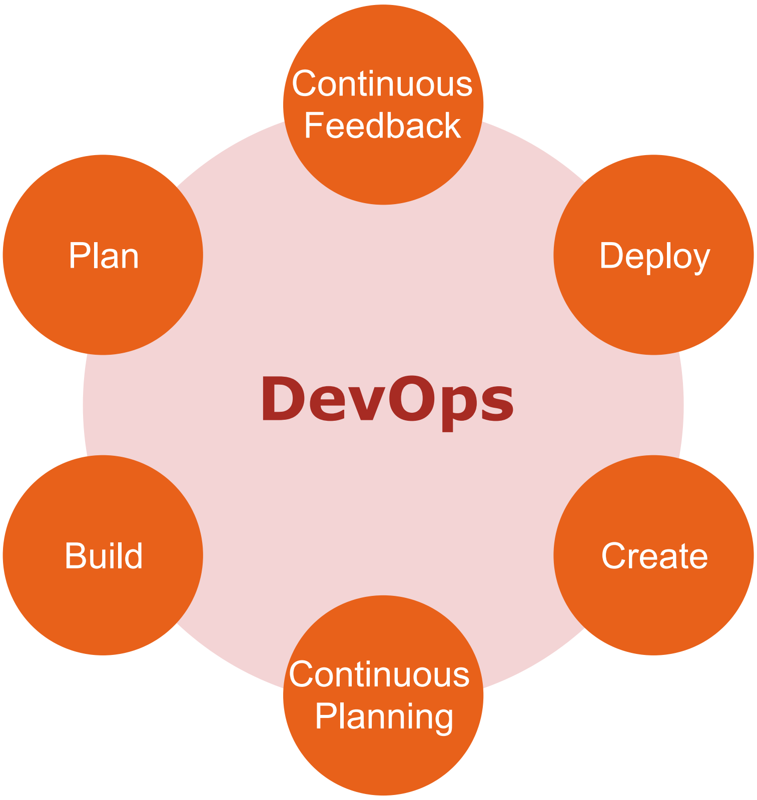
A DevOps Enabler Tool

In this lesson, you will learn how Docker containers ease deployment.

If you’re in a hurry, you can safely skip this chapter and head straight to the *Get Docker up and Running* chapter. This *Why Docker?* chapter is there for those that want to know why containers should be used.

Docker is an engine that runs containers. As a tool, containers allow you to solve many challenges created in the growing DevOps trend.



In DevOps, the Dev and Ops teams have conflicting goals:

Dev Team Seeks	Ops Team Seeks
Frequent deployments and updates	Stability of production apps

Easy creation of new resources	Manage infrastructure, not applications Monitoring and control

As an agile developer, I want to frequently publish my applications so that deployment becomes a routine. The rationale behind this is that this agility makes the “go-to production” event a normal, frequent, completely mastered event instead of a dreaded disaster that may awake monsters who hit me one week later. On the other hand, it is the Ops team that has to face the user if anything goes wrong in deployment - so they naturally want stability.

Containers make deployment easy. Deploying is as simple as running a new container, routing users to the new one, and trashing the old one. It can even be automated by orchestration tools. Since it’s so easy, we can afford to have many containers serving a single application for increased stability during updates.

If you don’t use containers, Ops need to handle your hosting environment: runtimes, libraries, and OS needed by your application. On the other hand, when using containers, they need one single methodology that can handle the containers you provide no matter what’s inside them. You may as well use .NET Core, Java, Node.JS, PHP, Python, or another development tool: it doesn’t matter to them as long as your code is containerized. This is a considerable advantage for containers when it comes to DevOps.

In the *Docker with Common Development Profiles* chapter, we’ll see how to create container images for specific development technologies. However, your Ops don’t see how you create the images as they only see container images.

In the next lesson, we will go over another advantage of Docker containers.