**Lab Exercise 3 - Git Cherry-Pick**

**Objective:**

Learn how to use git cherry-pick to apply specific commits from one branch to another.

**Steps:**

1. **Initialize a Repository:**

git init cherry-pick-lab

cd cherry-pick-lab

1. **Create and Commit a File on the Main Branch:**

echo "Initial content" > file.txt

git add file.txt

git commit -m "Initial commit"

1. **Create a New Branch:**

git checkout -b feature-branch

1. **Add and Commit Changes in the Feature Branch:**
   * Add a new line to the file:

echo "Feature branch update 1" >> file.txt

git commit -am "Add update 1 in feature branch"

* + Add another change:

echo "Feature branch update 2" >> file.txt

git commit -am "Add update 2 in feature branch"

1. **Switch Back to the Master Branch:**

git checkout master

1. **Cherry Pick a Commit:**
   * Get the commit hash of the second commit from the feature branch:

git log feature-branch --oneline

(Copy the hash for the commit **"Add update 2 in feature branch"**.)

* + Cherry pick the commit:

git cherry-pick <commit-hash>

1. **Verify the Cherry-Picked Changes:**
   * Check the file contents:

cat file.txt

* + View the commit history:

git log --oneline