# Explore best practices for source control

Completed100 XP

* 2 minutes
* **Make small changes**. In other words, commit early and commit often. Be careful not to commit any unfinished work that could break the build.
* **Do not commit personal files**. It could include application settings or SSH keys. Often personal files are committed accidentally but cause problems later when other team members work on the same code.
* **Update often and right before pushing to avoid merge conflicts**.
* **Verify your code change before pushing it to a repository**; ensure it compiles and tests are passing.
* **Pay close attention to commit messages, as it will tell you why a change was made**. Consider committing messages as a mini form of documentation for the change.
* **Link code changes to work items**. It will concretely link what was created to why it was created—or modified by providing traceability across requirements and code changes.
* **No matter your background or preferences, be a team player and follow agreed conventions and workflows**. Consistency is essential and helps ensure quality, making it easier for team members to pick up where you left off, review your code, debug, and so on.