In order to better understand the capabilities and benefits of static analysis tools, learners will select a well-established (popular) static analysis tool with online documentation and report on two important features of the tool.

For each feature, the discussion will be graded on whether the purpose, instructions to use, and importance of the tool is:

* present,
* clear,
* reasonable, and
* well-written (clear language, proper grammar, correct spelling).

Select a well-established static analysis tool which has online documentation of its capabilities. Select a feature of the tool. Briefly explain the feature's purpose, how to use it, and why it is important to have this analysis capability.

<https://www.perforce.com/>

Speed - It takes time for developers to do manual code reviews. Automated tools are much faster. Static code checking addresses problems early on. And it pinpoints exactly where the error is in the code. So, you’ll be able to fix those errors faster. Plus, coding errors found earlier are less costly to fix.

Use the same static analysis tool selected for the previous prompt. Select another feature of the tool. Briefly explain this feature's purpose, how to use it, and why it is important to have this analysis capability.

<https://www.perforce.com/>

Accuracy - Manual code reviews are prone to human error. Automated tools are not. They scan every line of code to identify potential problems. This helps you ensure the highest-quality code is in place — before testing begins. After all, when you’re complying a coding standard, quality is critical