

Blockchain Foundations and Application Development

A five-day introduction to distributed ledgers, tokenisation and smart contracts

FA-0598 · 5 days

DevOps, Cloud & Infrastructure · Software Development

Explore blockchain architecture and tokenisation, then build a small smart-contract application in a learning environment.

Introduction

This five-day course connects blockchain fundamentals and cryptocurrency mechanisms with practical smart-contract development. Participants examine network, wallet and scalability trade-offs, explore illustrative tokenisation scenarios and develop a small application using a prepared development environment.

The property example is explicitly illustrative, not a claim about the instructor's achievements or a client's project. NFT creation and transfers use test assets; token ownership alone does not establish legal ownership of property or intellectual-property rights. Advanced security and business integration are introductory reviews, not assurances of production readiness or financial return.

Learning outcomes

The course teaches participants to:

- Describe distributed ledgers, nodes, blocks and consensus mechanisms.
- Compare cryptocurrency transactions, wallet models and scalability trade-offs.
- Analyse a fictional property-tokenisation design and its technical and legal dependencies.
- Create and inspect a small NFT example using learning-only assets.
- Write and test a basic Ethereum smart contract and connect a simple DApp.
- Evaluate integration and security questions before proposing a real-world blockchain system.

Prerequisites

- Basic IT concepts and familiarity with web technologies and databases.
- Basic programming knowledge for the guided development sessions; prepared examples support newcomers.
- A compatible development environment and learning-only accounts, without real-fund requirements.

Training outline

1. Day 1 — Blockchain Fundamentals

- Distributed-ledger concepts and evolution from Bitcoin to Ethereum and other networks.
- Nodes, blocks, transactions and consensus; distinguish proof-of-work mining from Ethereum's proof of stake.
- Trust assumptions, transparency, privacy, security and common limitations.
- Compare illustrative applications with conventional database alternatives.

2. Day 2 — Cryptocurrency Mechanisms and Wallets

- Bitcoin, Ethereum and other crypto-asset network concepts.
- Wallet types, keys, transaction signing and security responsibilities.
- Transactions, network fees, consensus and scalability trade-offs.
- Compare regulatory questions by jurisdiction and date; identify issues requiring current authoritative/legal review rather than treating a general overview as legal advice.

3. Day 3 — Illustrative Tokenisation and NFTs

- A fictional property-tokenisation case: concept, design choices, risks and dependencies.
- Distinguish on-chain tokens from enforceable off-chain property or intellectual-property rights.
- NFT applications in art, collectibles and utility; standards and metadata concepts.
- Create, manage and simulate a transfer or trade of test NFTs; discuss custody, phishing and market risk without investment recommendations.

4. Day 4 — Smart Contracts and a Small DApp

- Smart-contract purpose, constraints and selected applications.
- Configure a compatible Ethereum development environment and write a small contract.
- Compile, test and deploy to a local development network or suitable testnet.

- Connect a prepared frontend and inspect transactions and errors.

5. Day 5 — Security, Integration and Project Review

- Introductory gas optimisation, contract-security review and development practices.
- Business-system integration, off-chain dependencies and governance considerations.
- Compare Ethereum, BNB Smart Chain and Cardano using selected architecture/tooling criteria rather than claims of superiority.
- Refine one bounded application for an illustrative problem; demonstrate and review limitations.
- Identify additional security, legal, operational and testing requirements before any real deployment.