Student Handout: Lab 1 – Git + Cloud9 Collaboration

Objective

- Set up GitHub and Cloud9

- Use Git to manage code

- Simulate team workflow with branches

- Add basic CI check using Git hook

PART A: Setup (15 mins)

1. **Create GitHub Repo**

   - Name: `devops-lab`

   - Add README.md

2. **Launch Cloud9**

   - Name: `DevOpsLabEnv`

   - Region: `us-west-2`

3. **Configure Git**

```
git config --global user.name "Your Name"
git config --global user.email "you@example.com"
```

4. **Clone Repo**

```
git clone https://github.com/your-username/devops-lab.git
cd devops-lab
```

5. **Authenticate**

   - Use GitHub **PAT** instead of password

PART B: Create Feature Branch (30 mins)

```
git checkout -b feature-hello-script
echo -e "#!/bin/bash\necho Hello from DevOps Lab!" > hello.sh
chmod +x hello.sh
git add hello.sh
git commit -m "Add hello script"
git push origin feature-hello-script
```

- Go to GitHub → Create a Pull Request (PR) → Merge

```
git checkout main
git pull origin main
```

PART C: Simulate CI with Git Hook (30 mins)

```
mkdir -p .git/hooks
cat <<'EOF' > .git/hooks/pre-commit
#!/bin/bash
for file in $(git diff --cached --name-only); do
```

```
  if [[ "$file" == *.sh && ! -x "$file" ]]; then
    echo "$file is a shell script but not executable"
    exit 1
  fi
done
EOF
chmod +x .git/hooks/pre-commit
```

- Test:

```
echo "echo This should fail" > test.sh
git add test.sh
git commit -m "Bad commit"
```

Key Takeaways

- Work with Git in cloud

- Use branches + PR for team workflow

- Simulate CI using Git hooks