**Foundry Zksync**

In this lesson, we'll explore Layer 2 deployment on ZKsync, which involves a different compilation method compared to Ethereum. This difference arises because ZKsync uses unique opcodes. While Solidity code behaves similarly on both platforms, the low-level outputs generated by Foundry in the /out folder will not be entirely compatible with the ZKsync VM.

**Foundry ZKsync**

To get started with ZKsync, we will follow these three steps:

1. 🛠️ Install foundry-zksync
2. 🧑‍💻 Compile the Solidity contract with the --zksync flag
3. 🔄🏠 Reinstall the original Vanilla Foundry

**👀❗IMPORTANT**Installing foundry-zksync will override any existing Foundry binaries, such as forge and cast.

The GitHub resources for this course contain a link to the [Foundry ZKsync repository](https://github.com/Cyfrin/foundry-full-course-cu?tab=readme-ov-file#compiling-to-zksync-in-foundry-zksync). foundry-zksync is a fork of Foundry tailored for the ZKsync environment. The [repository](https://github.com/matter-labs/foundry-zksync) includes quick install instructions to help you set up the tool.

* First, clone the Foundry ZKsync repository in a different directory from your Foundry project. Use the git clone command to clone the repository locally on your computer.
* Once cloned, navigate to the created Foundry ZKsync directory and run the installation command:

./install-foundry-zksync

This command requires a Unix-like environment, such as WSL on Windows, or a Mac or Linux system. After running the command, verify the installation by checking the version with forge --version. A different version number will indicate the successful installation of Foundry ZKsync.

* To keep your environment flexible, you can switch to Foundry ZKsync by running foundryup-zksync. After using it, it's recommended to switch back to Vanilla Foundry by running the foundryup command. This removes ZKsync-specific flags and settings, allowing you to easily toggle between Foundry ZKsync and Vanilla Foundry as needed.