Case Study 4 Exercises: Automation & Security

- Add a CI workflow in `.github/workflows/ci.yml` that runs `npm test` (or `pytest`) on every push and PR. Push it and verify the Actions tab shows a passing run.

- Enable CodeQL code scanning and Dependabot alerts. Introduce a known-vulnerable dependency in your manifest, then fix it after Dependabot opens a PR.

- Create a custom release workflow in `.github/workflows/release.yml` that uses an auto-tag action to bump versions when `main` is pushed. Test by pushing a commit with a version bump in `package.json` or `pyproject.toml`.

- View your repository's **Dependency graph**, locate any alerts, and upgrade or remove a flagged package.

Case Study 5 Exercises: Open Source & Advanced Git

- Fork an external open-source repo, clone your fork, and add `upstream` as a remote. Fetch from `upstream` and ensure you can merge new commits into your `main`.

- Identify a small fix commit in `upstream/main`, cherry-pick it into a new branch in your fork, resolve any conflicts, then push and open a PR upstream.

- On your fork's `main`, create an annotated tag following semantic versioning (e.g., `v0.1.1-patch`), push it, and draft a GitHub release with release notes.

- On a feature branch with multiple commits, perform an interactive rebase to squash them into a single commit, then force-push the branch to your fork.