



GLORY ACADEMY (G.A)  
P.O.BOX 2280 KIGALI-RWANDA  
Email: [info@gloryacademy.ac.rw](mailto:info@gloryacademy.ac.rw)  
Website: [www.gloryacademy.ac.rw](http://www.gloryacademy.ac.rw)



## Practical Task for Learning Outcome 1: Setup Repository

Students must set up a Git repository for a web development project simulating SEZERANO ALPHA's Innovate company Ltd forms (choose one: login form, student registration, entertainment, courses registration, or book registration). Develop the form using HTML for structure, CSS for styling (responsive design with Flexbox), and JavaScript for basic validation (e.g., email format check). Properly introduce Git, install/configure it, init locally, create GitHub remote repo, and set remote URL.

Instructions:

- Create HTML form with at least 5 inputs, styled CSS, interactive JS validation showing errors dynamically.
- Terminal commands: `git --version`, `git config --global user.name "Your Name"`, `git config --global user.email "your@email.com"`, `git init`, create/configure `.gitignore` for `node_modules/temp` files.
- GitHub: New repo "setup-repo-[yourname]", `git remote add origin [repo-url]`, initial commit/push to main.
- Time: 4 hours. Individual work.
- Invite [nyirinkindimarcel@gmail.com](mailto:nyirinkindimarcel@gmail.com) as collaborator. Submit repo URL and terminal screenshots of setup.

## Practical Task for Learning Outcome 2: Manipulate Files

Extend your LO1 form project by adding new features (e.g., dropdown selections, dynamic fields via JS). Demonstrate file manipulation: create/modify/delete files (add `new.html`, update CSS/JS), check status (`git status`), stage changes selectively (`git add specific-file`), commit with descriptive messages (`git commit -m "Add dynamic fields"`), create/manage branches (new "feature-form-enhance" branch, switch, list, rename if needed).

Instructions:

- Enhance form: Add JS to append fields on button click, improve validation with real-time feedback.
- Git workflow: git status (view untracked/modified), git add . or specific, git reset to unstage, git rm file.html (remove and stage), git log --oneline, git branch feature-form-enhance, git checkout feature-form-enhance.
- Commit best practices: Meaningful messages, amend if needed (git commit --amend).
- Push branch to GitHub from LO1 repo (reuse or new "manipulate-files-[yourname]").
- Time: 4 hours. Individual.
- Invite [nyirinkindimarcel@gmail.com](mailto:nyirinkindimarcel@gmail.com), submit branch comparison screenshot.

### **Practical Task for Learning Outcome 3: Ship Codes**

Complete your LO2 enhanced form by adding final features (e.g., form submission simulation storing data in localStorage). Ship codes: fetch from remote (git fetch), pull latest (git pull origin main), push feature branch (git push origin feature-form-enhance), create Pull Request on GitHub for merging to main, handle merge/rebase conflicts if simulated. Demo synchronization between local/remote.

Instructions:

- JS: Form submit → localStorage save/display submitted data as list.
- Git commands: git fetch origin, git pull origin main, git push -u origin feature-final, GitHub PR creation with description/review.
- Merge options: git merge feature-final or rebase (git rebase main), resolve conflicts.
- Use repo from LO2: "ship-codes-[yourname]" or continue previous.
- Time: 4 hours. Individual work.
- Submit PR link (approved/merged), terminal logs of fetch/pull/push/merge. Invite [nyirinkindimarcel@gmail.com](mailto:nyirinkindimarcel@gmail.com) as collaborator for review