## **Git Conflict Resolution Assignment**

Name: Emmadi Vivek Reddy  
**Hall** Ticket Number: 2406CDO111  
Date: 29/03/2025

### **Objective**

The objective of this assignment is to create and resolve a Git conflict using two branches in a GitHub repository. The steps include creating a conflict scenario, resolving the conflict, and documenting the process with screenshots.

## **Step 1: Create a New Feature Branch and Modify a File**

Now, create a new branch to modify a file

$ git checkout -b feature-branch

Open ec2.tf and change the tag name.

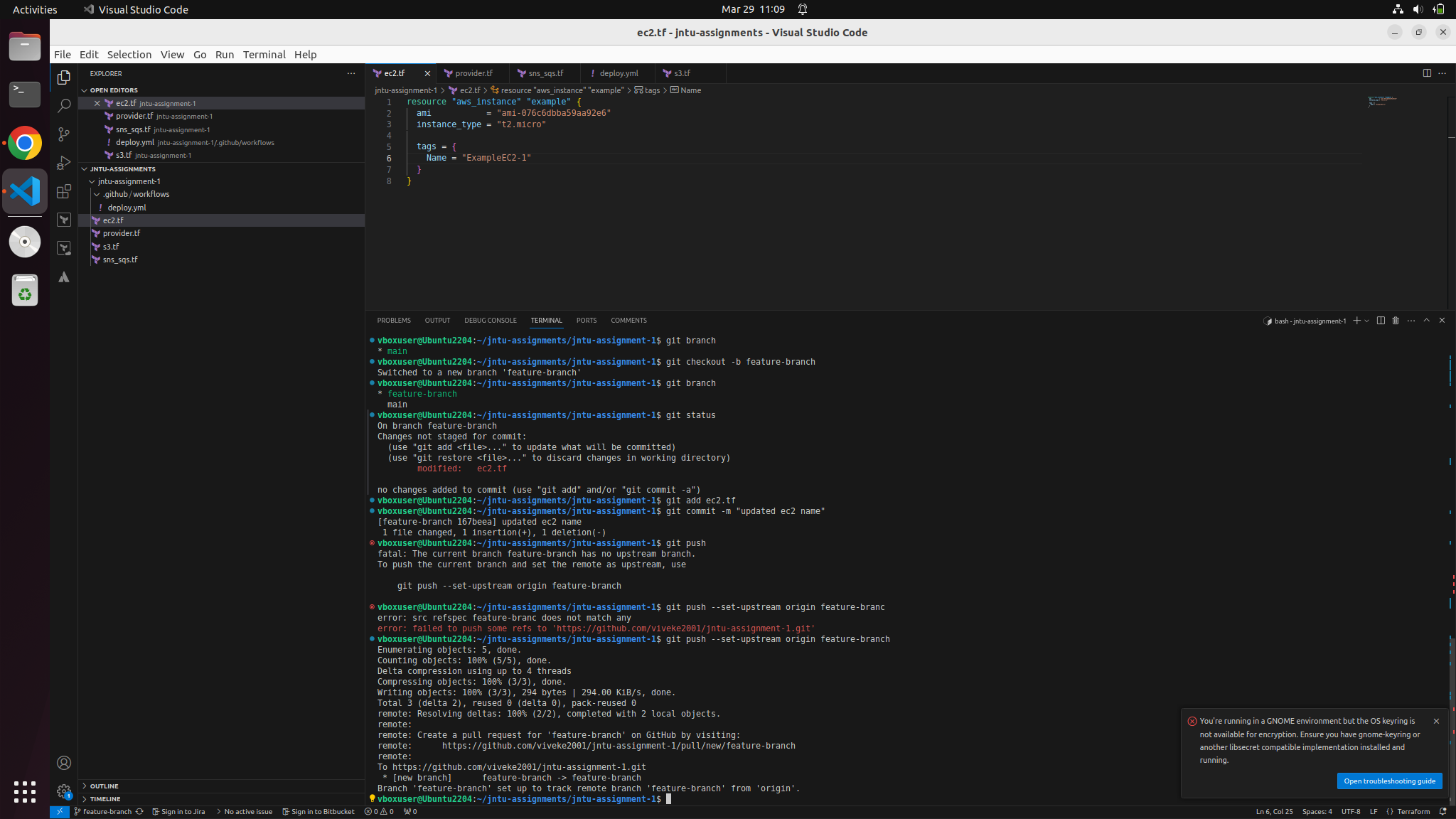
Original in main branch

Name = "ExampleEC2"

Modify in feature-branch

Name = "ExampleEC2-1”

Save the file and commit the change.



## **Step 2: Modify the Same File in** main

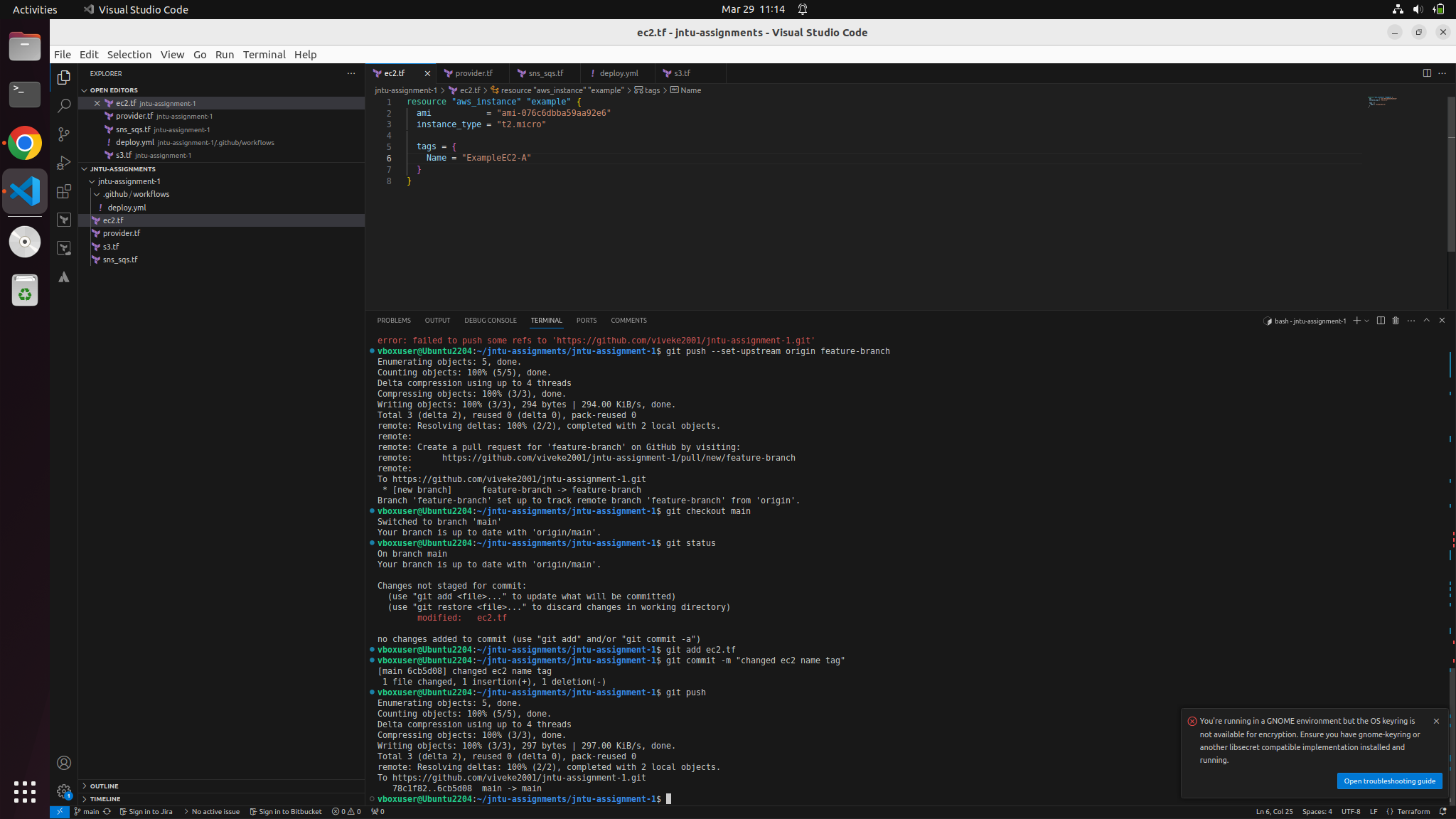
Switch back to main and modify the same line in the same file with a different value.

$ git checkout main

#### Modify in main branch

Name = "ExampleEC2-A"

**Save the file and commit the change.**



## **Step 3: Merge and Resolve Conflict**

Now, try to merge the feature branch into main.

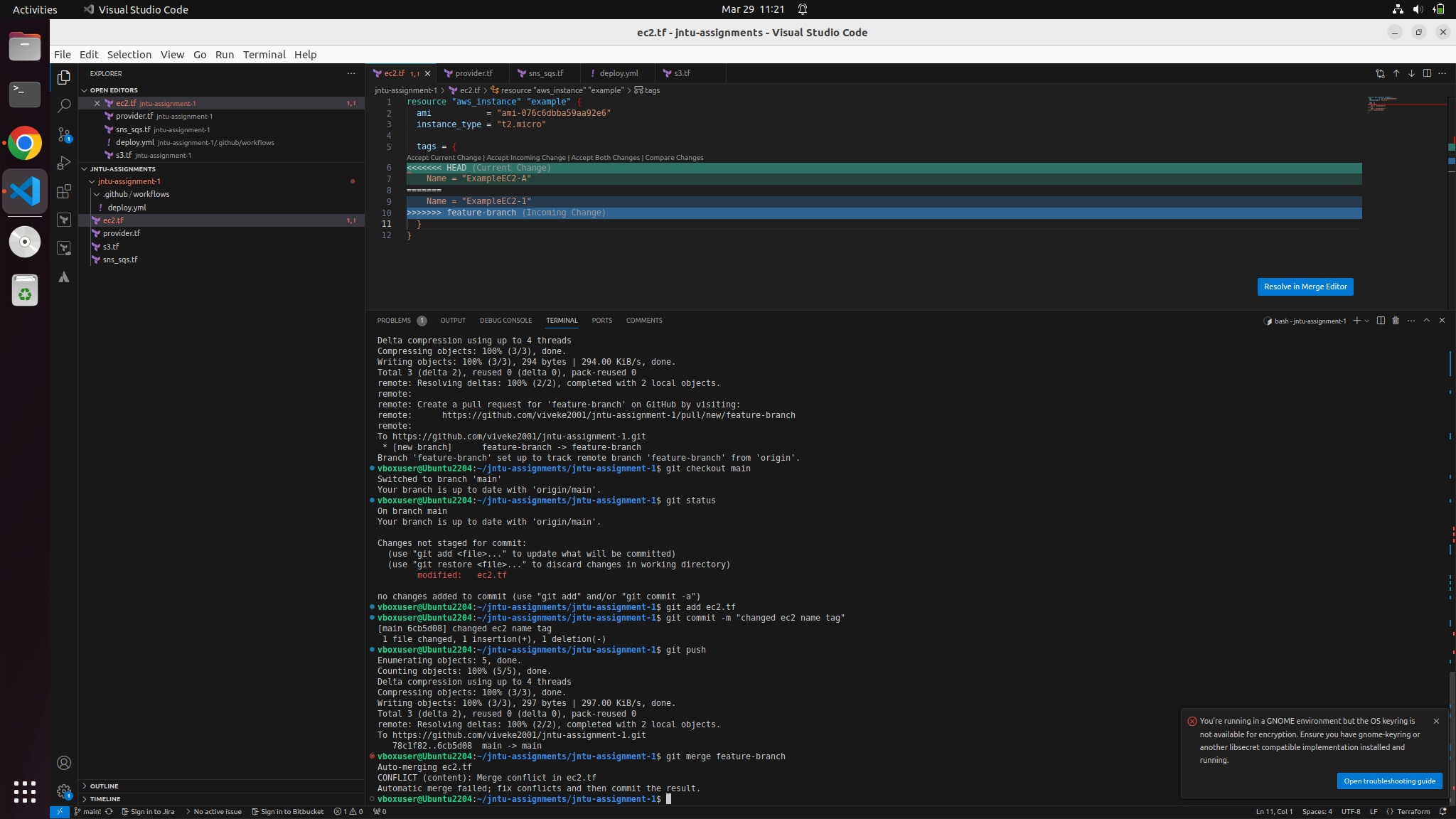
$ git merge feature-branch

Since both branches modified the same line, Git will detect a merge conflict.

Open ec2.tf, and you’ll see conflict markers (<<<<<<<, =======, >>>>>>>).

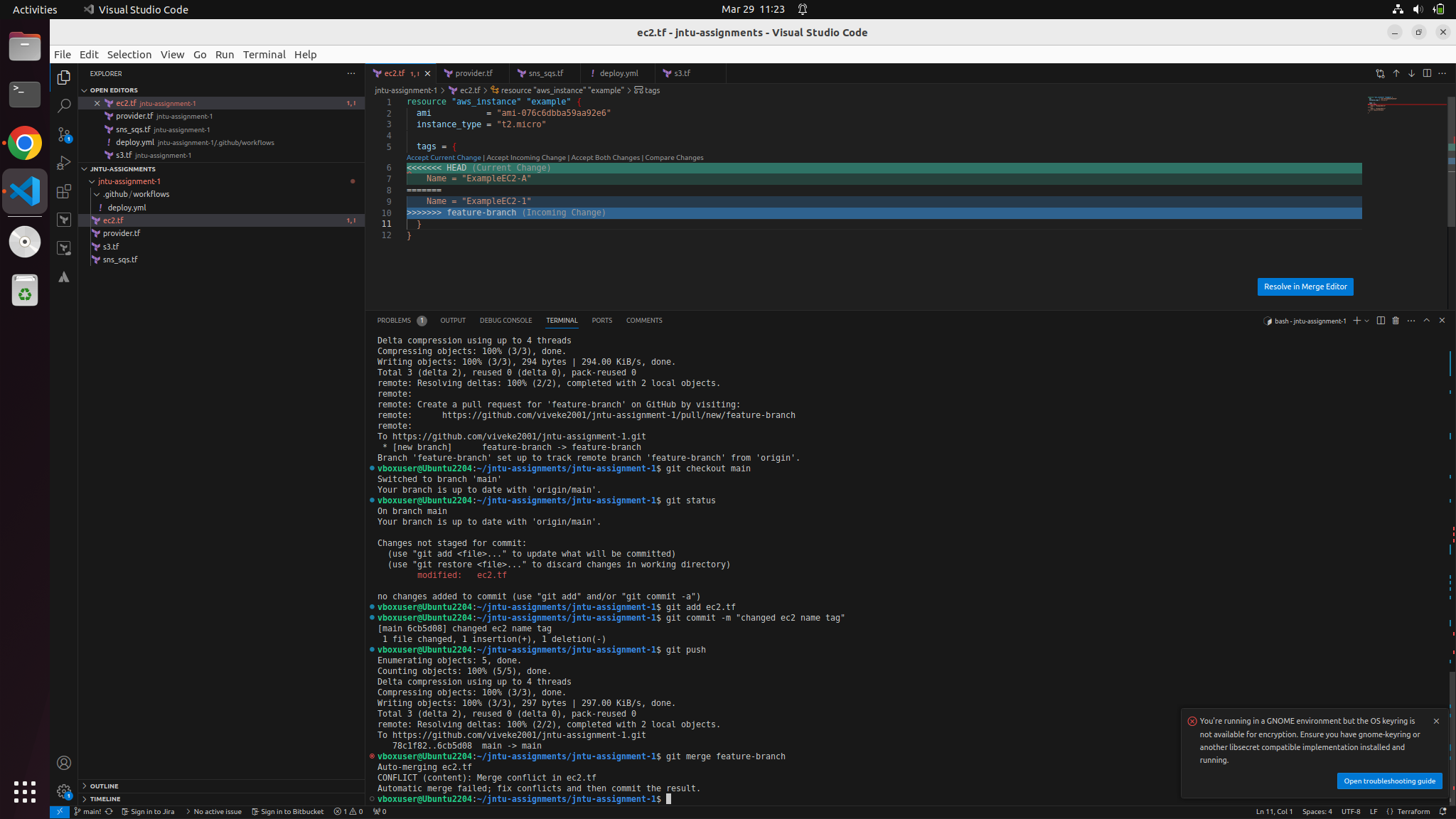
Manually resolve the conflict by choosing the correct value:

In the below picture there is conflict shown in the terminal after the merge command and even in the ec2.tf file.

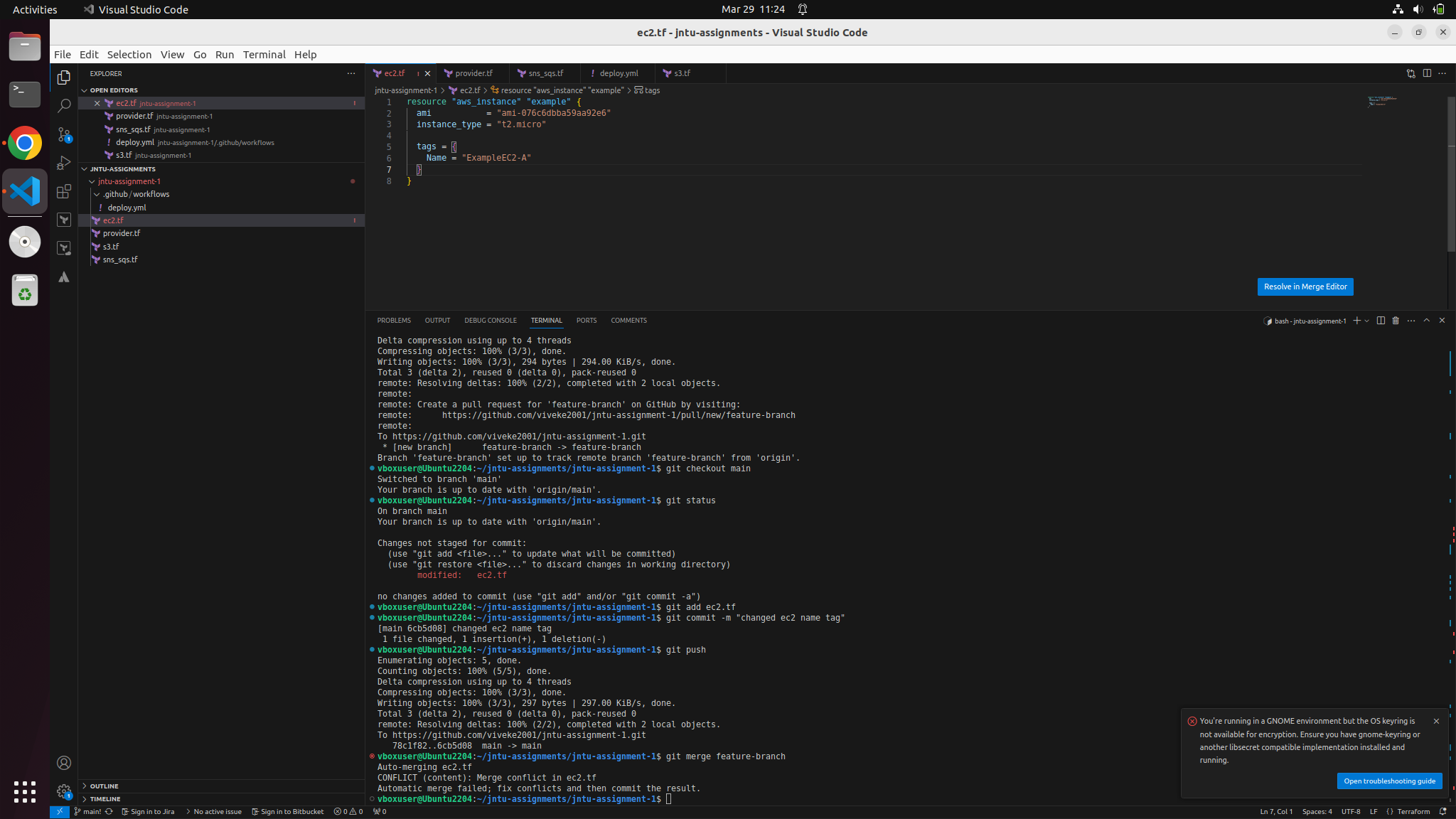


In this below picture there is a conflict and there are multiple options to go with, those are

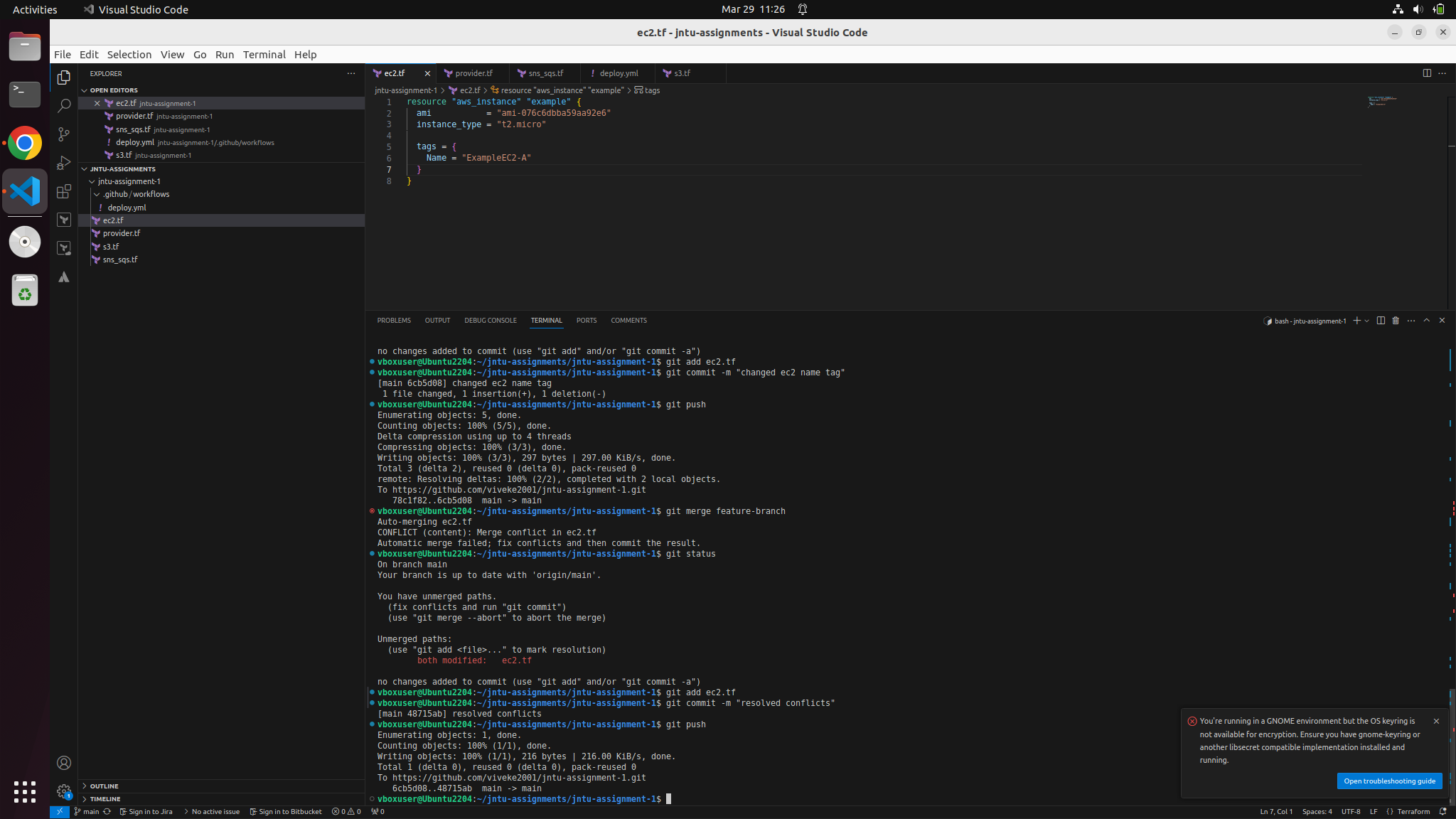
Accept Current Changes, Accept Incoming Changes, Accept Both Changes and i am going with the current changes and the below picture is before accepting the current changes



And this is what it looks like when the current changes are accepted and we can see the name accepting the current changes inside the tags attribute



Now commit and push the changes as shown in the below picture



## **Step 4: Verify Merge History**

Run the command to verify the merge history

$ git log --oneline --graph –all

