**DevOps:**

The DevOps is a combination of two words, one is software Development, and second is Operations. This allows a single team to handle the entire application lifecycle, from development to **testing, deployment**, and **operations**.

DevOps Advantages and Disadvantages

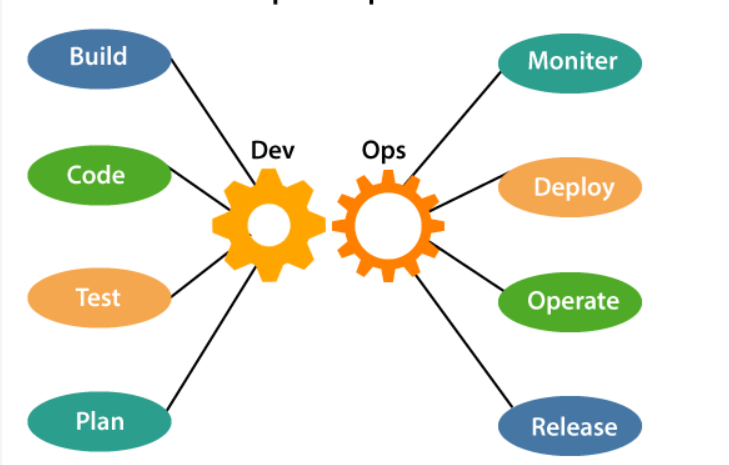
Here are some advantages and disadvantages that DevOps can have for business, such as:

Advantages

* DevOps is an excellent approach for quick development and deployment of applications.
* It responds faster to the market changes to improve business growth.
* DevOps escalate business profit by decreasing software delivery time and transportation costs.
* DevOps clears the descriptive process, which gives clarity on product development and delivery.
* It improves customer experience and satisfaction.
* DevOps simplifies collaboration and places all tools in the cloud for customers to access.
* DevOps means collective responsibility, which leads to better team engagement and productivity.

Disadvantages

* DevOps professional or expert's developers are less available.
* Developing with DevOps is so expensive.
* Adopting new DevOps technology into the industries is hard to manage in short time.
* Lack of DevOps knowledge can be a problem in the continuous integration of automation projects.



**What is Continuous Integration, Continuous Delivery, and Continuous Deployment?**

* **Continuous integration** is a software development method where members of the team can integrate their work at least once a day. In this method, every integration is checked by an automated build to search the error.
* **Continuous delivery** is a software engineering method in which a team develops software products in a short cycle. It ensures that software can be easily released at any time.
* **Continuous deployment** is a software engineering process in which product functionalities are delivered using automatic deployment. It helps testers to validate whether the codebase changes are correct, and it is stable or not.

A CI/CD pipeline is a runnable specification of the steps that any developer should perform to deliver a new version of any software. Failure in each and every stage triggers a notification via email, Slack, or other communication platforms. It enables responsible developers to know about the important issues.

**Jenkins:**

Jenkins is a powerful application that allows continuous integration and continuous delivery of projects, regardless of the platform you are working on. It is a free source that can handle any kind of build or continuous integration. You can integrate Jenkins with a number of testing and deployment technologies.

Jenkins is an open source automation tool written in Java programming language that allows continuous integration.

Automated testing:

[Jenkins - Automated Testing (tutorialspoint.com)](https://www.tutorialspoint.com/jenkins/jenkins_automated_testing.htm)

[Jenkins for Test Automation : Tutorial | BrowserStack](https://www.browserstack.com/guide/jenkins-for-test-automation)

<https://www.softwaretestinghelp.com/integration-of-jenkins-with-selenium-webdriver/>

Jenkins Pipelines:

[Jenkins Pipeline Tutorial: How to Create JenkinsFile (Example) (guru99.com)](https://www.guru99.com/jenkins-pipeline-tutorial.html)

[Jenkins Pipeline: How to Create JenkinsFile - Eduguru](https://blog.eduguru.in/devops/jenkins-pipeline-how-to-create-jenkinsfile)

[How To Create Jenkins multibranch pipeline (lambdatest.com)](https://www.lambdatest.com/blog/how-to-create-jenkins-multibranch-pipeline/)

Different DevOps tools:

Jenkins, Prometheus, Kubernetes, Docker, Hudson, GIT etc

For more details: [30 Best DevOps Tools & Technologies (2021 List) (guru99.com)](https://www.guru99.com/devops-tools.html)

**Docker:**

Docker is a centralized platform for packaging, deploying, and running applications.

Docker is an **open-source centralized platform designed** to create, deploy, and run applications. Docker uses **container** on the host's operating system to run applications. It allows applications to use the same **Linux kernel** as a system on the host computer, rather than creating a whole virtual operating system. Containers ensure that our application works in any environment like development, test, or production.

### Docker Containers

Docker containers are the **lightweight** alternatives of the virtual machine. It allows developers to package up the application with all its libraries and dependencies, and ship it as a single package. The advantage of using a docker container is that you don't need to allocate any RAM and disk space for the applications. It automatically generates storage and space according to the application requirement.

Advantages of Docker

There are the following advantages of Docker -

* It runs the container in seconds instead of minutes.
* It uses less memory.
* It provides lightweight virtualization.
* It does not a require full operating system to run applications.
* It uses application dependencies to reduce the risk.
* Docker allows you to use a remote repository to share your container with others.
* It provides continuous deployment and testing environment.

# **Docker Container and Image**

Docker container is a running instance of an image. You can use Command Line Interface (CLI) commands to run, start, stop, move, or delete a container. You can also provide configuration for the network and environment variables. Docker container is an isolated and secure application platform, but it can share and access to resources running in a different host or container.

An image is a read-only template with instructions for creating a Docker container. A docker image is described in text file called a **Dockerfile**, which has a simple, well-defined syntax. An image does not have states and never changes. Docker Engine provides the core Docker technology that enables images and containers.

$ docker run hello-world

1) **docker:** It is docker engine and used to run docker program. It tells to the operating system that you are running docker program.

2) **run:** This subcommand is used to create and run a docker container.

3) **hello-world:** It is a name of an image. You need to specify the name of an image which is to load into the container.

A Dockerfile is a text document that contains commands that are used to assemble an image. We can use any command that call on the command line. Docker builds images automatically by reading the instructions from the Dockerfile.

Difference between Docker and Kubernetes:

[Difference between Kubernetes vs Docker - Great Learning (mygreatlearning.com)](https://www.mygreatlearning.com/blog/difference-between-kubernetes-vs-docker/)

**Prometheus:**

[How To Create Jenkins multibranch pipeline (lambdatest.com)](https://www.lambdatest.com/blog/how-to-create-jenkins-multibranch-pipeline/)

[With Grafana and Prometheus | Grafana Labs](https://grafana.com/docs/grafana/latest/getting-started/getting-started-prometheus/)

[Prometheus | Grafana Labs](https://grafana.com/oss/prometheus/)

**Agile Release management:**

[What Is Release Management and What’s Its Role in DevOps? | Bunnyshell |](https://www.bunnyshell.com/blog/what-is-release-management)

[What is a DevOps Release Management and its Practices (staragile.com)](https://staragile.com/blog/devops-release-management)