# **Week1 Reading/Resources**

**What is DevOps?**

DevOps is the combination of cultural philosophies, practices, and tools that increases an organization’s ability to deliver applications and services at high velocity. Put simply, DevOps is a way for development teams and operations teams to work better together and ultimately share the responsibility of the software they build. For more information about what DevOps is and how it works, see: [What is DevOps?](https://aws.amazon.com/devops/what-is-devops/)

**AWS CodeCommit**

AWS CodeCommit is a secure, highly scalable, managed source-control service that hosts private Git repositories. For more information about what CodeCommit is and how it works, see: [AWS CodeCommit User Guide](https://docs.aws.amazon.com/codecommit/latest/userguide/welcome.html) After a repository is created in CodeCommit, you can securely interact with it with the proper authentication and access permissions. For more information about authentication and access in CodeCommit, see: [Authentication and access control for AWS CodeCommit](https://docs.aws.amazon.com/codecommit/latest/userguide/auth-and-access-control.html)

**AWS CodeBuild**

You can use AWS CodeBuild for continuous integration (CI) practices. It compiles your source code, runs unit tests, and produces artifacts that are ready to deploy. It also reduces the need to provision, manage, and scale your own build servers. With CodeBuild, you only pay for the service when you are using it. For more information about what CodeBuild is and how it works, see: [AWS CodeBuild User Guide](https://docs.aws.amazon.com/codebuild/latest/userguide/welcome.html) In this course, you also learned about two important things to configure so you can get your build processes into CodeBuild: a build project and a build specification (buildspec) file.

* **Build project** - Includes information about how to run a build, including where to get the source code, which build environment to use, which build commands to run, and where to store the build output.
* **Buildspec file** - Is a collection of build commands and related settings, in YAML format, that CodeBuild uses to run a build.

For more information about build projects, see: [Working with build projects](https://docs.aws.amazon.com/codebuild/latest/userguide/working-with-build-projects.html) For more information on buildspec files, see: [Build specification reference for CodeBuild](https://docs.aws.amazon.com/codebuild/latest/userguide/build-spec-ref.html)

**Branching strategies**

A branch in Git is a pointer to a commit—that’s it! There are many ways to design your branching strategies. For reference, see the strategies used in the following resources:

* [Understanding the GitHub flow](https://guides.github.com/introduction/flow/)
* [Gitflow Workflow](https://www.atlassian.com/git/tutorials/comparing-workflows/gitflow-workflow)

# **Week 2 Reading/Resources**

**Testing**

To ensure quality, it’s important for teams to incorporate testing into the software development lifecycle at different stages of the continuous integration and continuous delivery (CI/CD) pipeline. Overall, testing should start as early as possible.

While there are many kinds of application tests, the three mentioned in this week are:

* **Regression testing** - Tests to ensure that previously developed applications don’t break with new changes
* **Integration** **testing** - Tests to ensure separately developed modules work together as expected
* **Unit** **testing** - Tests a specific section of code to ensure the code does what it is expected to do

The exercises in this course used pytest for unit testing. For more information, see: [pytest: helps you write better programs](https://docs.pytest.org/en/6.2.x/)

**Automate testing**

AWS CodeBuild will provision a clean and consistent environment to perform various application tests. You can also create reports in CodeBuild that contain details about tests that are run during builds.

For more information about CodeBuild and test reporting, see: [Working with test reporting in AWS CodeBuild](https://docs.aws.amazon.com/codebuild/latest/userguide/test-reporting.html)

If you’re interested, check out this GitHub repository, which contains Dockerfiles of official, curated CodeBuild Docker images: [aws-codebuild-docker-images](https://github.com/aws/aws-codebuild-docker-images)

**AWS CodePipeline**

You can automate your release process by using AWS CodePipeline to test your code and run your builds with CodeBuild.

CodePipeline is a continuous delivery (CD) service that you can use to model, visualize, and automate the steps required to release your code. This process includes building your code. A pipeline is a workflow construct that describes how code changes go through a release process.

**AWS CodeDeploy**

If you want to read more about in-place and blue/green deployments with AWS CodeDeploy, see: [Working with deployments in CodeDeploy](https://docs.aws.amazon.com/codedeploy/latest/userguide/deployments.html)

To use CodeDeploy on Amazon Elastic Compute Cloud (Amazon EC2) instances or on-premises servers, the CodeDeploy agent must be installed first. For more information, see: [Install the CodeDeploy agent](https://docs.aws.amazon.com/codedeploy/latest/userguide/codedeploy-agent-operations-install.html)

**Troubleshooting deployment details and errors**

You can view the log data created by a CodeDeploy deployment by setting up the Amazon CloudWatch Logs agent to view aggregated data in the CloudWatch console or by logging into an individual instance to review the log file. For more information, see: [View log data for CodeDeploy EC2/On-Premises deployments](https://docs.aws.amazon.com/codedeploy/latest/userguide/deployments-view-logs.html)

For general troubleshooting tips for CodeDeploy, see the following resources:

* [General troubleshooting issues](https://docs.aws.amazon.com/codedeploy/latest/userguide/troubleshooting-general.html)
* [Troubleshoot EC2/On-Premises deployment issues](https://docs.aws.amazon.com/codedeploy/latest/userguide/troubleshooting-deployments.html)
* [Troubleshoot AWS Lambda deployment issues](https://docs.aws.amazon.com/codedeploy/latest/userguide/troubleshooting-deployments-lambda.html)
* [Error codes for AWS CodeDeploy](https://docs.aws.amazon.com/codedeploy/latest/userguide/error-codes.html)