Codestar integrate with IDEs- ECLIPSE, CLOUD9,VS CODE.

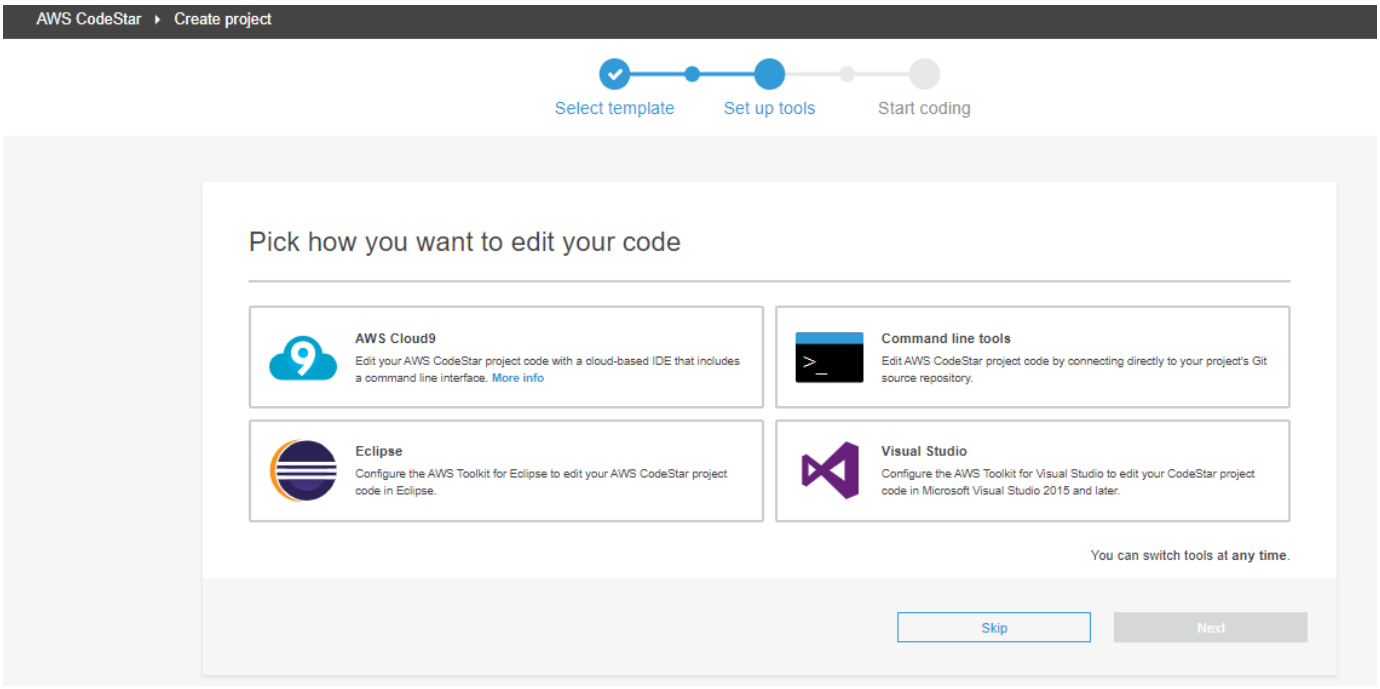
Use AWS Cloud9 with AWS CodeStar

You can use AWS Cloud9 to make code changes and develop software in an AWS CodeStar project. AWS Cloud9 is an online IDE, which you access through your web browser. The IDE offers a rich code editing experience with support for several programming languages and runtime debuggers, as well as a built-in terminal. In the background, an Amazon EC2 instance hosts an AWS Cloud9 development environment. This environment provides the AWS Cloud9 IDE and access to the AWS CodeStar project's code files.

You can use the AWS CodeStar console or AWS Cloud9 console to create AWS Cloud9 development environments for projects that store their code in CodeCommit. For AWS CodeStar projects that store their code in GitHub, you can only use the AWS Cloud9 console.

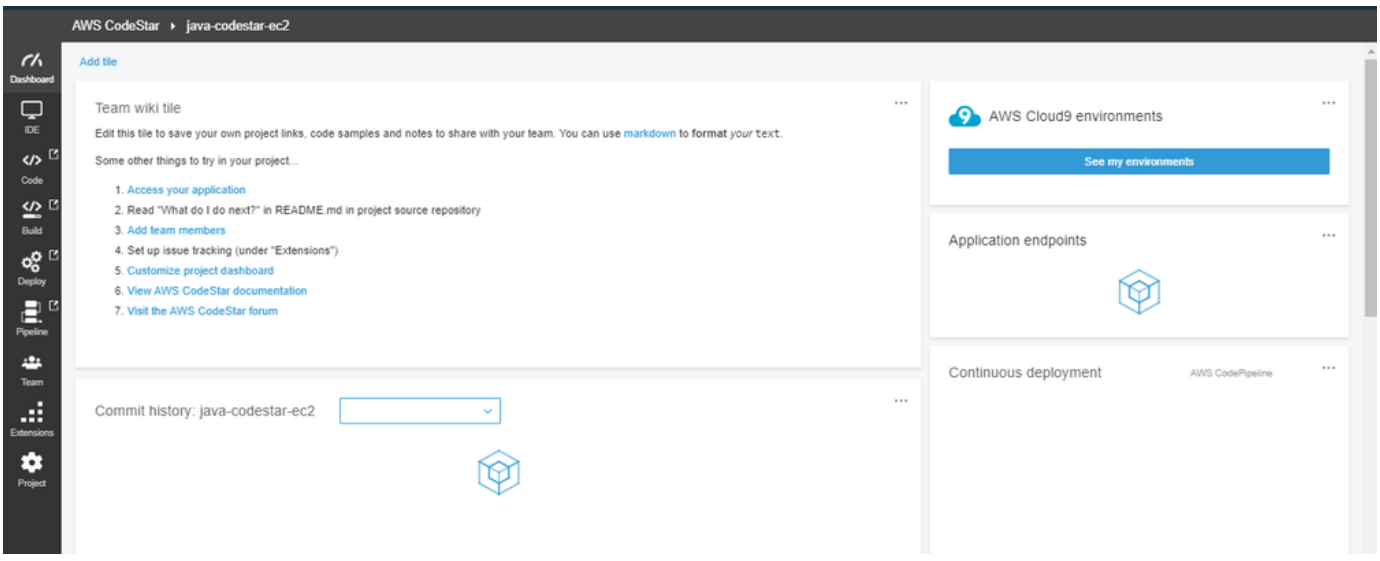
To use AWS Cloud9, you need:

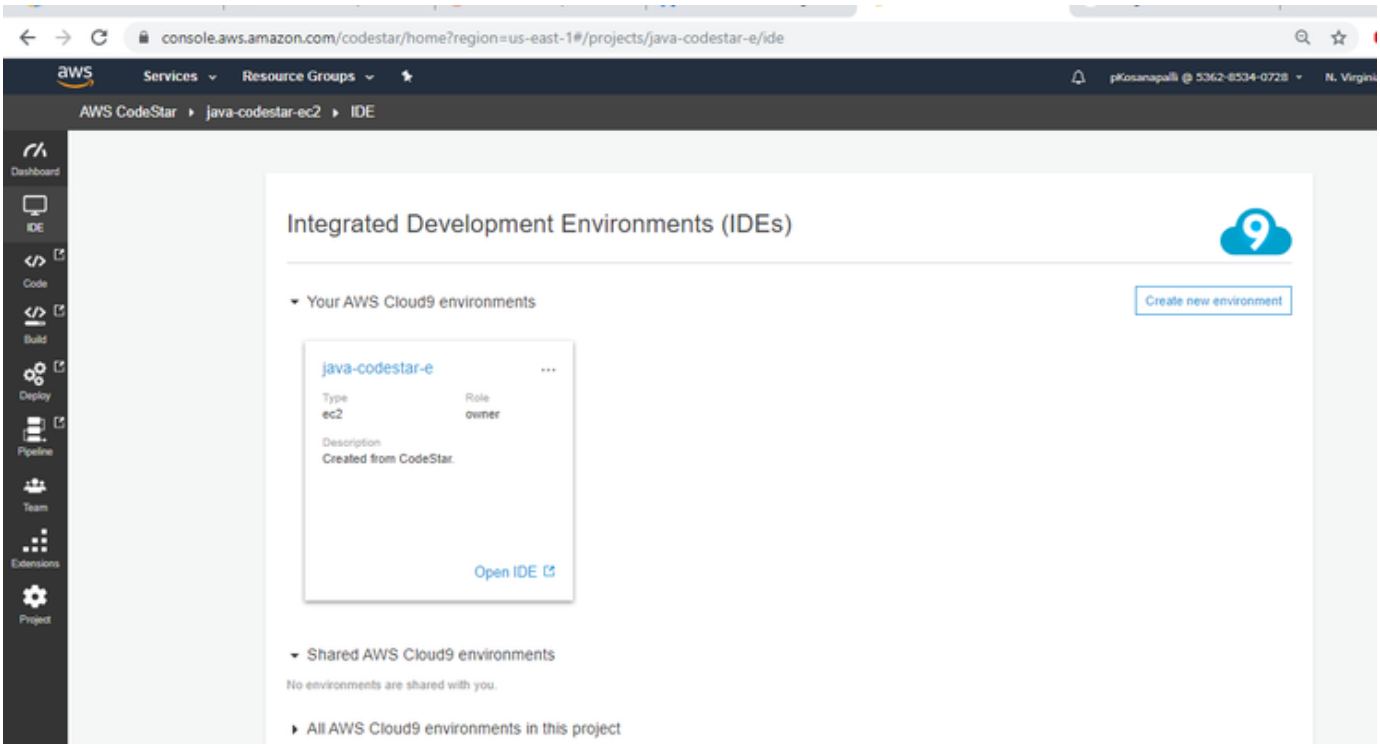
- An IAM user that has been added as a team member to an AWS CodeStar project.
- If the AWS CodeStar project stores its source code in CodeCommit, AWS credentials for the IAM user.



Choose AWS Cloud9 Environment for a Project

- Follow these steps to create an AWS Cloud9 development environment for a AWS CodeStar project.
- Open the project in the AWS CodeStar console. On the side navigation bar, choose **IDE**. Choose **Create new environment.**





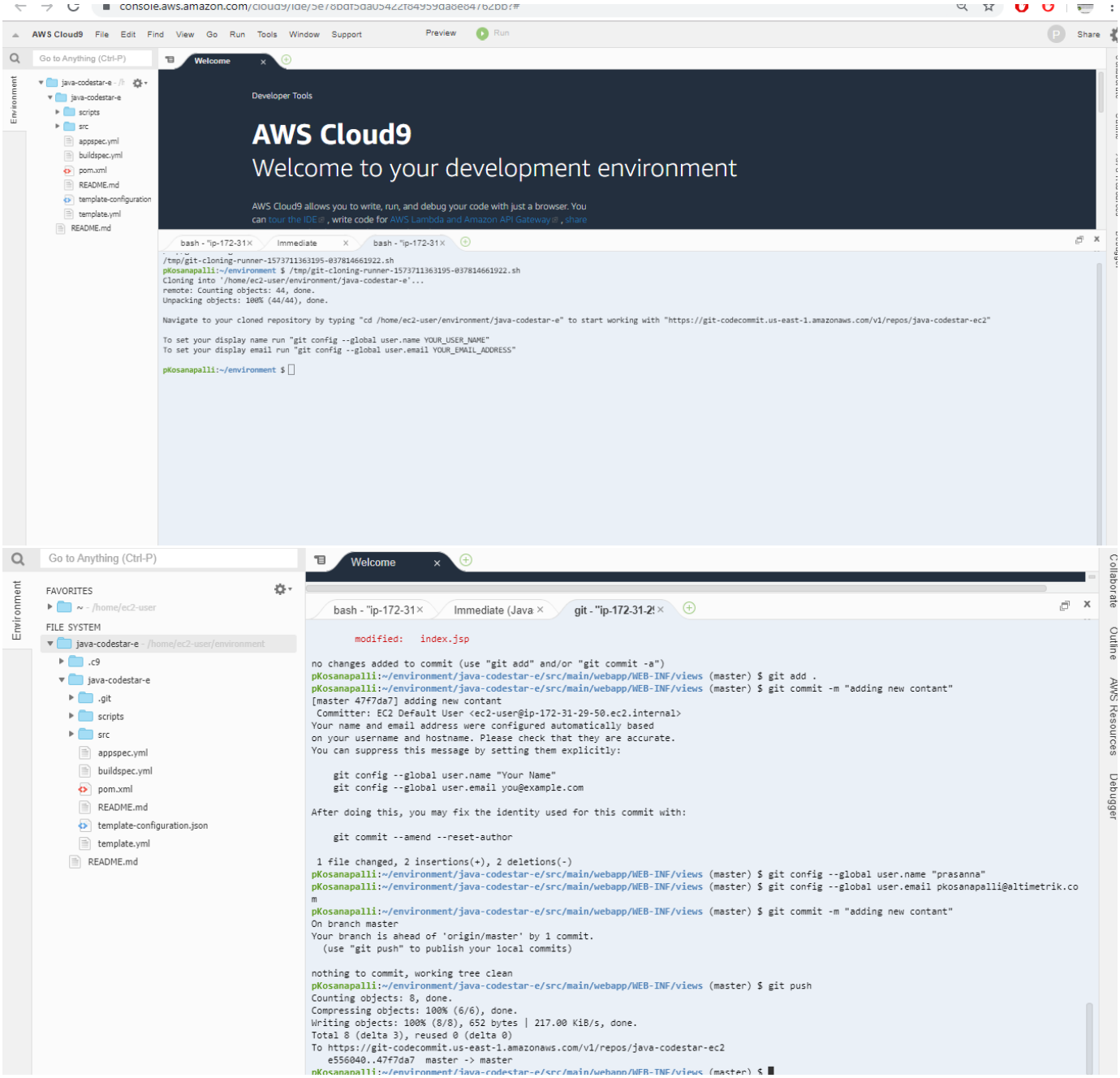
On **Set up your AWS Cloud9 environment**, customize the project defaults.

1. To change the default type of Amazon EC2 instance to host the environment, for **Pick an instance type for the IDE (not your overall project)**, choose the instance type.
2. To change the default environment name and add a description, expand **Environment name and description**, and then make your changes.
3. AWS Cloud9 uses Amazon Virtual Private Cloud (Amazon VPC) in your AWS account to communicate with the instance.
4. To change the default time period after which AWS Cloud9 shuts down the environment when it has not been used, expand **Cost-saving options**, and then change the setting.
5. Choose **Next**.

Open an AWS Cloud9 Environment for a Project

Follow these steps to open an AWS Cloud9 development environment that you created for an AWS CodeStar project.

1. With the project open in the AWS CodeStar console, on the side navigation bar, choose **IDE**.
2. For **Your AWS Cloud9 environments** or **Shared AWS Cloud9 environments**, choose **Open IDE** for the environment you want to open.
3. top screen is editor and bottom console is command prompt. you can work this like cli and editor, left-side is navigator.



Delete an AWS Cloud9 Environment from a Project

When you delete a project and all its AWS resources from AWS CodeStar, all related AWS Cloud9 development environments that were created with the AWS CodeStar console are also deleted and cannot be recovered. You can delete a development environment from a project without deleting the project.

1. With the project open in the AWS CodeStar console, in the side navigation bar, choose **IDE**.
2. On the tile for the environment you want to delete, choose the ellipses (**...**).
3. Enter the name for the development environment, and then choose **Delete**.

Use GitHub with AWS Cloud9

For AWS CodeStar projects that have their source code stored in GitHub, the AWS CodeStar console doesn't support working with AWS Cloud9 development environments directly. However, you can use the AWS Cloud9 console to work with source code in GitHub repositories.

1. Use the AWS Cloud9 console to create an AWS Cloud9 development environment.
2. Use the AWS Cloud9 console to open the development environment.

3. In the IDE, use a terminal session to connect to the GitHub repository (a process known as *cloning*). If a terminal session isn't running, on the menu bar in the IDE, choose **Window, New Terminal**. For the commands to use to clone the GitHub repository, To navigate to the main page of the GitHub repository, with the project open in the AWS CodeStar console, on the side navigation bar, choose **Code**.
4. Use the **Environment** window and editor tabs in the IDE to view, change, and save code.
5. Use Git in the IDE's terminal session to push your code changes to the repository and periodically pull code changes from others from the repository.

Use Eclipse with AWS CodeStar

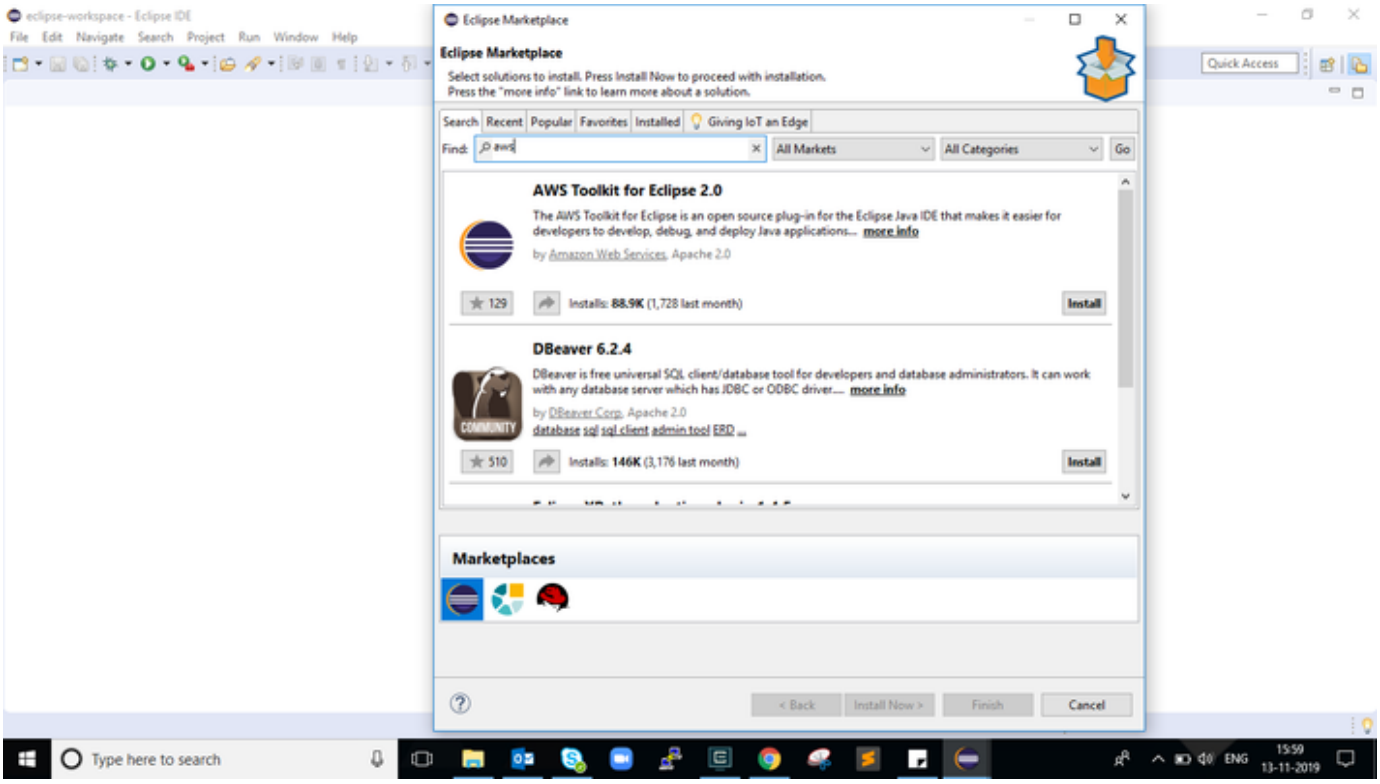
You can use Eclipse to make code changes and develop software in an AWS CodeStar project. You can edit your AWS CodeStar project code with Eclipse and then commit and push your changes to the source repository for the AWS CodeStar project.

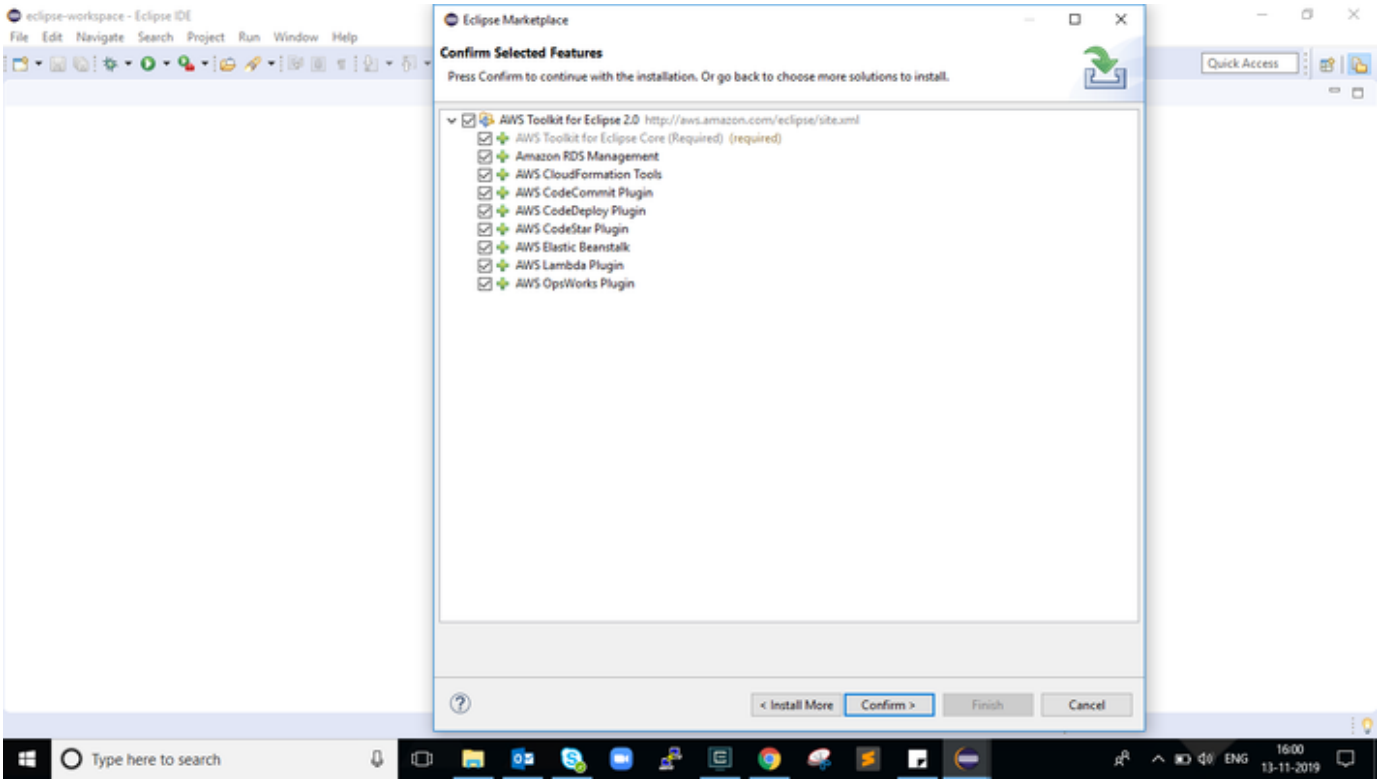
To use Eclipse, you also need:

- An IAM user that has been added to an AWS CodeStar project as a team member.
- If the AWS CodeStar project stores its source code in CodeCommit, Git credentials (user name and password) for the IAM user.
- Sufficient permissions to install Eclipse and the AWS Toolkit for Eclipse on your local computer.

Step 1: Install AWS Toolkit for Eclipse

The Toolkit for Eclipse is a software package you can add to Eclipse. It is installed and managed in the same way as other software packages in Eclipse. The AWS CodeStar toolkit is included as part of the Toolkit for Eclipse.





To install the Toolkit for Eclipse with the AWS CodeStar module

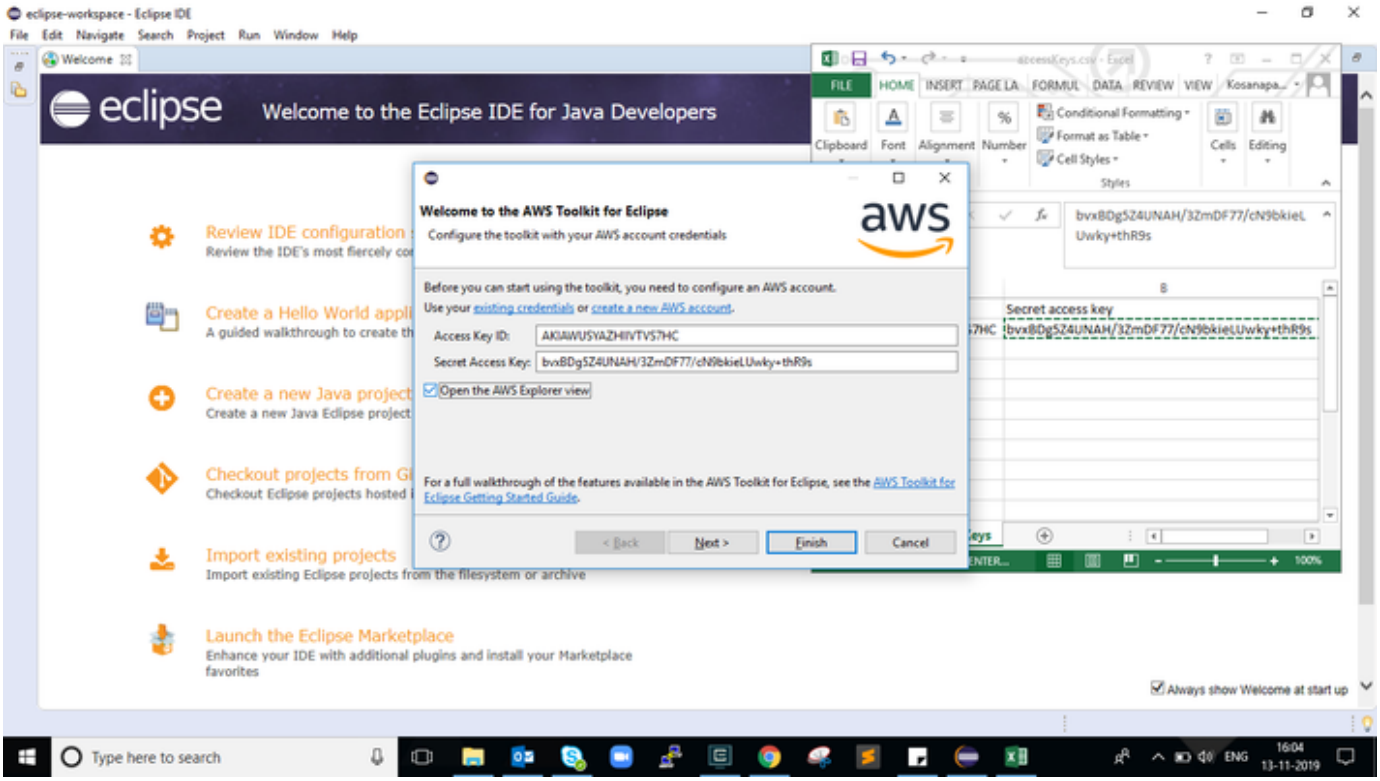
1. Install Eclipse on your local computer.
2. Download and install the Toolkit for Eclipse.
3. In Eclipse, choose **Help**, and then choose **Install New Software**.
4. In **Available Software**, choose **Add**.
5. In **Add Repository**, choose **Archive**, browse to the location where you saved the .zip file, and open the file. Leave **Name** blank, and then choose **OK**.
6. In **Available Software**, choose **Select all** to select **AWS Core Management Tools** and **Developer Tools**, and then choose **Next**.
7. In **Install Details**, choose **Next**.
8. In **Review Licenses**, review the license agreements. Choose **I accept the terms of the license agreement**, and then choose **Finish**. Restart Eclipse.

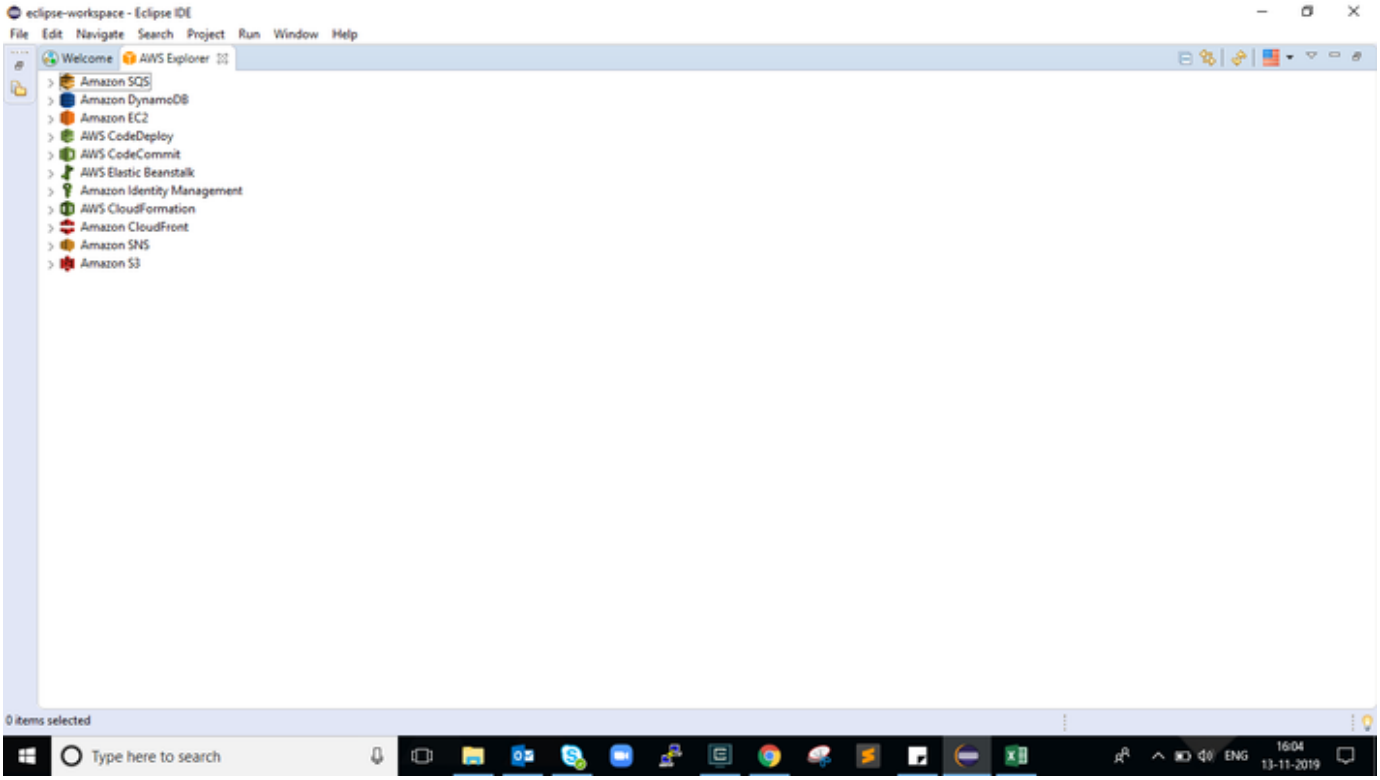
Step 2: Import Your AWS CodeStar Project to Eclipse

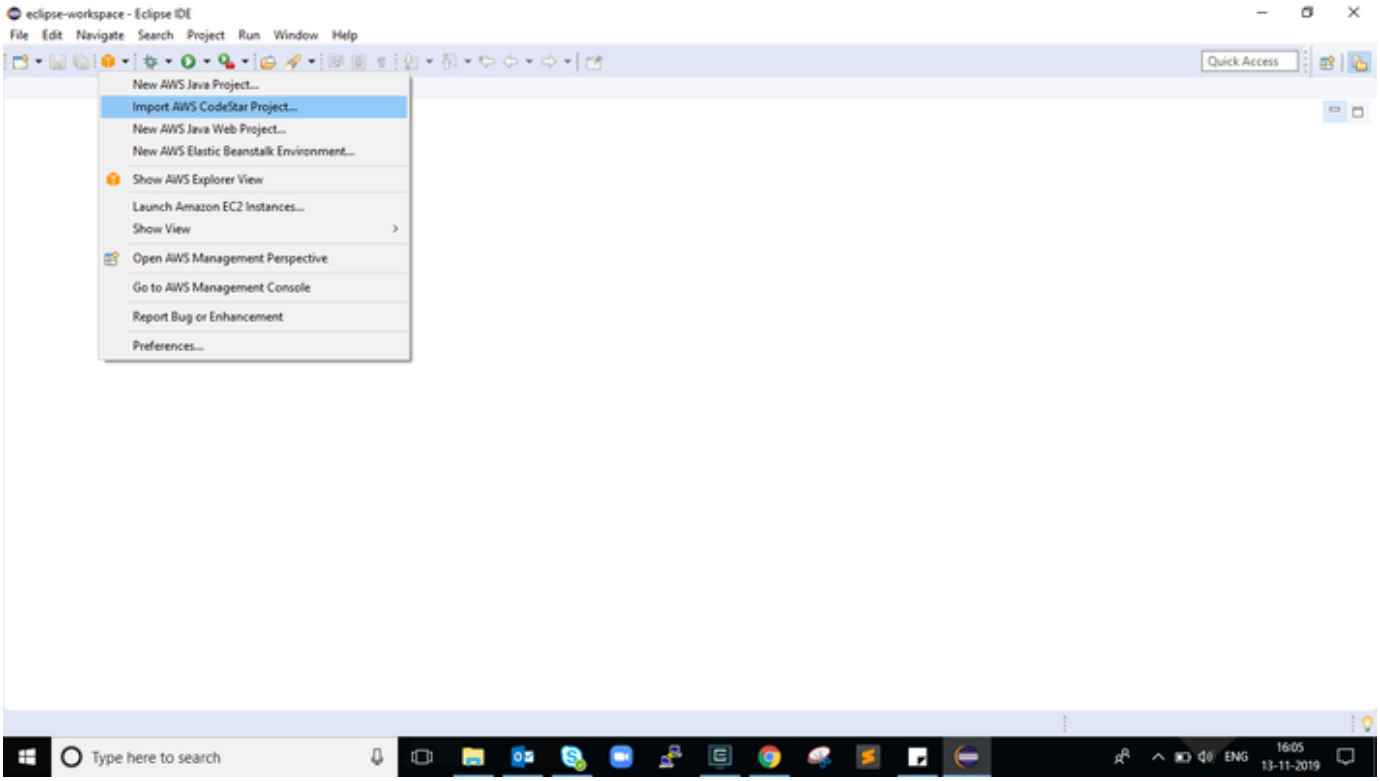
After you have installed the Toolkit for Eclipse, you can import AWS CodeStar projects and edit, commit, and push code from the IDE.

To import an AWS CodeStar project

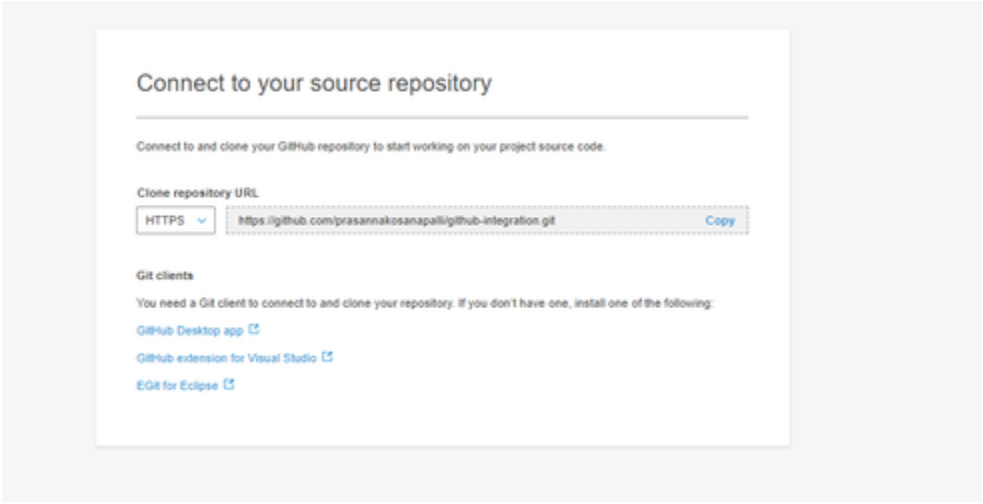
1. From the AWS menu, choose **Import AWS CodeStar Project**. Alternatively, choose **File**, and then choose **Import**. In **Select**, expand **A WS**, and then choose **AWS CodeStar Project**.Choose **Next**.


eclipse-workspace - Eclipse IDE
File Edit Navigate Search Project Run Window Help
Welcome
eclipse
Welcome to the Eclipse IDE for Java Developers
Review IDE configuration
Review the IDE's most fiercely co
Create a Hello World appli
A guided walkthrough to create th
Create a new Java project
Create a new Java Eclipse project
Checkout projects from Gi
Checkout Eclipse projects hosted
Import existing projects
Import existing Eclipse projects from the filesystem or archive
Launch the Eclipse Marketplace
Enhance your IDE with additional plugins and install your Marketplace favorites
Always show Welcome at start up
Welcome to the AWS Toolkit for Eclipse
Configure the toolkit with your AWS account credentials
aws
Before you can start using the toolkit, you need to configure an AWS account.
Use your existing credentials or create a new AWS account.
Access Key ID: AKIAWUSYAZHIVTVS7HC
Secret Access Key: bvxBDg5Z4UNAH/3ZmDF77/cN9bkieLUwky+thR9s
Open the AWS Explorer view
For a full walkthrough of the features available in the AWS Toolkit for Eclipse, see the AWS Toolkit for Eclipse Getting Started Guide.
< Back
Next >
Finish
Cancel
Secret access key
bvxBDg5Z4UNAH/3ZmDF77/cN9bkieLUwky+thR9s
Type here to search
ENG
16:04
13-11-2019


eclipse-workspace - Eclipse IDE
File Edit Navigate Search Project Run Window Help
Welcome
AWS Explorer
Amazon SQS
Amazon DynamoDB
Amazon EC2
AWS CodeDeploy
AWS CodeCommit
AWS Elastic Beanstalk
Amazon Identity Management
AWS CloudFormation
Amazon CloudFront
Amazon SNS
Amazon S3
0 items selected
Type here to search
ENG
16:04
13-11-2019

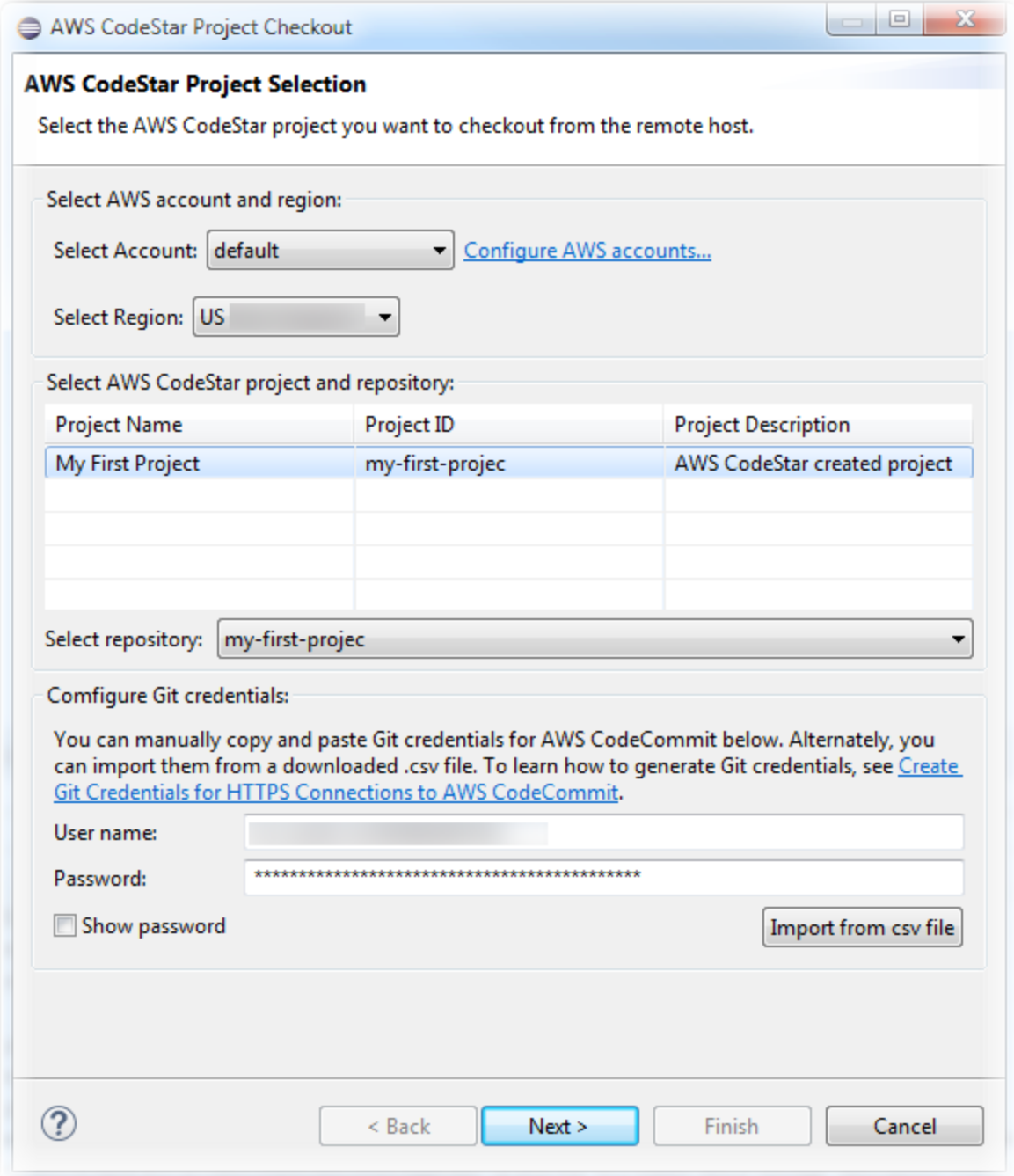


1. In **AWS CodeStar Project Selection**, choose your AWS profile and the AWS Region where the AWS CodeStar project is hosted. If you do not have an AWS profile configured with an access key and secret key on your computer, choose **Configure AWS accounts** .



1.

   In **Select AWS CodeStar project and repository**, choose your AWS CodeStar project. In **Configure Git credentials**, enter the user name and password you generated for access to the project's repository. Choose **Next**.



2. All branches of the project's repository are selected by default. If you don't want to import one or more branches, clear the boxes, and then choose **Next**.
3. In **Local Destination**, choose a destination where the import wizard creates the local repo on your computer, and then choose **Finish**.
4. In **Project Explorer**, expand the project tree to browse the files in the AWS CodeStar project.

Step 3: Edit AWS CodeStar Project Code in Eclipse

After you have imported an AWS CodeStar project into an Eclipse workspace, you can edit the code for the project, save your changes, and commit and push your code to the source repository for the project. This is the same process you follow for any Git repository using the EGit plugin for Eclipse.

To edit project code and make your first commit to the source repository for an AWS CodeStar project

1. In **Project Explorer**, expand the project tree to browse the files in the AWS CodeStar project.
2. Edit one or more code files and save your changes.
3. When you are ready to commit your changes, open the context menu for that file, choose **Team**, and then choose **Commit**.You can skip this step if the **Git Staging** window is already open in your project view.
4. In **Git Staging**, stage your changes by moving changed files into **Staged Changes**. Enter a commit message in **Commit Message**, and

then choose **Commit and Push**.
5. for more details go to eclipse section in workstation development Developers workstation setup Guide.

To view the deployment of your code changes, return to the dashboard for your project.

Use Visual Studio with AWS CodeStar

You can use Visual Studio to make code changes and develop software in an AWS CodeStar project.

To use Visual Studio to edit code in the source repository for an AWS CodeStar project, you must install a version of the AWS Toolkit for Visual Studio that supports AWS CodeStar. You must be a member of the AWS CodeStar project team with the owner or contributor role.

To use Visual Studio, you also need:
- An IAM user that has been added to an AWS CodeStar project as a team member.
- AWS credentials for your IAM user (for example, your access key and secret key).
- Sufficient permissions to install Visual Studio and the AWS Toolkit for Visual Studio on your local computer.

The Toolkit for Visual Studio is a software package you can add to Visual Studio. It is installed and managed in the same way as other software packages in Visual Studio.

To install the Toolkit for Visual Studio with the AWS CodeStar module and configure access to your project repository

1. Install Visual Studio on your local computer.
2. Download and install the Toolkit for Visual Studio and save the .zip file to a local folder or directory. On the **Getting Started with the AWS Toolkit for Visual Studio** page, enter or import your AWS credentials, and then choose **Save and Close**.
3. In **Visual Studio**, open **Team Explorer**. In **Hosted Service Providers**, find **CodeCommit**, and choose **Connect**.
4. In **Manage Connections**, choose **Clone**. Choose your project's repository and the folder on your local computer where you want to clone the repository, and then choose **OK**.
5. If you are prompted to create Git credentials, choose **Yes**. The toolkit attempts to create credentials on your behalf. Save the credentials file in a secure location. This is the only opportunity you have to save these credentials. If the toolkit cannot create credentials on your behalf, or if you chose **No**, you must create and provide your own Git credentials.
6.

   When you have finished cloning the project, you're ready to start editing your code in Visual Studio and committing and pushing your changes to your project's repository in CodeCommit.

see the workstation developer vs code developer section Developers workstation setup Guide.

thats all done !!!