions** view (Ctrl+Shift+X).
3. Search for **PMD for Apex** by Chuck Jonas or any other preferred PMD extension (<https://marketplace.visualstudio.com/items?itemName=chuckjonas.apex-pmd>).
4. Install the extension.

### Configure the Extension

Once installed, check that the extension is already analyzing the code, using by default the `ruleset.xml`, otherwise, configure the extension to use the `ruleset.xml` already provided in the project root.

1. Open your VSCode **Settings** (Ctrl+, or Cmd+,).
2. Search for **PMD**.
3. Set the following:
  - **Path to PMD:** Provide the full path to the `pmd` executable. Example:
    - On Mac/Linux: `/Users/youruser/tools/pmd-bin-7.0.0/bin/pmd`
    - On Windows: `C:\tools\pmd\bin\pmd.bat`
  - **Rulesets:** Use the relative path from the root of your repo:

```
./ruleset.xml
```
  - **Language:** Set to `apex`.

Some extensions may allow configuring these settings via `.vscode/settings.json` in your repo:

```
{
  "pmd.binPath": "/absolute/path/to/pmd",
  "pmd.rulesets": [".ruleset.xml"],
  "pmd.language": "apex"
}
```

## 3. Running PMD in VSCode

Once configured, you can run PMD in the following ways:

- Automatically via the extension (on file save)
- Via the Command Palette:

- Open Command Palette (Ctrl+Shift+P)
- Type and select: PMD: Run

You can also run PMD manually via the command line if needed:

```
pmd -d ./force-app -R ./ruleset.xml -f text
```

*Replace ./force-app with the path to your source folder if different.*

---

## 4. Suppressing warnings

Check [https://pmd.github.io/pmd/pmd\\_userdocs\\_suppressing\\_warnings.html](https://pmd.github.io/pmd/pmd_userdocs_suppressing_warnings.html)

Important: For any warning suppressed, there must be a comment in the code explaining the reason why it's suppressed, which must be a solid reason.

i.e

// NOTE: This PMD Vulnerability is not fixed due to the high risk of occasionating a regression in this critical part of the logic, given the accumulated technical debt and complexity of it

## 5. Request removing rule from Copado Quality Gates

The PMD rules enforced in Copado can be customized, where the priority can be changed and even the rule could be removed if there is a reason to do so.

Reach out to [GASQUETON, JULIEN](#) / [Abhishek Jyoti](#)

## Additional Resources

- PMD official docs: <https://pmd.github.io>
- Apex ruleset reference: [https://pmd.github.io/latest/pmd\\_rules\\_apex.html](https://pmd.github.io/latest/pmd_rules_apex.html)
- PMD GitHub repo: <https://github.com/pmd/pmd>