🏠 › Getting Started

Getting Started

Welcome to the DevOps with Docker course! This course is designed to be completed sequentially, from start to finish. Each part builds on the previous one, so it's important to read the material carefully and complete the exercises to develop the necessary skills.

To pass the course, you will need to complete all of the exercises. However, one exercise can be skipped per part, except for the exercises marked as mandatory. The mandatory exercises cannot be skipped.

The exercises are designed to reinforce the material covered in each part and are placed at strategic points in the course to ensure that you have learned the necessary skills before attempting each exercise. You can complete the exercises at your own pace, and there is no deadline for submission outside of the last date when the entire course ends.

When submitting your exercises, please follow the instructions provided in the course material. Exercises should be submitted in a specific format, which will be outlined in the instructions for each exercise.

We hope that you enjoy the course and find it informative and engaging. Good luck!

Prerequisites

The course is suitable for anyone interested in Docker or containerization and has at least some experience with the Linux command line. It also helps if you have some experience in web development.

About different architectures and contributing

Please note that while Docker runs on all major operating systems and even on ARM architecture, this course material may not cover platform-specific details for all operating systems. However, we've had students successfully complete the course using a variety of machines and operating systems.

If you encounter any issues while working through the course material on your particular system, we recommend consulting the Docker documentation or seeking help on the course forums. Our community is here to support you and help you succeed in the course!

We welcome contributions to the course material from students and other members of the DevOps community! If you notice any mistakes, typos, or errors in the material, please consider submitting a pull request to the course repository on GitHub.

Thank you in advance for your contributions to this open-source project!

Grading

Passing this course requires you to do the exercises for each part. This means generally every exercise, but you are allowed to skip one non-mandatory exercise in each part. Some of the exercises are mandatory and can not be skipped.

This course is worth 1-3 credits depending on the completed parts. Completing part 1 gives you 1 credit. Completing parts 1 and 2 is worth 2 credits. Completing all of the parts will grant you 3 credits.

There are additional instructions for completion after each part and at the end of this page.

Learning objectives

Part 1: DevOps with Docker (TKT21036)

- Understand the fundamental concepts of Docker, including images and containers.
- Learn how to build Docker images for existing projects and run them.
- Understand how Docker can simplify the development process.

Part 2: DevOps with Docker: docker-compose (TKT21037)

- Learn how to manage complex multi-container applications with Docker Compose.
- Understand the role of Docker Compose in container orchestration.
- Practice deploying and managing real-world applications using Docker Compose.

Part 3: DevOps with Docker: security and optimization (TKT21038)

- Learn how to optimize Docker images for production, including reducing image size and improving security.
- Understand the limitations of using Docker Compose in production environments and the need for more advanced orchestration tools.
- Explore alternative container orchestration solutions, including Kubernetes.

Where to find information about the course?

All of the details you need to complete the course should be found on this page. If something is missing or unclear after reading this page, please contact matti.luukkainen@helsinki.fi or get in touch through Discord.

Discord

This course has a Discord group where we discuss everything about the course. Support is available almost 24/7, with the discussion being in both English and Finnish.

Join our discord group here.

All inappropriate, degrading or discriminating comments on the channel are prohibited and will lead to action taken against the commenter.

Warning: Before installing Docker

Containers leverage the power of your own operating system. As such **by default** any containerized application, or user who has external access to your container, would have super user privileges to your computer.

I will try my best to alert you of potential risks as we encounter them, but due to the structure of the course we will focus on security in part 3.

Please keep this in mind as you move through the installation and exercises. If you ever feel unsure about what you're doing, come to the Discord channel and have a chat with us.

Installing Docker

Use the official documentation to find download instructions for docker-ce for the platform of your choice:

Ubuntu

MacOS

Windows

Confirm that Docker installed correctly by opening a terminal and running `docker -v` to see the installed version.

> 💡 **DOCKER GROUP**
> To avoid writing sudos you may consider adding yourself to docker group
>
> Keep in mind that if you do so, you can now run containers without sudo and containers give you super user access to the computer.

Deadline

The sign up for ECTS credits and the course ends 16.6.2024! After that course is locked and submissions can no longer be made or credits earned. As the certificate is received through submissions, you have to submit everything before the course ends. More details under completion and after each part.

General guidance

Do not alter the code of the projects, unless by pull-requests to the original projects

You do not need to touch Ruby, Java, Javascript or Python code during this course. You may have to read their error messages.

Visit the Discord channel if you are stuck!

How to submit the exercises

Make a repository to GitHub and publish your solutions in clearly ordered files / folders. If you need help publishing using Git you should refer to their guide. Make sure that the repository is available to us, either by using a public repository or a private repository and adding Jakousa and mluukkai as collaborators.

Most of the exercises will be focused on a Dockerfile and/or docker-compose.yml. In those cases, submitting the file is enough. In other cases, a picture or copy-paste from your command line or a link to Docker Hub and/or project inside the repository is enough. For the command line exercises at start the command script may be helpful.

Because the course exercises are designed to build upon each other, it's more important that you document the exercises for yourself. We will be looking at the submissions of the later exercises as they are more demanding.

When you have completed a part, use the submission application to mark your exercises. You can **not** edit a submission, so make sure you have completed enough exercises for a part before submitting.

Completing

The certificate is available from the small icon beneath your submissions!

After you have returned all of the required exercises and wish to end your course completion and want the ECTS credits press the following button in the submission application (ignore the message about an "exam" as there is no exam in this course):

You are entitled to **3 credits** based on the exercises you have submitted

Certificate 🇬🇧

I have completed the course (exam done in Moodle and will not do more exercises) and want to get university credits registered.

After that, double-check that the application has the message "Course marked as completed" and the date. If the date is not visible, we have not been notified.

My submissions

part	exercises	hours	github
1	17	15	https://github.com/Jakousa
2	10	14	https://github.com/Jakousa
3	8	17	https://github.com/Jakousa
total	35	46	

Course marked as completed 5.2 2021

You are entitled to **3 credits** based on the exercises you have submitted

Certificate 🇬🇧

If you want to continue your progress, press here.

✏ Edit this page