Coding standards and practices followed in IT industries

Consistent formatting and indentation: Most companies have a standard format for code, such as a particular indentation style, naming conventions, and commenting style. This makes the code easier to read and maintain

Version control: Companies use version control systems like Git to manage changes to code and collaborate on projects. This allows multiple developers to work on the same codebase and track changes made over time

Testing and debugging: Companies typically follow a rigorous process of testing and debugging to ensure that their code works as intended. This includes unit testing, integration testing, and acceptance testing

Code reviews: Many companies have a process for peer review, where other developers review code before it is merged into the main codebase. This helps catch errors and ensures that the code is consistent with company standards

Security: Companies follow best practices for security to protect their code and data. This includes techniques like secure coding practices, vulnerability assessments, and penetration testing

Documentation: Companies typically document their code, including its purpose, how it works, and any dependencies. This helps other developers understand the code and make changes as needed

Design patterns: Companies often use design patterns to solve common software design problems. Using design patterns can make code more reusable, maintainable, and easier to understand.

Code modularity: Modularity refers to breaking up code into smaller, reusable components that can be used across different projects. This helps reduce code duplication and makes it easier to maintain and update code

Error handling: IT companies typically follow best practices for error handling to ensure that errors are handled gracefully and that users receive clear error messages.

Performance optimization: Companies often optimize code for performance, especially in cases where code needs to handle a large amount of data or requests. This might involve techniques like caching, lazy loading, or optimizing database queries

Code documentation tools: IT companies often use code documentation tools like Javadoc or Doxygen to automatically generate documentation from code comments. This can help developers understand code and reduce the time required for manual documentation

Code profiling: Code profiling tools are used to analyze the performance of code and identify potential bottlenecks or areas for optimization

Code reuse: Many companies use open-source libraries and frameworks to reuse existing code and avoid reinventing the wheel. This can save time and reduce the risk of errors

Accessibility: IT companies often follow accessibility guidelines to ensure that their code is accessible to users with disabilities. This might include techniques like using alternative text for images, providing keyboard navigation options, and ensuring that content is easily readable

Hariharan C <hariharan.c@indiumsoft.com>