



# Introduction to IT Systems for Beginners

Introduction to Computer Science for Beginners with Fundamentals of IT Systems — from their design, through management and creation, to implementation.

A course for individuals wishing to explore the world of IT and understand its language and related concepts. An ideal introduction before starting a first job or considering a career change.

**Duration:** 2 hours x 6 meetups

**Apply:** at <https://coderbrother.pl> or directly [maciej@coderbrother.pl](mailto:maciej@coderbrother.pl)

## Topics

### Computer Hardware

- Server
- Workstation
- Operations and machine code
- Data centers

### Computer Systems

- Desktop applications
- Web applications
  - Communication between the browser and the server
- Mobile applications
  - Communication between the phone and the server

### Operating Systems

- Linux
- Windows
- Bash/Shell
- Remote desktop
- Multithreading: processes and threads
  - Task scheduling and execution
  - Multithreading issues: deadlocks, resource races, dining philosophers problem
  - Methods for solving multithreading problems
- Application operation on operating systems: from code to CPU operations

## Computer Network

- What is the internet?
- Data exchange and transmission
- Client-server and peer-to-peer models
- ISO/OSI model
- Protocols: HTTP, TCP/IP, UDP
  - HTTP and HTTP verbs
- SSH and SSL
- Domains/CNAME
- VPN
- Proxy

## Programming Languages

- Front-End Programming
  - HTML and CSS
  - JavaScript
- Back-End Programming
  - Java, .Net, Go
  - PHP
- Classification of programming languages: compiled and interpreted
- Libraries and frameworks (Spring, .Net Framework, WordPress)
- Open-source software

## Security

- Cryptography
- Encryption
- Hashing and hashing methods: MD5
- Hash functions
- Public/private keys
- SSL

## Roles in an Organization

- Task Allocation by Role
- Positions: developer, tester/QA, scrum mater, product owner, UX/UI designer, business analyst, project manager, business intelligence, DevOPS, OPS/infrastructure, support

## IT Project Development Cycles

- Management
  - Tools: JIRA, Confluence, Trello
  - Methodologies: SCRUM, Kanban, Waterfall
  - User stories

- Gathering Requirements
  - Task Prioritization
  - Road Map and Milestones
  - MVP/POC
- System Architecture
  - Application and Language Selection
  - Microservice and Monolith
  - Components of Systems
  - Flow Diagrams/UML
  - Databases (SQL, NoSQL)
    - SQL Queries
    - ACID/CAP
    - Transactionality in Databases
    - RDBMS
    - Eventual Consistency
  - Queues
  - Data exchange between services: sync/async
  - Serverless and Lambda
- Infrastructure
  - Private Cloud
  - Cloud
  - Data Center
- Software Development
  - API and API Contracts
    - Data Formats: JSON/XML/CSV
    - Restful API
    - GraphQL API
    - WebServices and CORBA/RMI
  - Version Control (GIT), GitHub/BitBucket
  - Design Patterns
  - Algorithms and Data Structures
  - DRY
  - KISS
  - SOLID
  - Circuit breakers
  - Cache Memory: Caching (Elastic Cache)
  - UX/UI - Similarities and Differences
- Software Deployment
  - Running BE on the server
  - Running FE on the server
  - CI/CD: deployment and process automation: Jenkins
  - DevOps/OPS

- Containers and Docker
  - Hosting
  - Application Servers
  - Virtual Servers
  - Web Servers
  - Application Environments: DEV/SIT/UAT/PROD
- Testing
  - Unit Tests
  - Automated Tests
  - Behavioral Tests
  - TDD/BDD/ATDD
  - Regression
  - Smoke Tests
  - Scripts and Automation Tools: JMeter, Cucumber, Gatling
  - Attacks and Security of Solutions: SQL Injection, DDoS
  - Tools: Postman
- Support
  - Application Monitoring
  - Concepts and Differences: Monitoring/Observability
  - Logs
  - Tools: Kibana/New Relic/OpenTelemetry/ELK Stack

## Cloud

- Concept
- Comparison with Data Centers and Physical Machines
- AWS
  - Concept
  - Key Elements
- Comparison of AWS with Azure/GCP

## Challenges in Software Development

- Collaboration with Developers
- Communication with Project Sponsors (the Business)
- Translating Requirements into Programming Language: Task Descriptions and Key Elements
- Impact of Requirements on System Design and Source Code
- Problem Solving from Developers' Perspective: Why Sometimes Certain Things Cannot Be Done

## Additional Topics

- SSO
- Push and Pull Models
- Relationships (One-to-Many, One-to-One, Many-to-One)