Civic Code Lab - Foundations Guide (v0.2 - 2024)

Note for New Team Members

Please consult with your manager to create a personalised learning path based on:

- Your current role requirements
- Professional development goals
- Prior experience level

The Learning Journey (The Taggart Institute: Master Your Craft) is an excellent starting point for how to self teach using resources like those below.

Note: Timeline estimates assume 5-10 hours of study per week alongside regular work duties. Progress may vary based on prior experience and available study time.

Getting Started

Useful tools

Development Environment

- Github Quickstart for GitHub Codespaces - GitHub Docs

Kubernetes (CNCF) Tools

- Skaffold Quickstart
- Minikube Hello Minikube
- Talos Linux Getting Started

Infrastructure as Code

- AWS IaC AWS | Terraform | HashiCorp Developer
- Azure IaC Azure | Terraform | HashiCorp Developer

Free Training Resources

These are our recommended resources for building core technical skillsets.

1. Cloud Native Fundamentals (4-6 weeks)

👉 **New Team Members Start Here!**

- Linux Foundation Cloud Intro Introduction to Cloud Infrastructure Technologies (LFS151) | Linux Foundation Education - Expected outcomes: Understanding of cloud native architecture and principles
- Linux Foundation Kubernetes Intro Introduction to Kubernetes (LFS158) | Linux Foundation Education - Expected outcomes: Understanding of container orchestration and basic DevOps practices

2. Cloud Platform Essentials (2-3 weeks)

AWS Resources

- AWS Cloud Practitioner Self-paced digital training on AWS - AWS Skill Builder
- AWS Security Fundamentals Self-paced digital training on AWS - AWS Skill Builder

Azure Resources

- Azure Fundamentals Course AZ-900T00-A: Microsoft Azure Fundamentals - Training
- Microsoft Security Getting Started with Microsoft Security

3. Development Paths

Understanding the OWASP Projects (Projects | OWASP Foundation) focused on Secure Development Life Cycle's (SDLC) will help in getting across the common security capabilities platforms and software should generally have. The below **Data Integration** and **Frontend Frameworks** all are typically secure by default, however the way they are deployed can significantly change their risk profile. For production or high risk environments it's always best to review the operational procedures against the 2 above standards.

Backend Development (4-6 weeks)

Expected outcomes: Basic programming skills, understanding of software design principles and secure development practices

- Python Software Design by Example
- JavaScript Software Design by Example
- **Software and code security**
  - Safestack intro: OneHourAppSec - Free Application Security Program
  - OWASP ASVS OWASP Application Security Verification Standard (ASVS) | OWASP Foundation for software executing on servers (such as Websites and API's) - default to ASVS Level 2
  - OWASP MASVS OWASP MASVS - OWASP Mobile Application Security for software executing on clients (such as Mobile Applications) - default to MAS-L2
  - Testing & QA - Scan for vulnerabilities and configuration/secret issues with Overview - Trivy
  - Testing & QA - Fast and reliable end-to-end testing for modern web apps | Playwright

Data Integration (2-3 weeks each)

Expected outcomes: Understanding how to ingest/manage data with Python and SQL and generate reports with Markdown

- SQL & Data Pipelines: Overview - SQLMesh
  - R Programming: The {targets} R package user manual (for data scientists familiar with R)
- Documentation & Reporting: Quarto

Frontend frameworks (1-2 weeks each)

The below frameworks are all focused on having simple, minimal code easy for small teams to maintain.

- Evidence Web Reports Evidence - Business Intelligence as Code - Web based reporting stack ties nicely in with sqlmesh flows all built on An in-process SQL OLAP database management system
- Capacitor UI Abstraction Building Your UI | Capacitor Documentation - Capacitor has excellent web/PWA/iOS/Android targeting support, focusing on Ionic React | Ionic Documentation likely easiest for a newcomer due to availability of docs etc.
- Streamlit in browser python prototypes stlite sharing can be used for fast prototypes in python that compiles to browser
- Zola (Zola) is a fast simple static site generator, Astro (Astro) adds lots more js/app functions but is more complex

Paid Certifications (2-3 months)

- Linux Foundation CNCF Bundle Certified Kubernetes Administrator (CKA) + Certified Kubernetes Application Developer (CKAD) + Certified Kubernetes Security Specialist (CKS) Exam Bundle - Linux Foundation - Education
- AWS Solutions Architect AWS Certified Solutions Architect – Associate Certification
- Azure Developer Associate Microsoft Certified: Azure Developer Associate - Certifications