

# Blockchainkurs



<https://pixabay.com/de/illustrations/tablet-steuerung-industrie-4-netz-4457913>

## Inhalt

1	Kursinhalt .....	2
2	Kursunterlagen .....	3
3	Kurssoftware .....	4
4	Links .....	5

## 1 Kursinhalt

Herzlich Willkommen zum Kurs „**Blockchain-Technologie und Smart Contract**“

### Blockchain-Grundlagen

- Funktion einer Blockchain am Beispiel Bitcoin und Ethereum
- Kauf und Aufbewahrung von Kryptowährungen
- Praxisübung: Einsatz von Hard- und Software-Wallets

### Smart Contract Programmierung

- Funktion eines EVM-Smart-Contracts
- Einführung in die Smart-Contract Programmierung mit Solidity
- Praxisübung: Smart-Contract-Programmierung für einen Verkaufsautomaten

### Blockchain-Anwendungen

- Szenario: NFTs als Grundlage für Digital-Twins und Real-World-Assets
- Szenario: Blockchain im “Supply-Chain-Management”
- Technology-Impact-Radar: Web3, Tokenisierung, Dezentrale Identitäten, Defi
- Praxisübung: dApp-Entwicklung für einen Eigentumsnachweis nach ERC-721

### Lernsituation für IT-Berufe

- Lernsituation zur Kühlkettenüberwachung

## 2 Kursunterlagen

Die Kursunterlagen befinden sich auf der Plattform „Github“ und können kostenfrei kopiert werden. Folgen Sie hierzu der folgenden Anleitung:

### (1) Git-Installation:

Bitte installieren Sie die Software „git“: <https://git-scm.com/downloads>

Git ist ein kostenloses und quelloffenes verteiltes Versionskontrollsystem, das für die schnelle und effiziente Abwicklung kleiner bis sehr großer Projekte konzipiert ist.

### (2) Erstellen Sie den Kursordner:

Bitte starten Sie eine „**Eingabeaufforderung**“ (cmd). Nutzen sie bitte **nicht Powershell**.

Ihr persönliches **Benutzerverzeichnis** sollte automatisch als Arbeitsverzeichnis voreingestellt sein.

Erstellen Sie bitte einen Ordner für diesen Kurs mit dem Namen „**blockchain**“.

```
C:\Users\Benutzername> mkdir blockchain  
C:\Users\Benutzername> cd blockchain
```

Zur besseren Übersicht werden alle Dokumente und Praxisübungen in diesem Verzeichnis abgelegt. Sie können vorhandene Datei- oder Verzeichnisnamen auch nur teilweise eingeben und mit der **Tab-Taste** vervollständigen.



### (3) Konfigurieren Sie ihre Git-Identität:

Die folgenden Befehle initialisieren die lokale git-Konfiguration.

```
C:\Users\Benutzername\blockchain> git init  
C:\Users\Benutzername\blockchain> git config --global user.email "user@example.com"  
C:\Users\Benutzername\blockchain> git config --global user.name "Local User"
```

Git verwendet den Namen und die E-Mail-Adresse, um festzulegen, wer Änderungen an einem Projekt vorgenommen hat. Diese Informationen erscheinen in den Commit-Metadaten und sind für die Nachverfolgbarkeit und Zusammenarbeit entscheidend.

### (4) Kursunterlagen kopieren:

Die folgenden Befehle kopieren die Kursunterlagen von der Plattform „github“.

```
C:\Users\Benutzername\blockchain> git clone https://github.com/StefanRosemann/blockchainkurs  
C:\Users\Benutzername\blockchain> cd blockchainkurs  
C:\Users\Benutzername\blockchain\blockchainkurs> git pull
```

Mit dem Befehl **git pull** können Sie Kursupdates abrufen.

### 3 Kurssoftware

Viele Praxisübungen des Kurses basieren auf „Node.js“.

#### (1) Softwarepaket „Node.js“:

Bitte installieren Sie das Softwarepaket „Node.js“ (Windows oder Linux)

Node.js (>= v18.18): <https://nodejs.org/en/download/>

Node.js® ist eine kostenlose, plattformübergreifende Open-Source-JavaScript-Laufzeitumgebung, mit der Entwickler Server, Webanwendungen, Befehlszeilentools und Skripte erstellen können.

#### (2) Paketmanager „yarn“:

Bitte installieren Sie den Paketmanager „yarn“:

```
C:\Users\Benutzername> npm install --global yarn
```

Yarn ist ein Paketmanager für JavaScript, ähnlich wie npm (Node Package Manager). Er wird verwendet, um Abhängigkeiten in einem Projekt zu verwalten, Pakete zu installieren und Skripte auszuführen. Yarn wurde ursprünglich von Facebook entwickelt, um einige der Schwächen von npm zu verbessern, insbesondere in Bezug auf Geschwindigkeit, Zuverlässigkeit und Konsistenz bei der Verwaltung von Abhängigkeiten.

Merkmal	Yarn	npm
Geschwindigkeit	Schneller dank paralleler Tasks	Etwas langsamer
Lock-Datei	yarn.lock	package-lock.json
Caching	Lokal gespeichert und offline verfügbar	Offline-Caching ist weniger umfassend
CLI-Befehle	Unterschiedliche Syntax	Standardmäßige Syntax

## 4 Links

“Solidity” lernen:	<a href="https://www.rareskills.io/learn-solidity">https://www.rareskills.io/learn-solidity</a>
Ethereum lernen:	<a href="https://speedrunethereum.com/">https://speedrunethereum.com/</a>
Buch “Mastering Ethereum”	<a href="https://github.com/ethereumbook/ethereumbook">https://github.com/ethereumbook/ethereumbook</a>
Faucet für Sepolia	<a href="https://cloud.google.com/application/web3/faucet/ethereum/sepolia">https://cloud.google.com/application/web3/faucet/ethereum/sepolia</a>
Ethereum-Foundation:	<a href="https://ethereum.org/de/developers/docs/intro-to-ethereum/">https://ethereum.org/de/developers/docs/intro-to-ethereum/</a>
Einführung in Ethereum:	<a href="https://docs.alchemy.com/docs/how-ethereum-transactions-work">https://docs.alchemy.com/docs/how-ethereum-transactions-work</a>