**Q1. What are Git Repository Guidelines in Agile Development?**

**Answer:** Git is a popular version control system that is widely used in Agile software development. Git provides a powerful set of features for managing source code, collaborating with other developers, and keeping track of changes over time. Here are some Git repository guidelines that can help teams using Agile development:

1. **Use Branches:** Branches are a powerful feature in Git that allow developers to work on different features or bug fixes independently of each other. Teams should use branches to isolate changes, minimize conflicts, and facilitate code reviews.
2. **Adopt a Workflow:** Teams should adopt a Git workflow that suits their development process. Popular workflows include Gitflow, Trunk-based Development, and Feature Branching. A workflow provides a set of rules and conventions that can help teams collaborate effectively.
3. **Use Pull Requests:** Pull Requests are a mechanism in Git that allow developers to submit changes for review and approval. Pull Requests can help teams catch errors, improve code quality, and facilitate collaboration between team members.
4. **Use Tags:** Tags are a way to mark specific versions or milestones in the codebase. Teams can use tags to denote releases, hotfixes, or other important events. Tags can help teams keep track of changes over time and communicate effectively.
5. **Use Commit Messages:** Commit messages are a way to document changes in the codebase. Teams should use clear and descriptive commit messages that provide context and explain the rationale behind each change. This can help team members understand the codebase and review changes more effectively.
6. **Use Git Hooks:** Git Hooks are scripts that run automatically when certain events occur in Git. Teams can use Git Hooks to automate tasks like running tests, formatting code, or enforcing coding standards. This can help teams save time and improve code quality.
7. **Use a Git Hosting Service:** Git hosting services like GitHub, GitLab, and Bitbucket provide a powerful set of features for managing repositories, collaborating with other developers, and automating workflows. Teams should use a Git hosting service that suits their needs and provides the features they require.

By following these Git repository guidelines, teams can effectively manage their codebase, collaborate more effectively, and improve the quality of their software.