

About the CodeRefinery project and CodeRefinery workshops in general

📌 Objectives

- Discuss what CodeRefinery is and how we got here
- Understand about the challenges to define our target audience

CodeRefinery is a [Nordic e-Infrastructure Collaboration \(NeIC\)](#) project that has started in October 2016 and is currently funded until February 2025. We are working on the continuation plans.

The funding from 2022-2025 is designed to keep this project active beyond 2025 by forming a support network and building a community of instructors and contributors.

💬 History

The CodeRefinery project idea grew out of two [SeSE](#) courses given at KTH Stockholm in 2014 and 2016. The project proposal itself was submitted to NeIC in 2015, accepted in 2015, and started in 2016.

We have started by porting own lessons to the Carpentries teaching style and format, and collaboratively and iteratively grew and improved the material to its present form.

Goals

- Develop and maintain **training material on good enough software development practices** for researchers that write code/scripts/notebooks.
- Our material addresses all academic disciplines and tries to be as programming language-independent as possible.
- Provide a [code repository hosting service](#) that is open and free for all researchers based in universities and research institutes from Nordic countries.
- Provide **training opportunities** in the Nordics using (Carpentries and) CodeRefinery training materials.
- Evolve the project towards a **community-driven project** with a network of instructors and contributors.

Community

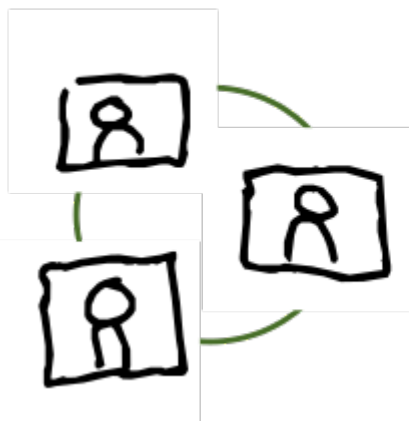


Image showing the key areas of the CodeRefinery community: Workshops, chat including help channel, online meetings and co-working, other collaborative training, eg on High Performance Computing topics.

CodeRefinery is not just workshops, we are community and want you to be part of it!

There are many different levels of involvement, from the occasional commenter in chat, CodeRefinery ambassador (people who like the project and workshops and help us spreading the word) or lesson issue creators and lesson contributors to local host, co-instructor or co-organizer.

Best **first step** in any case is to join the [CodeRefinery Zulip chat](#) or let us know about your interest at support@coderefinery.org.

Target audience

One common question we get is how do we relate to [the Carpentries](#). This section describes how we see it:

Carpentries audience

The Carpentries aims to teach computational **competence** to learners through an applied approach, avoiding the theoretical and general in favor of the practical and specific.

Mostly, learners do not need to have any prior experience in programming. One major goal

of a Carpentries workshop is to raise awareness on the tools researchers can learn/use to speed up their research.

By showing learners how to solve specific problems with specific tools and providing hands-on practice, learners develops confidence for future learning.

📌 Novices

We often qualify Carpentries learners as **novices**: they do not know what they need to learn yet. A typical example is the usage of version control: the Carpentries lesson [Version Control with Git](#) aims to give a very high level conceptual overview of Git but it does not explain how it can be used in research projects.

CodeRefinery audience

In that sense, CodeRefinery workshops differ from Carpentries workshops as we assume our audience already writes code and scripts and we aim at teaching them **best software practices**.

Our learners usually do not have a good overview of **best software practices** but are aware of the need to learn them. Very often, they know of the tools (Git, Jupyter, etc.) we are teaching but have difficulties to make the best use of them in their software development workflow.

Whenever we can, we direct learners that do not have sufficient coding experience to Carpentries workshops.

📌 Competent practitioners

We often qualify CodeRefinery learners as **competent practitioners** because they already have an understanding of their needs.

💬 Challenges related to defining our target audience

We often get the feedback “I wish I would have known X earlier!” *Competent practitioners* have run into issues with **not** caring (or not fully understanding) about version control, documentation, modularity, reproducibility before, so they are easily motivated to learn more.

For a *novice* these topics may seem unnecessary and “too much” and the workshop may feel too difficult to follow. However, the materials are designed so that one can always revisit a topic, when needed. The important part is that you know that “X” exists, and where to find more information, which is also beneficial for novices.

📌 Keypoints: CodeRefinery

- Teaches intermediate-level software development tool lessons
- It is difficult to define “best practices”, we try to teach **“good enough” practices**
- Training network for other lessons
- Publicly-funded discrete projects (3 projects actually) transitioning towards an open community project
- We have online material, teaching, and exercise sessions
- Our main target audience are competent practitioners, but also novices and experts can get something out of the workshops
- We want more people to work with us, and to work with more people