

### **Verify if master is in clean state**

```
git checkout master
```

```
git status
```

---

### **Create branch GitWork and add hello.xml**

```
git checkout -b GitWork
```

```
echo "<message>Hello from GitWork</message>" > hello.xml
```

---

### **Update the content of hello.xml**

```
echo "<note>Updated from GitWork branch</note>" >> hello.xml
```

```
git status
```

Expected:

modified: hello.xml

---

### **Commit the changes in GitWork**

```
git add hello.xml
```

```
git commit -m "Added and updated hello.xml in GitWork"
```

---

### **Switch to master**

```
git checkout master
```

---

### **Add hello.xml to master with different content**

```
echo "<master>Hello from master branch</master>" > hello.xml
```

---

### **Commit the changes to master**

```
git add hello.xml
```

```
git commit -m "Added hello.xml in master with different content"
```

---

### View logs using graph format

```
git log --oneline --graph --decorate --all
```

Example output:

```
* 9a3e1d5 (HEAD -> master) Added hello.xml in master with different content
| * f842a23 (GitWork) Added and updated hello.xml in GitWork
|/
* 8b1426c Initial commit
```

---

### Check differences with Git diff tool

```
git diff master GitWork
```

---

### Visual diff with P4Merge (if installed)

```
git config --global diff.tool p4merge
```

```
git config --global merge.tool p4merge
```

```
git difftool master GitWork
```

---

### Merge GitWork into master

```
git merge GitWork
```

! You will see a **conflict in hello.xml**, because both branches modified the same file.

---

### Git markup in conflicted file

Open hello.xml, see:

```
<<<<<<< HEAD
<master>Hello from master branch</master>
=====
<message>Hello from GitWork</message>
<note>Updated from GitWork branch</note>
>>>>>>> GitWork
```

---

### **Use 3-way merge tool (e.g., P4Merge)**

```
git mergetool
```

---

### **Commit the resolved merge**

```
git add hello.xml
```

```
git commit -m "Resolved merge conflict in hello.xml"
```

---

```
git status
```

Then:

```
echo "*.orig" >> .gitignore
```

---

### **Commit .gitignore**

```
git add .gitignore
```

```
git commit -m "Add .orig to .gitignore"
```

---

### **List all branches**

```
git branch
```

---

### **Delete merged branch**

```
git branch -d GitWork
```

---

### **View final log graph**

```
git log --oneline --graph --decorate
```

```
* 6c1e60e (HEAD -> master) Resolved merge conflict in hello.xml
```

```
|\
```

```
| * f842a23 Added and updated hello.xml in GitWork
```

```
* | 9a3e1d5 Added hello.xml in master with different content
```

```
|/
```

```
* 8b1426c Initial commit
```

