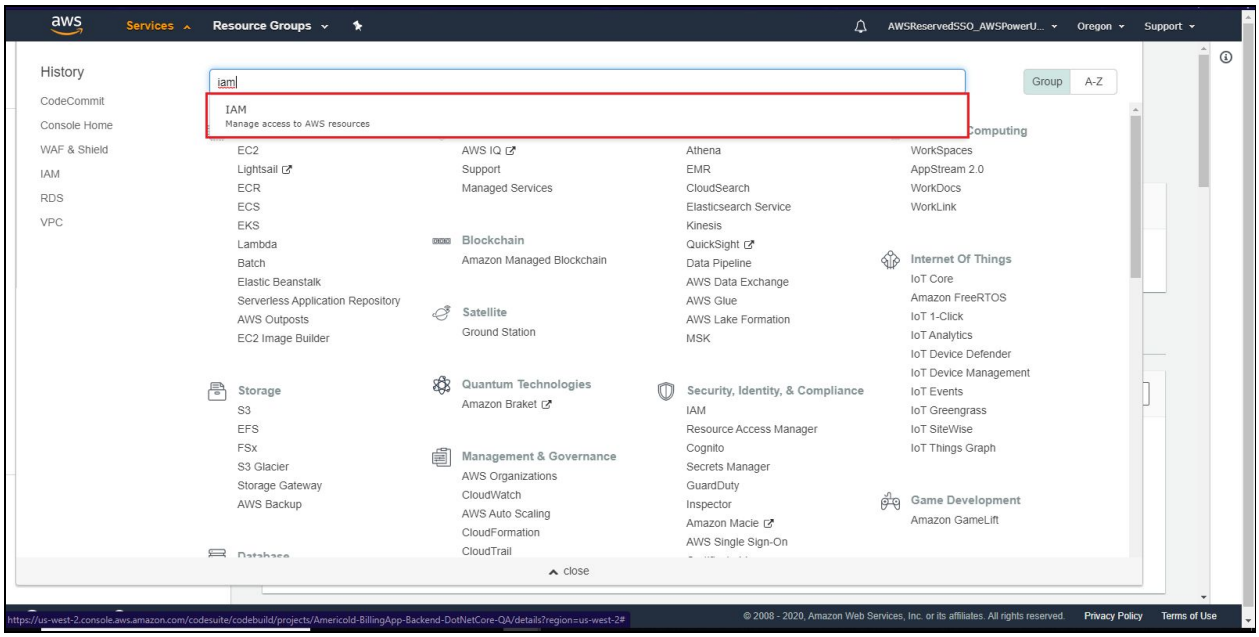
AWS Code Commit

Agenda:

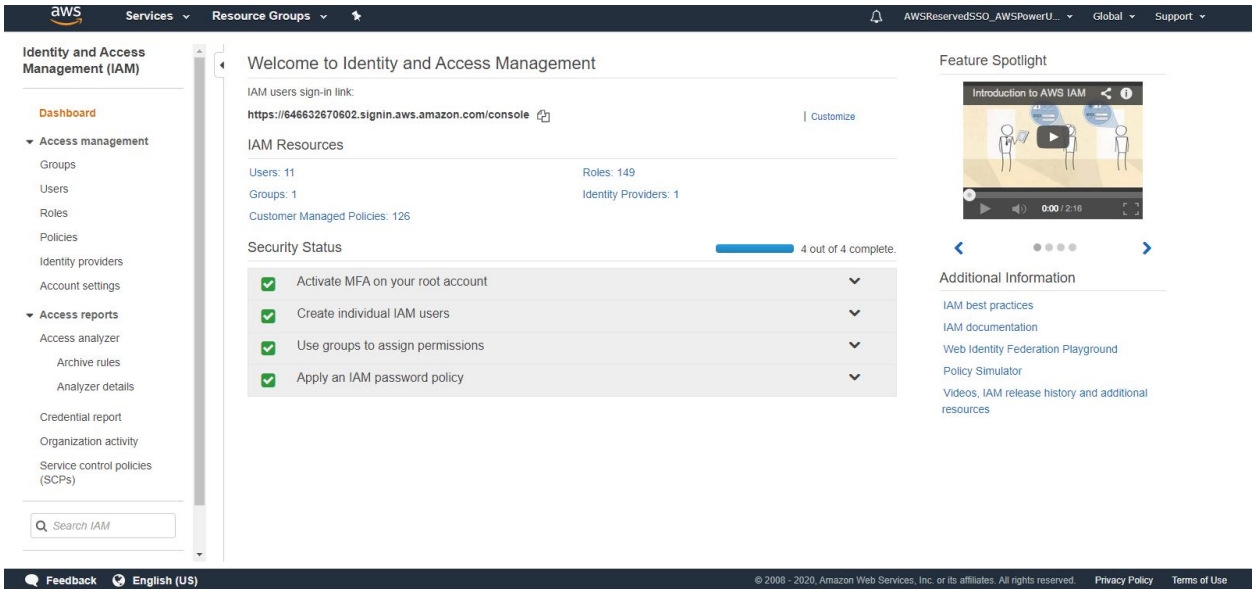
AWS code commit is a fully-managed source control service that hosts secure git-based repositories. It makes it easy for teams to collaborate on code in a secure and highly scalable ecosystem. code commit eliminates the need to operate your own source control system or worry about scaling its infrastructure. You can use code commit to securely store anything from source code to binaries, and it works seamlessly with your existing Git tools.

By the end of the tutorial, we learn how to how to provide access to the developers for generating git credentials, access the repositories, create a repository, and how to create branches, manage the pull requests and code merging from branch to branch.

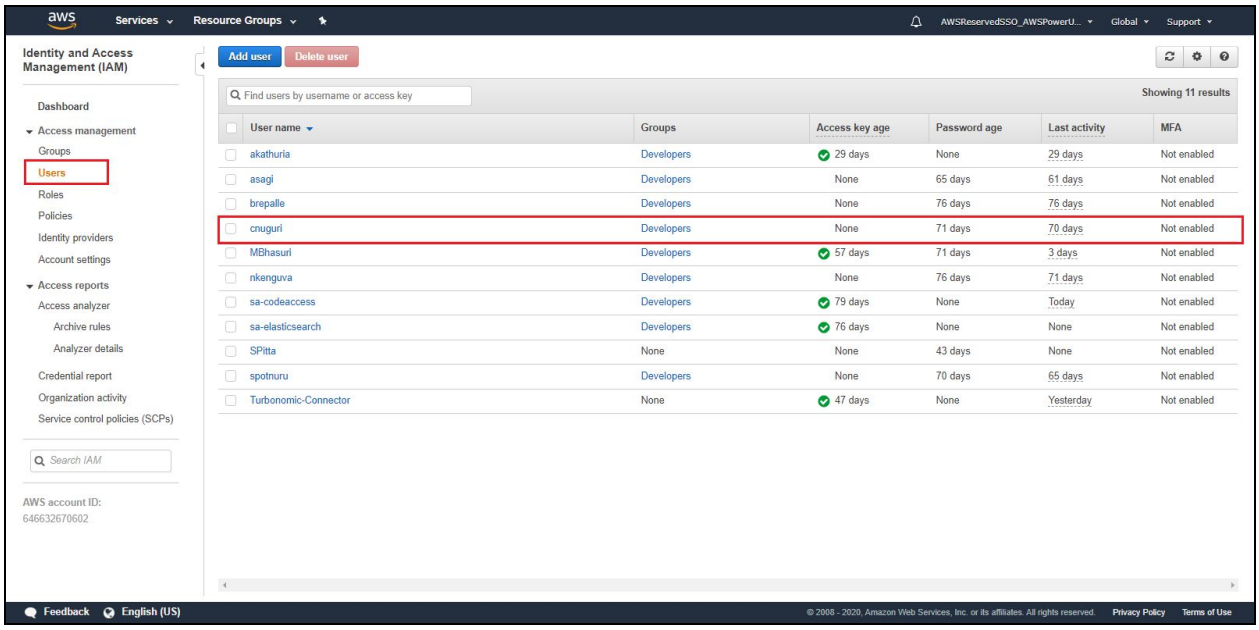
First, we need to login to the AWS console and search for the **IAM** service.



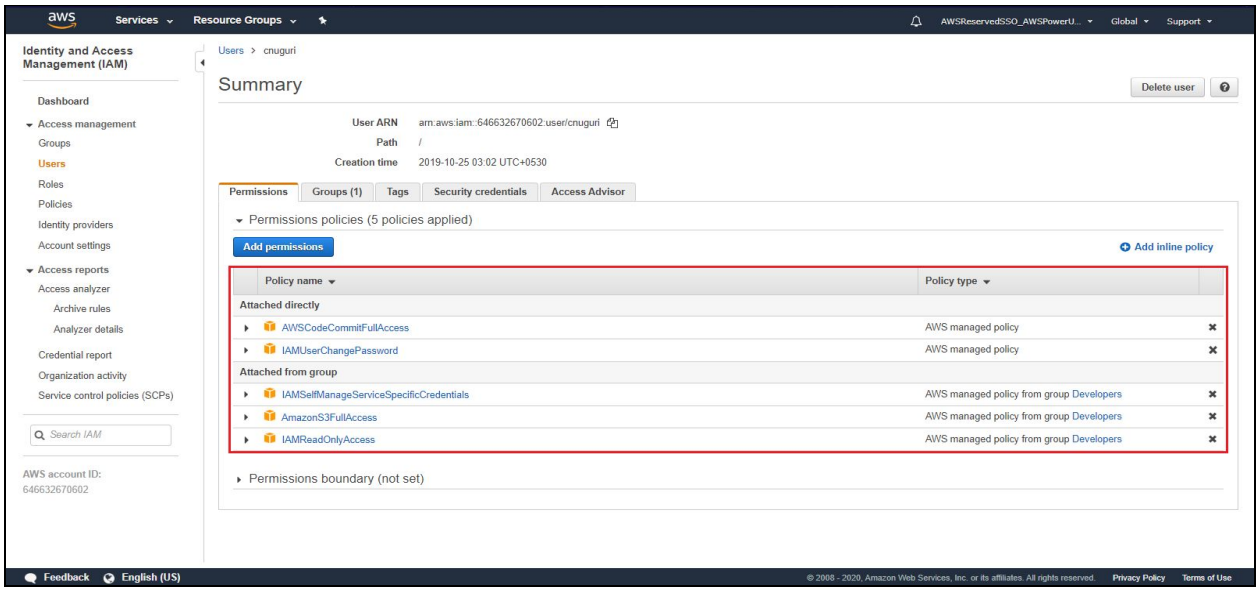
Following is the dashboard of the IAM service.



Goto users section in the dashboard and select an IAM user



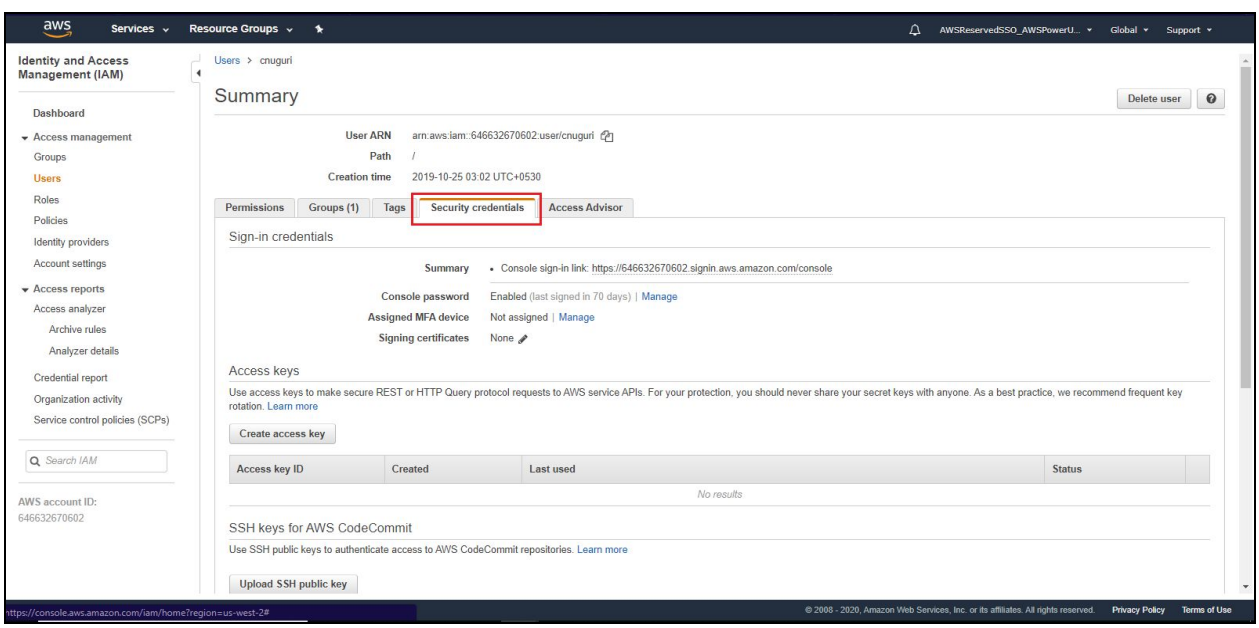
Now click on add permissions to provide necessary permissions to the IAM user to generate security credentials for git to access git repositories from development environment.



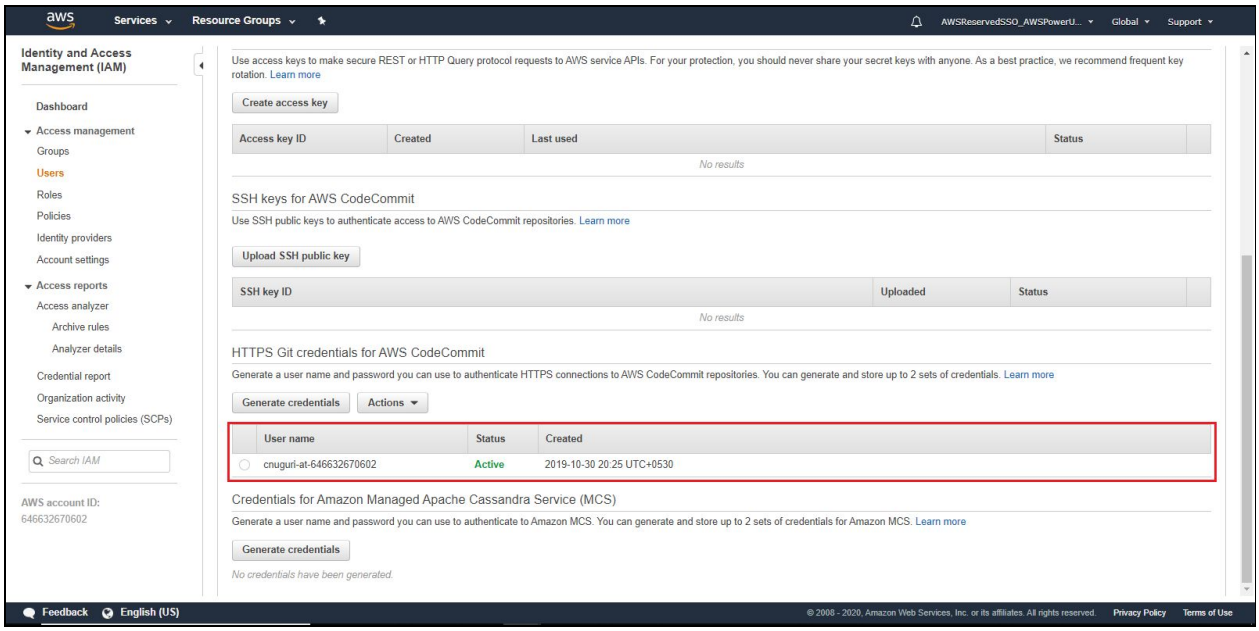
Here we have given permissions for

- **IAMReadOnlyAccess** will provide visibility for IAM dashboard.
- **CodeCommitFull access** to access repositories and can able to create repos and branches for repositories.
- **SelfManageServiceSpecificCredentials** to generate security credentials to manage the repositories from development environment.

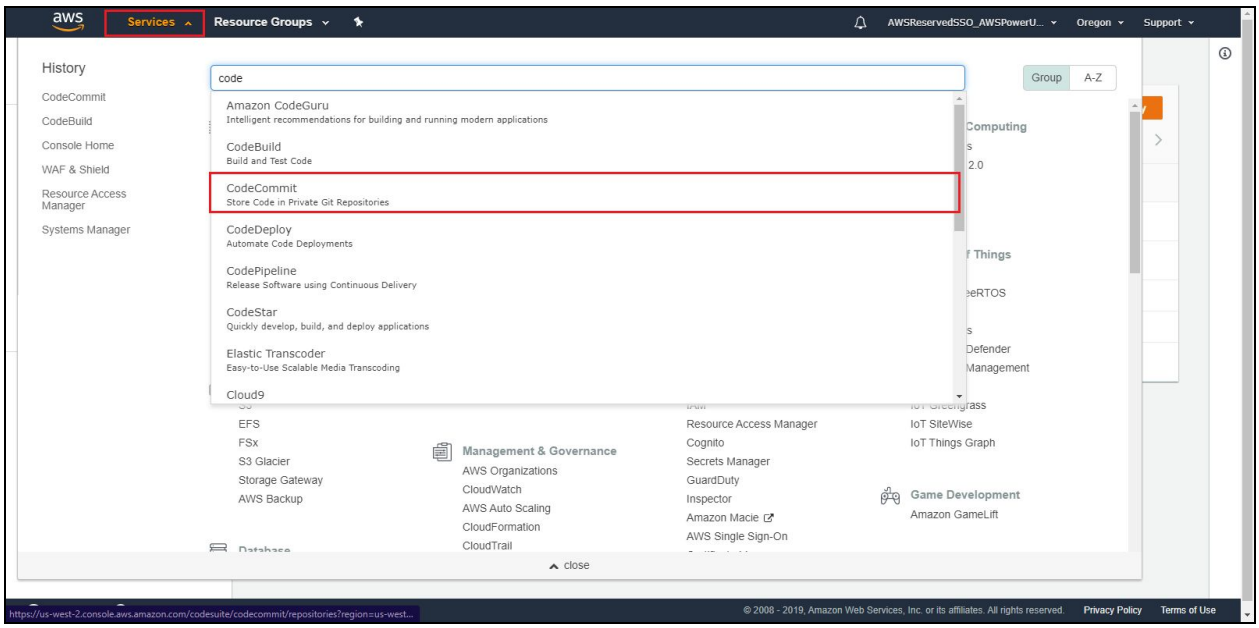
To generate security credentials, click on security credentials tab in IAM User dashboard.



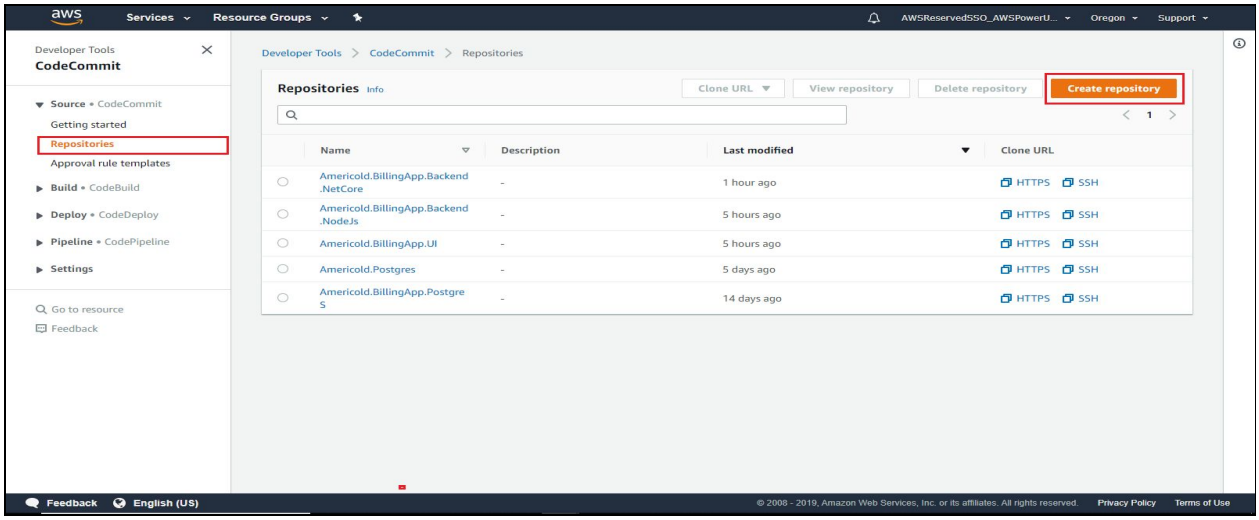
Click on generate credentials for HTTPS Git credentials for AWS CodeCommit and download the credentials. These credentials helps us to manage the repositories from development environment.



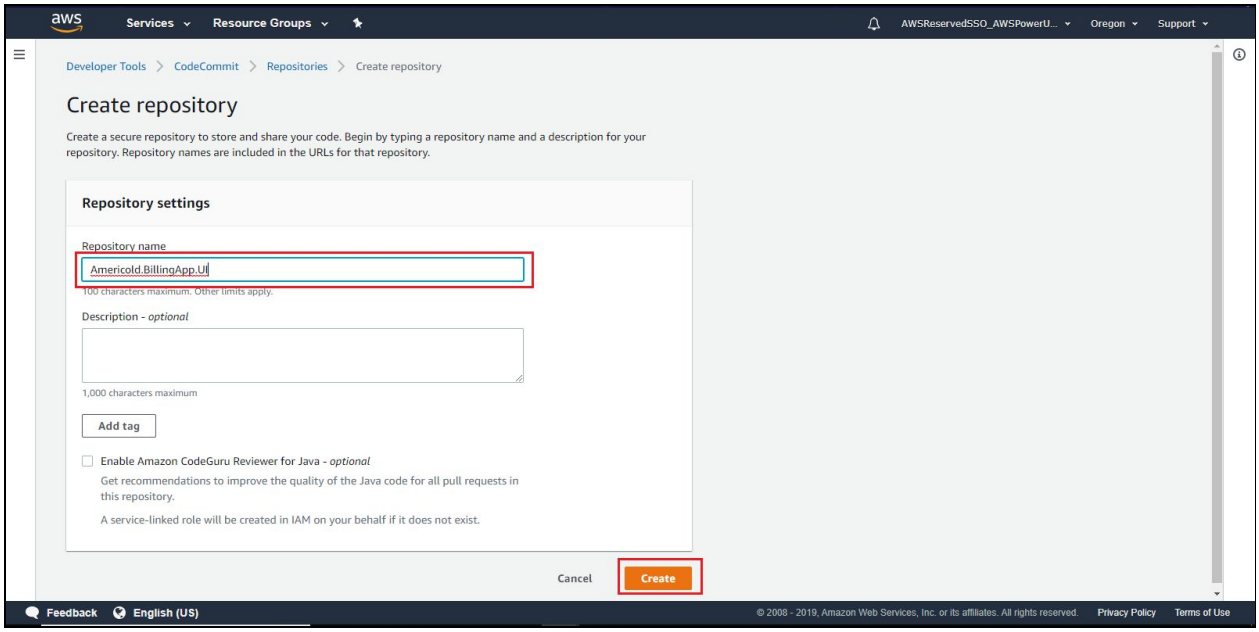
Now search for **CodeCommit** Service.



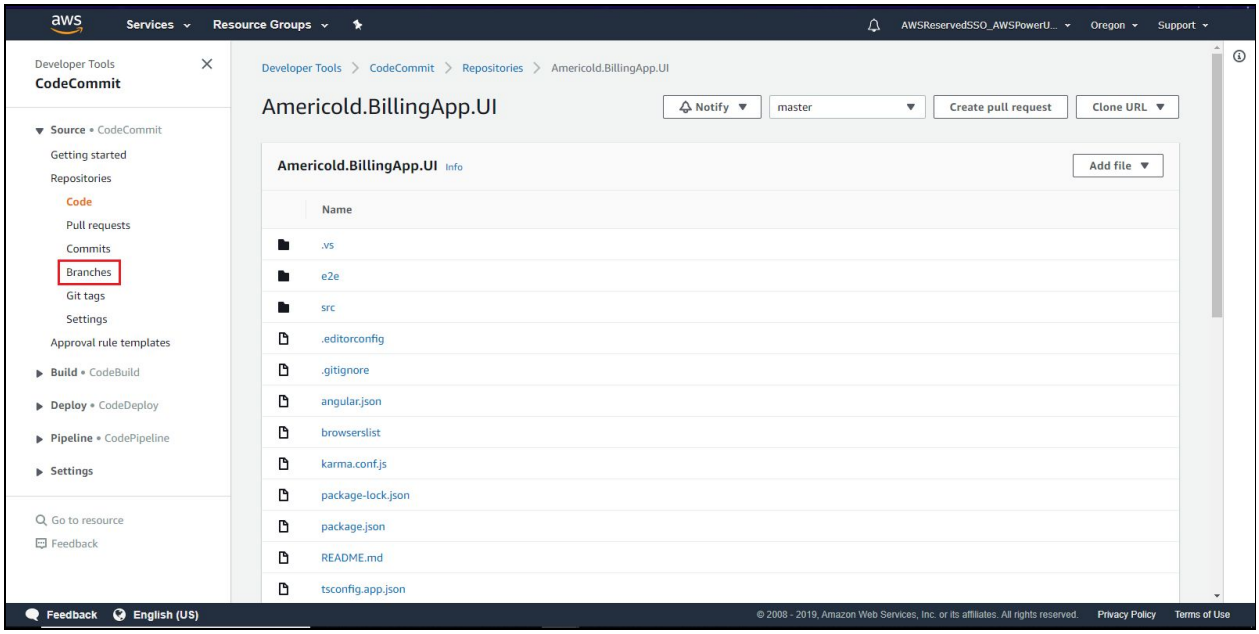
Now goto repositories and Click on **Create repository** to create a new repository**.**



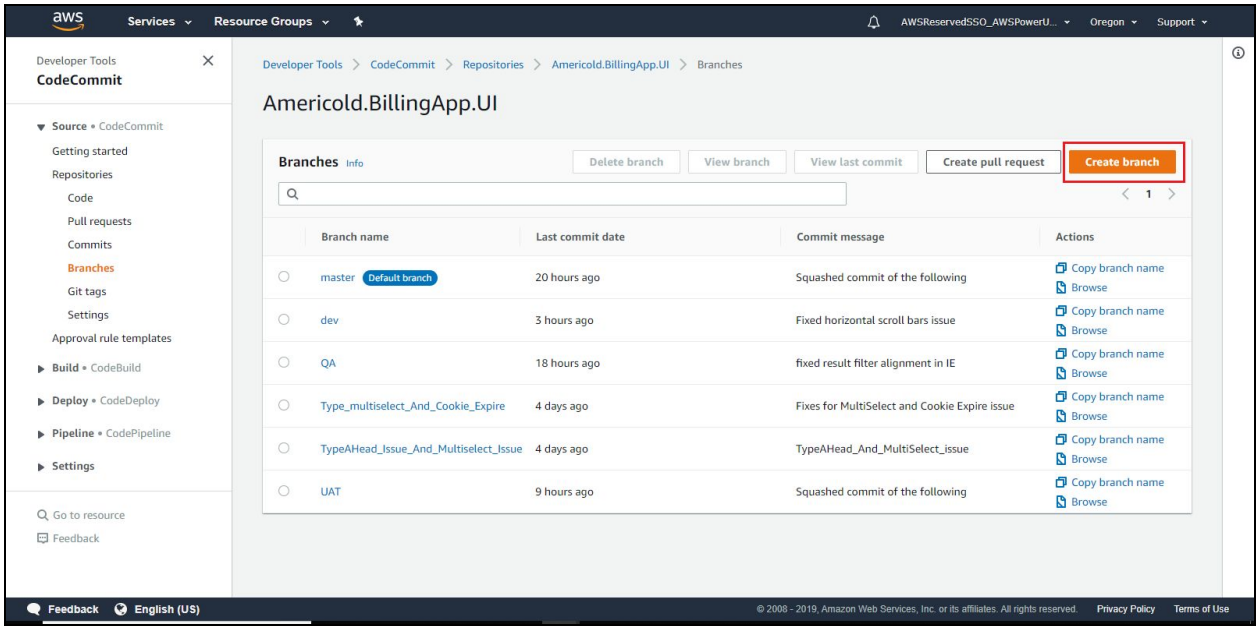
Enter the name of the repository as **Americold_BillingApp_UI** and Click on **Create.**



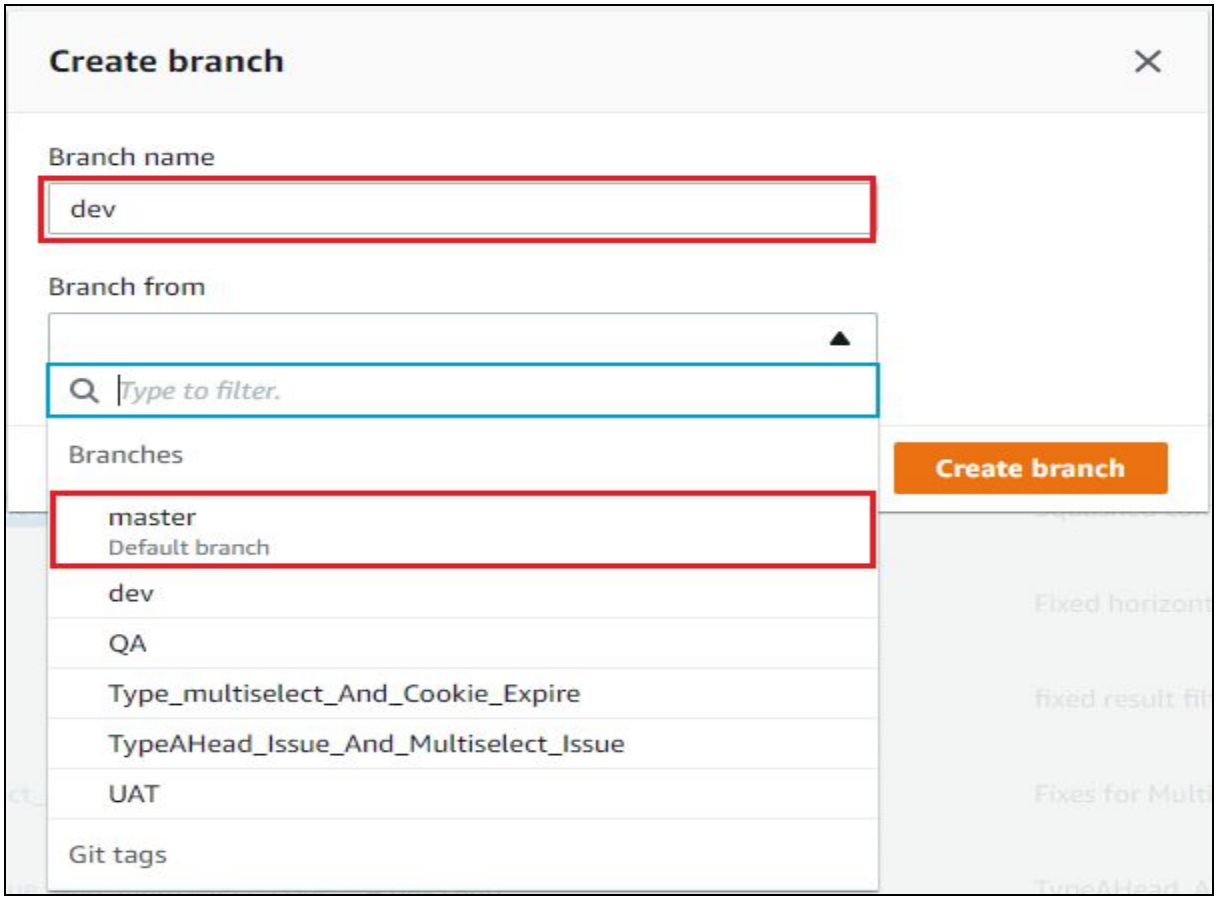
Once the repository got created, go to **branches** section which is shown in the left pane of the dashboard to manage branches for the repository.



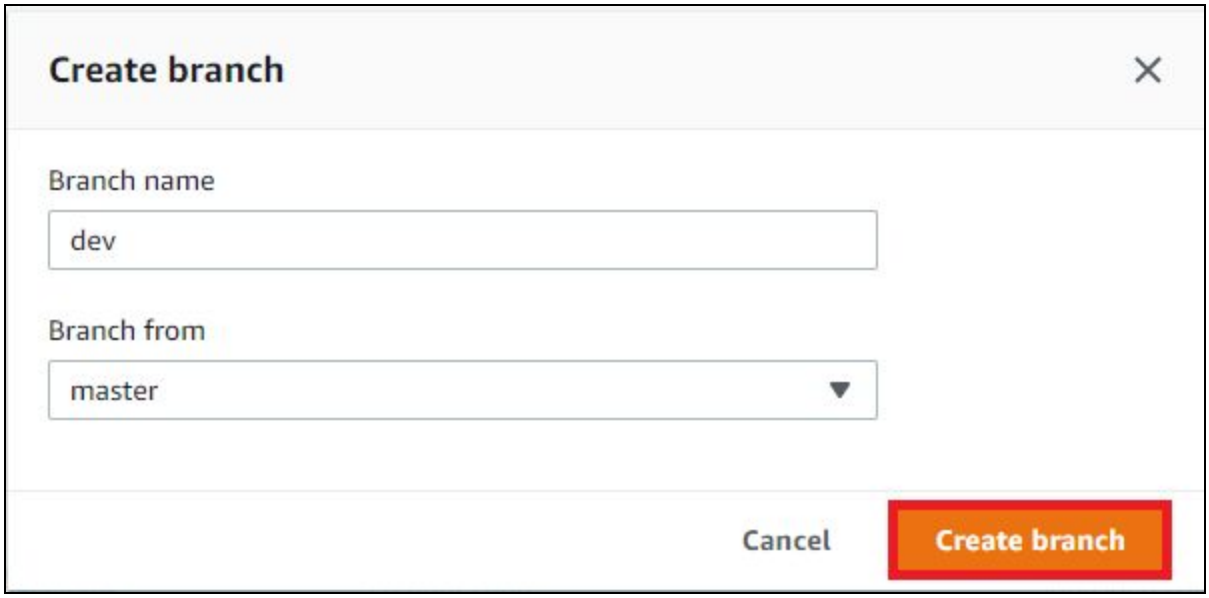
Now, Click on create branch to create a new branch in the repository.



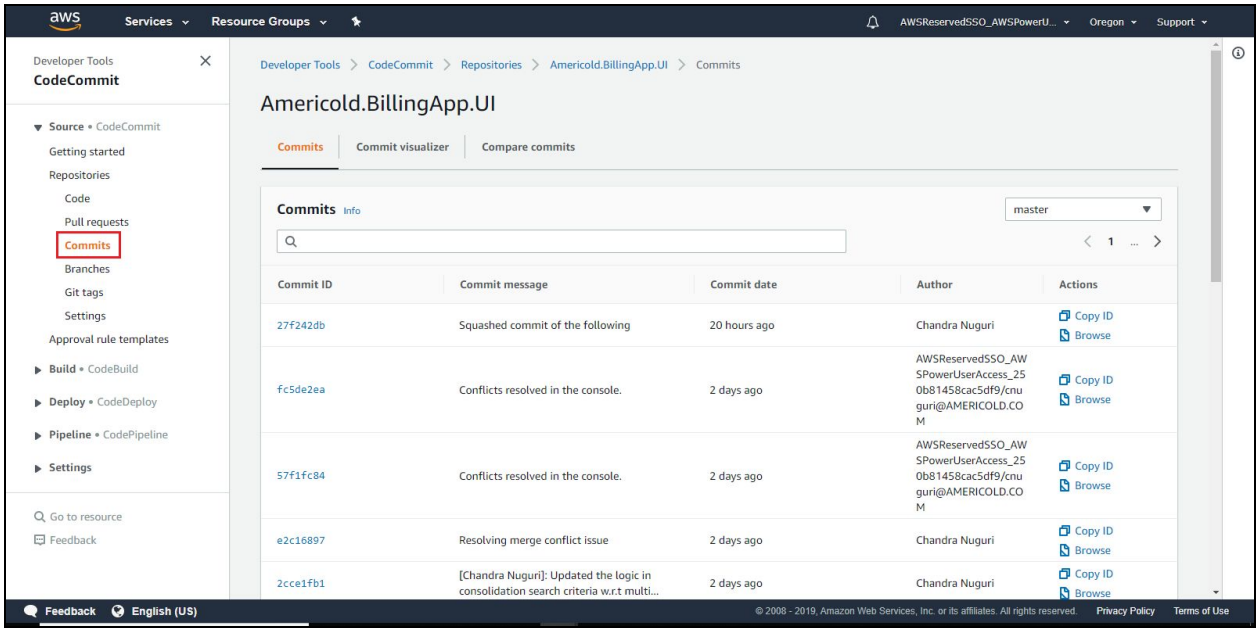
Enter the Branch name as **dev** and Select the **master branch** as a parent branch.



Next, Click on **Create branch**.

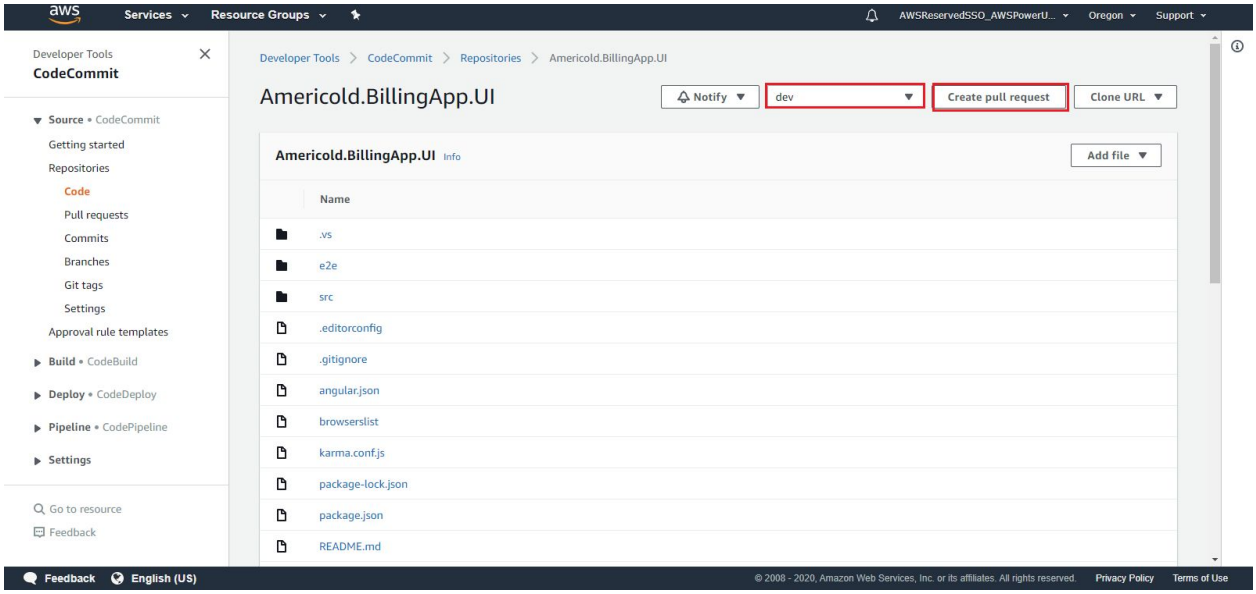


We can also see the Commit history of the repository in commits section.

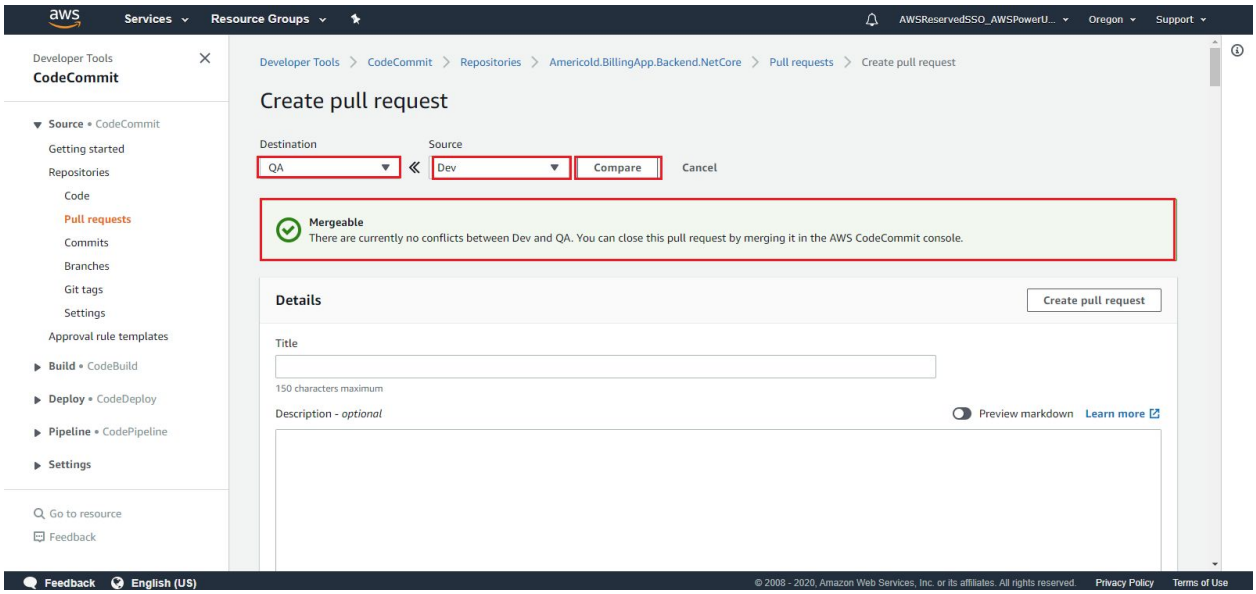


Merging code from one branch to another branch:

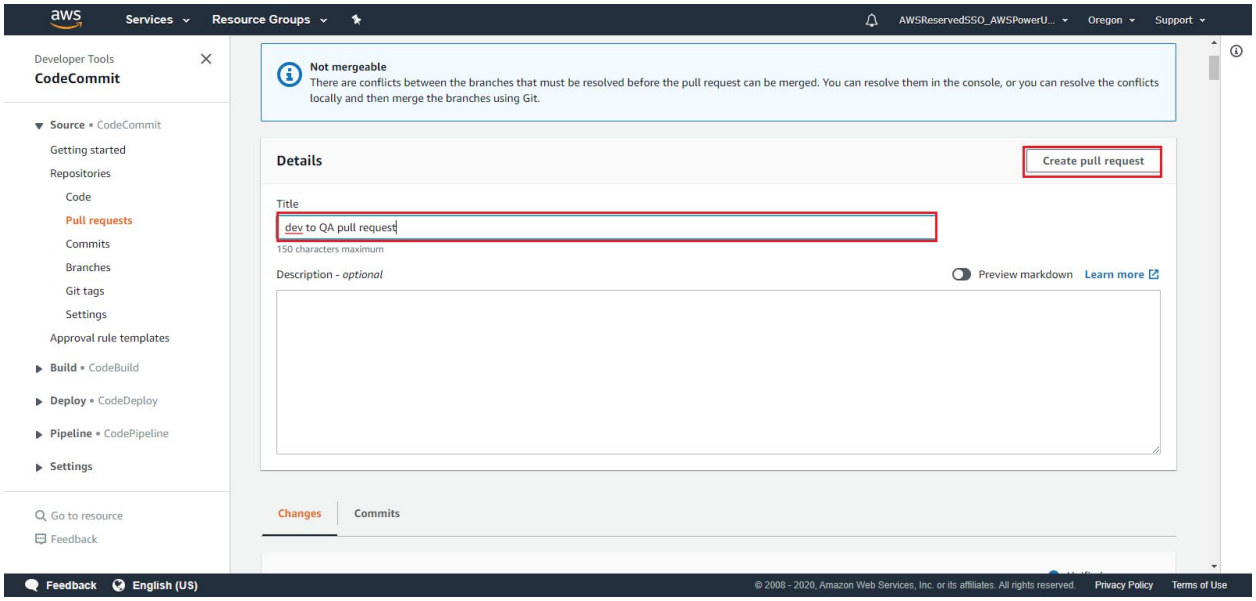
Currently we are in dev branch. To merge code from dev branch to QA we need to create a pull request. Now click on create pull request.



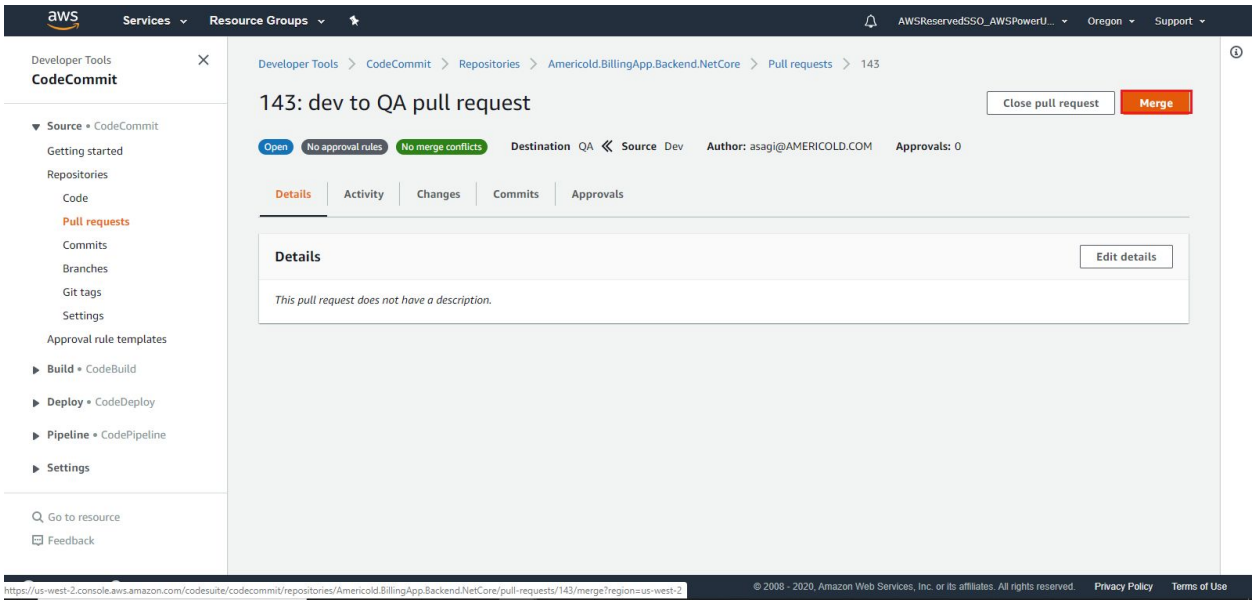
Select the source and destination branches and click on compare. If mergeable it will show the status as mergeable.



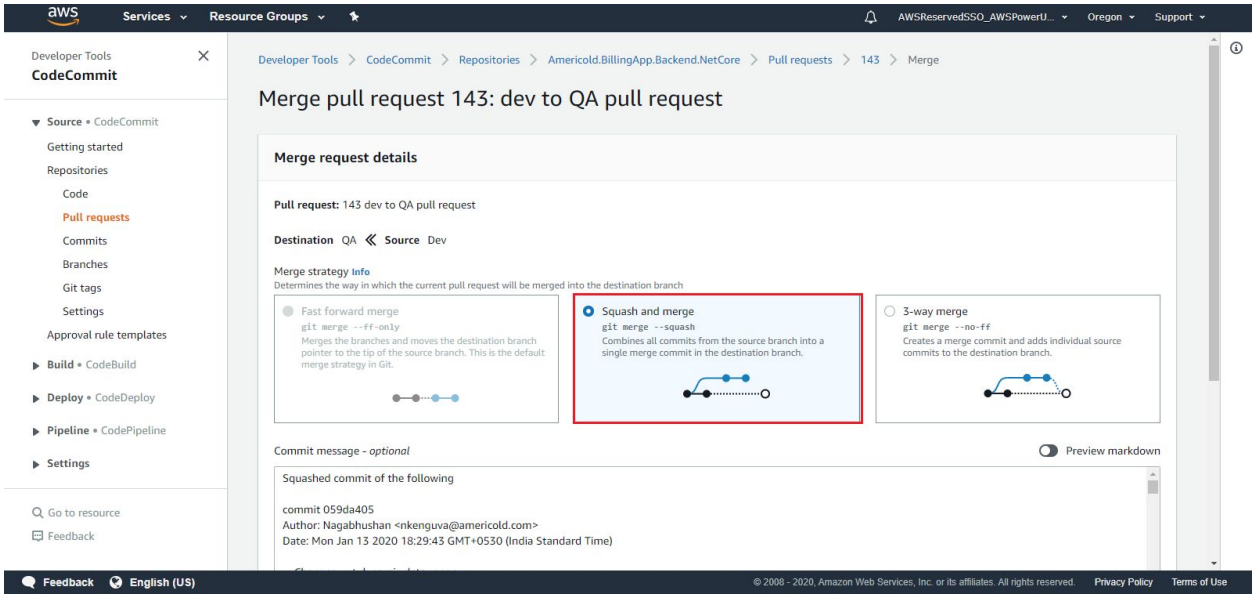
If it is mergeable give a title to the pull request and click on create pull request.



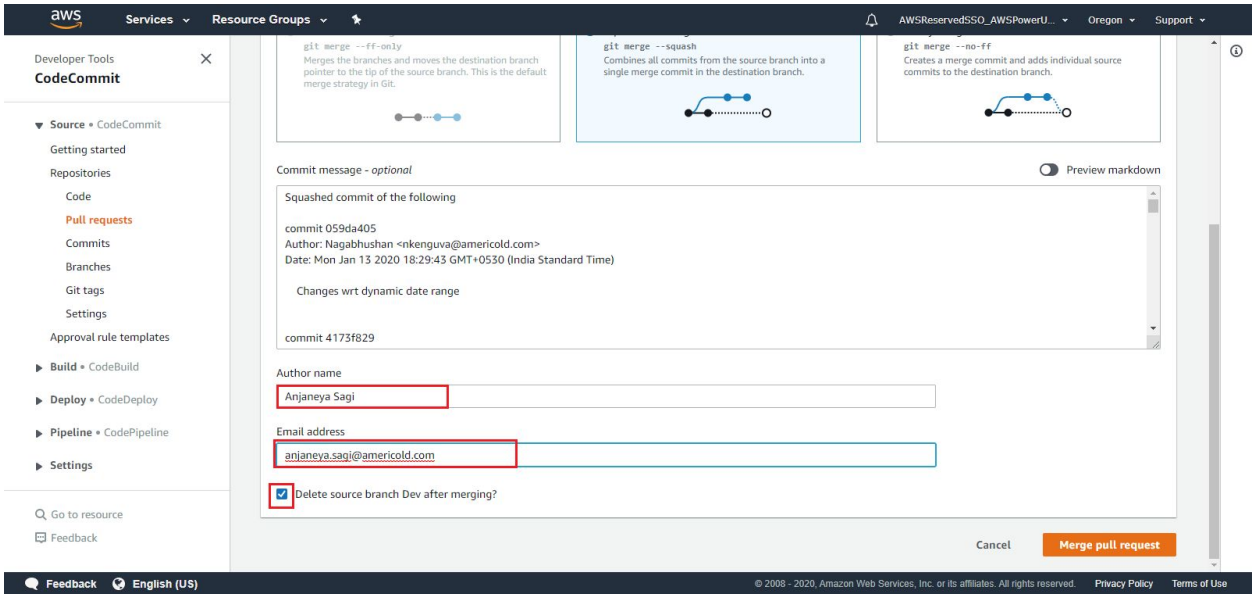
Now click on merge to merge the code from dev to QA



Once we click on merge in above step it redirects us to the following dashboard. Here we need to select a merge strategy for merging. Here we are selecting squash and merge.



Now give the name and email address of the author who is creating who is merging the code. By default delete source branch will be selected, we need to deselect the checkbox.



After deselecting the checkbox click on merge pull request, then the QA pipeline will automatically triggers and deployment to QA environment will be done.