

You need to deploy your smart contract for it to be available to users of an Ethereum network.

To deploy a smart contract, you merely send an Ethereum transaction containing the compiled code of the smart contract without specifying any recipient.

Prerequisites {#prerequisites}

You should understand [Ethereum networks](#), [transactions](#) and the [anatomy of smart contracts](#) before deploying smart contracts.

Deploying a contract also costs ether (ETH) since they are stored on the blockchain, so you should be familiar with [gas and fees](#) on Ethereum.

Finally, you'll need to compile your contract before deploying it, so make sure you've read about [compiling smart contracts](#).

How to deploy a smart contract {#how-to-deploy-a-smart-contract}

What you'll need {#what-youll-need}

- Your contract's bytecode – this is generated through [compilation](#)
- ETH for gas – you'll set your gas limit like other transactions so be aware that contract deployment needs a lot more gas than a simple ETH transfer
- a deployment script or plugin
- access to an [Ethereum node](#), either by running your own, connecting to a public node, or via an API key using [a node service](#)

Steps to deploy a smart contract {#steps-to-deploy}

The specific steps involved will depend on the development framework in question. For example, you can check out [Hardhat's documentation on deploying your contracts](#) or [Foundry's documentation on deploying and verifying a smart contract](#). Once deployed, your contract will have an Ethereum address like other [accounts](#) and can be verified using [source code verification tools](#).

Related tools {#related-tools}

Remix - *Remix IDE allows developing, deploying and administering smart contracts for Ethereum like blockchains*

- [Remix](#)

Tenderly - *Web3 development platform that provides debugging, observability, and infrastructure building blocks for developing, testing, monitoring, and operating smart contracts*

- [tenderly.co](#)
- [Docs](#)
- [GitHub](#)
- [Discord](#)

Hardhat - *A development environment to compile, deploy, test, and debug your Ethereum software*

- [hardhat.org](#)
- [Docs on deploying your contracts](#)
- [GitHub](#)
- [Discord](#)

thirdweb - *Easily deploy any contract to any EVM compatible chain, using a single command*

- [Documentation](#)

Related tutorials {#related-tutorials}

- [Deploying your first smart contract](#) – *An introduction to deploying your first smart contract on an Ethereum test network.*
- [Hello World | smart contract tutorial](#) – *An easy-to-follow tutorial to create & deploy a basic smart contract on Ethereum.*
- [Interact with other contracts from Solidity](#) – *How to deploy a smart contract from an existing contract and interact with it.*
- [How to downsize your contract size](#) - *How to reduce your contract's size to keep it under the limit and save on gas*

Further reading {#further-reading}

- <https://docs.openzeppelin.com/learn/deploying-and-interacting> - OpenZeppelin
- [Deploying your contracts with Hardhat](#) - Nomic Labs

Know of a community resource that helped you? Edit this page and add it!

Related topics {#related-topics}

- [Development frameworks](#)
- [Run an Ethereum node](#)
- [Nodes-as-a-service](#)