



Platinum
Solution Partner

Technical Course Catalog

**Sparking
Your
Technical
Excellence!**



WHO ARE WE?

We are a boutique company that strives to help our customers become more efficient and effective by providing the right technical solutions, guidance, and development fire power, exactly when and where they are needed.

We specialize in the following areas:

- Back-end development – Enterprise application development and architecture - using Java, C#, Spring, Scala, NoSQL.
- Web and Front-end development and architecture - using React, Angular, Vue, Node.js, and Typescript with performance and maintainability in mind.
- DevOps – building and supporting CI/CD pipelines and Cloud, using modern technologies such as Docker, Ansible, Kubernetes, and Istio.
- Testing and automation – building automated testing systems for Web, mobile, and APIs using the latest tools and technologies.
- Engineering practices - helping engineers produce code in an Agile way while “fighting” technical debt and creating a culture of speed and excellence: TDD, Unit testing and test automation, Refactoring, Breaking monoliths.

WHY CHOOSE US?

Experience, experience, experience. Founded in 2008.

We understand Development and Operations of software. From the processes level (Agile, Lean, DevOps), to testing strategies, clean coding, emerging architectures, all the way to coding standards and automation.

We specialize in solving hard problems and provide hands-on experts to work with your teams, coach them and help them achieve their goals quickly.

We understand your needs. By getting into the picture and understanding your unique context, we can quickly provide a matching solution.

We are easy to work with! Quick and efficient administration. Speedy response time. Accurate and flexible solutions.

OUR TRAINING LEADERSHIP



Erez Tatcher
CEO, Co-Founder



Yaki Koren
PST, Head of
Engineering Practices



Moti Sofer
Head of AgileSparks
CODE



Fanny Amsterdam
Learning Development
Specialist



WHAT OUR CLIENTS HAVE TO SAY?



AgileSparks

Excellent



■ **Excellent class.**

Highly recommended. - Very thorough Very well organized. Excellent 4-day class that was full of useful material.

■ **Yaki Koren was a great instructor** and he explained the material thoroughly. I would have liked some kind of summary on the 'how to's with a systematic overview because doing it while the practical lesson is taking place is challenging.

■ **Good and comprehensive course.**

Moti was professional, knows perfectly the material. He was attentive to the needs of the people.

All main subjects and knowledge were transferred and people gave good feedback and felt they gained knowledge.

A lot of added value from Moti himself beside the material of the course.

■ **Precise Content, well delivered.**

The content was precise, well delivered. The session was very interactive with sufficient videos and quiz.

■ **It was a very nice course,** Moti was an excellent tutor.

■ **Best. Class. Ever.**

Excellent instructors, excellent conversations, excellent learning!

■ **Great experience** with a good mix of theory, practice and Q&A's.

■ **Fantastic class.**

Being all remote was a smooth experience.

■ **Knowledgeable and dedicated** to the training and everything was perfect.

■ **Very informative!** Very good teacher!

■ **Nice course,** got the basics of k8s.

■ **Great trainers!!!**

Engage and able to answer at all participants questions.

■ **Kubernetes course by Agile sparks**

Kubernetes course held by Moti was really great. It seems that Moti really controls the material extremely well. The topics in the course were exactly what I needed to get my started with Kubernetes ...

■ **The course** was very good.

OUR COURSES

(Click to get to the syllabus)

ENGINEERING PRACTICES

Design Patterns and Refactoring	05
Handling Legacy Code	06
Test-Driven Development	07
Engineering Practices Boost	08

TECHNICAL EXCELLENCE

Into GraphQL	09
React Faster- React Webapp Performance	10
NodeJS Development	11
Frontend Development with Angular	12
Frontend development with React	13
Advanced React Development	14
Docker Deep Dive	15
Deep dive into Istio on Kubernetes.	16
Kubernetes Essentials	17
Modern Provisioning Using Ansible	18
Building CI/CD with Jenkins	19
AWS Architecting	20
Scala Programming	21
Advanced Scala	22
Advanced GoLang	23
Python Intro	24
Advanced Python	25
Pro Python for DevOps and QA	26
Java Programming	27
Extreme Java for Experts	28
Akka Fundamentals	29
Microservices with Spring Cloud	30
Elasticsearch	31
Intro to Data Science	32
and Machine Learning in Python	
MongoDB	33
Elasticsearch and Kibana	34



Design Patterns and Refactoring

1 Day



Overview

In this one day course, attendees will learn about design patterns, when and how to use them and how to gradually and safely change the code toward the patterns.



Who Should Attend

Developers at all levels, who want to add these powerful tools to their toolbox.



Prerequisites

Programming experience.



Course Topics

- Principles of software design, including SOLID and Design patterns.
- Refactoring methods and how to use with IDE.
- How to change software in small, safe steps.
- Practice on existing code how to change toward patterns.
- We will cover, as time allows, Creation Methods, Factory, Strategy, Decorator, Template Method, Adapter, Null Object and Visitor.



After attending the class, attendees should be able to

See possible patterns in their code and be able to move toward it.



Handling Legacy Code

1 Day



Overview

In this one day course, attendees will stop fearing legacy code and learn how to change it so it will be more accessible and future changes will be safer and easier.



Who Should Attend

Developers at all levels, who want to add these powerful tools to their toolbox.



Prerequisites

Programming experience.



Course Topics

- What is legacy code, What is refactoring.
- Most common refactoring techniques.
- IDE refactoring.
- Adding tests to legacy code.
- Tests coverage.
- Harnessing legacy code – typical scenarios observed in many codebases.
- TDD on legacy code.
- Practice, practice, practice.



After attending the class, attendees should be able to

Identify patterns in legacy code and know how to exploit them to change the code in a safe manner and to make the required changes.



Test-Driven Development

1 Day



Overview

In this one day course, attendees will learn a different approach to software development - test driven development.



Who Should Attend

Developers, beginners and experienced.



Prerequisites

Programming experience.



Course Topics

- Emergent Design vs. traditional design.
- Test types.
- Test frameworks.
- Unit/Integration/E2E tests.
- Test-Driven Development.
- Bottom-up vs Top-down approaches.
- Practice practice practice.



After attending the class, attendees should be able to

Drive end to end development with emergent design concepts.



Engineering Practices Boost

2 Days



Overview

In this two days course, attendees will learn to develop in TDD, use design patterns and handle Legacy code. This course ramps up attendees and brings them to a different level of professionalism and provides valuable tools for developing and maintaining code.



Who should attend

Developers at all levels, who want to add these powerful tools to their toolbox.



Prerequisites

Programming experience.



After attending the class, attendees should be able to

Develop new code fast and in high quality and change existing code to be easily maintained and changed.



Course Topics

- Emergent Design vs. traditional design.
- Test types.
- Test frameworks.
- Unit/Integration/E2E tests.
- Test-Driven Development.
- Bottom-up vs Top-down approaches.
- Principles of software design, including SOLID and Design patterns.
- Refactoring methods and how to use with IDE.
- How to change software in small, safe steps.
- Practice on existing code how to change toward patterns.
- We will cover, as time allows, Creation Methods, Factory, Strategy, Decorator, Template Method, Adapter, Null Object and Visitor.
- What is legacy code.
- Adding tests to legacy code.
- Tests coverage.
- Harnessing legacy code – typical scenarios observed in many codebases.
- TDD on legacy code.
- Practice, practice, practice.



GraphQL

Into GraphQL

8 Hours

Overview

GraphQL is a relatively new method for building APIs that aims to improve on the shortcomings of REST. It was developed by Facebook and offers great flexibility, self-documentation and type safety.

However, it presents a mind shift which is not always easy to wrap one's head around. In this workshop we will learn the basics of GraphQL, both on the client side and the server side, through some presentations and lots of hands-on exercises.



Who should attend

Momo Farouq



Prerequisites

- Good knowledge of JavaScript / EcmaScript.
- Some experience with React.
- A laptop with:
 - NodeJS installed.
 - An editor of your choice. VS Code is free and it has everything that you need out of the box.



Course Topics

Introduction

- History.
- Components of a GraphQL query: schema, resolvers, query, engine.

GraphQL on the Server

- Implement a basic GraphQL server using graphql.js.
- Setup a GraphQL server using Apollo-Server.
- Querying relations.
- Making mutations.

GraphQL on the Client

- Using Apollo-Client to connect a React application with our server.
- How to set everything up
- Querying data
- Making mutations
- Render Props vs hooks
- Debugging with the dev tools



React Faster- React Webapp Performance

2 Days



Overview

The web has evolved. Today's web applications are more powerful than ever and do a lot more than just show some content. The users of the web have also evolved. They know there are alternatives and they expect an excellent experience from the services they are using.

A large part of the user experience comes from how fast a web app is. Research has shown that the performance of a web app has a direct effect over user engagement and thus over the business aspects. This is the reason most companies today take performance seriously and employ a range of techniques to keep it in shape.

In this course we will learn what we mean when we say performance, why it's important, and how we can keep it under control. We will learn many tools, techniques, and best practices, which are available for us as web developers to optimize performance in React apps, and in general.



Who Should Attend

Frontend developers who wish to learn how to improve the performance of their applications.



Prerequisites

- Basic familiarity with JS.
- Basic familiarity with the WEB world including HTTP, CSS, HTML.
- Some React experience is recommended.



Course Topics

- The meaning and theory of performance.
- Tools to help us do it right.
- Optimizing performance in all life stages of the application.
- Correctly utilizing the browser for performance.
- Performance in React applications.



NodeJS Development

4 Days



Overview

NodeJS is all about using JavaScript across the stack. With NodeJS you can use the same infrastructure/utilities for both client and server. NodeJS offers an impressive ecosystem of open source packages that can be combined to create a first-class server-side application.

During this course, we cover all the “must-know” details. We start with a brief introduction to NodeJS and then quickly step into core concepts like NPM, versioning, streams, testing, and more. At the end of the course, you will be able to develop server-side applications using NodeJS best practices.



Who Should Attend

Software developers.



Prerequisites

- Good familiarity with one of the OOP popular languages like Java or C#.
- Basic familiarity with Javascript.
- Basic familiarity with how the WEB works (HTTP, REST APIs).
- No NodeJS experience is required.



Course Topics

- Advanced Javascript.
- Architecture of NodeJS.
- Packages and core modules.
- Express and other frameworks.
- Communicating with the OS, databases and the network.
- Best practices for development and running in production.



Frontend Development with Angular

5 Days



Overview

Server-side developers (Java, .NET, and others) usually find the web platform as a mess. Some parts look quite interesting and advanced and some parts look like they were invented a decade ago.

Becoming a FED (Front End Developer) is a journey. First, we discover the power (and limitations) of the JavaScript programming language. Second, we explore browser APIs like DOM, AJAX, and HTML5 features. Last, we integrate them all into a modern client-side framework like Angular.

During this course, we will learn and deep-dive into every major concept that is a must for a serious FED. At the end of the course you will be able to develop rich client-side web applications using the latest web technologies.



Course Topics

- The three pillars of web applications: Javascript, CSS, HTML.
- How the modern web works: ES, DOM, AJAX, JSON, and other acronyms.
- Developing single-page applications with Angular.
- Under the hood with Angular.



Who Should Attend

Developers who want to get into Frontend development with Angular.



Prerequisites

- Good familiarity with one of the OOP popular languages like Java or C#.
- Basic familiarity with JS.
- Basic familiarity with WEB world including HTTP, CSS, HTML.



Frontend development with React

5 Days



Overview

Server-side developers (Java, .NET and others) usually find the web platform a mess. Some parts look quite interesting and advanced and some parts look like they were invented a decade ago.

Becoming a FED (Front End Developer) is a journey. First, we discover the power (and limitations) of the JavaScript programming language. Second, we explore browser APIs like DOM, AJAX and HTML5 features. Last, we integrate them all into a modern client-side framework like React.

During this course, we will learn and deep dive into every major concept that is a must for a serious FED that wants to create React applications. At the end of the course you will be able to develop rich client-side web applications using the latest web technologies.



Who Should Attend

Software developers who want to get into Frontend development with React.



Prerequisites

- Good familiarity with one of the OOP popular languages like Java or C#.
- Basic familiarity with Javascript.
- Basic familiarity with WEB world including HTTP, CSS, HTML.



Course Topics

- The three pillars of web applications: Javascript, CSS, HTML.
- How the modern web works: ES, DOM, AJAX, JSON, and other acronyms.
- Developing single-page applications with React.
- Under the hood with React: reconciliation and components.
- More advanced React topics: routing, state management, styling, and more.



Advanced React Development

3 Days



Overview

Developing a single or a number of components using React is easy. However, as the application grows, it becomes crucial to understand the mechanism beyond React and be able to profile it and hook into it. During this 3 day intensive course, we deep-dive into React implementation details and cover architecture details and alternatives. At the end of the course, you will be able to describe best practices, profile existing applications and optimize different aspects of the application.



Who Should Attend

Frontend developers.



Prerequisites

- At least ½ year experience with React.
- Good familiarity with JavaScript.
- Good familiarity with the WEB world including HTTP, CSS and HTML.



Course Topics

- React under the hood: understanding the rendering process.
- Advanced component patterns and architectures.
- Profiling and optimizing performance.
- Redux in depth.
- More advanced topics and tools: virtualization, SSR, styling and more.



Docker Deep Dive

2 Days



Overview

Docker is a very popular framework for building, shipping, and running distributed applications. It eases the deployment process.

In this course, the students understand the concepts, architecture, and use-cases of Docker and Docker Swarm.

The course involves extensive hands-on exercises which are crucial to the understanding of the Docker mechanics.



Who Should Attend

Developers, DevOps, System Administrators.



Prerequisites

- Basic Linux commands.
- Basic Programming Concepts (control flows, conditionals, scripting).



Course Topics

- Introduction to Docker- The problem domain, How Docker works (Linux namespaces, cgroups), Docker vs VM, Docker architecture, Installing Docker.
- Containers- Running containers (3rd party containers, Logging), Managing containers.
- Volumes- Mounting volumes to a container, Managing Volumes (Listing, Deleting), using data containers.
- Building images- Dockerfile syntax (FROM, ADD, RUN, CMD), Build configuration, Image caching, Managing Docker images (Tagging, Publishing to the registry, Deletion).
- Docker compose- Docker compose syntax (Services, Build definitions, Inheritance), Networking, Docker-compose cli api.



Istio

Deep dive into Istio on Kubernetes

2 Days



Overview

As the complexity of microservices applications grows, it becomes extremely difficult to track and manage interactions between services. Wouldn't it be better if developers could focus on writing the code for the services business logic, instead of spending their time wiring application logic to support the service mesh?

Istio to the rescue.

The course involves extensive hands-on exercises which are crucial to the understanding of the Istio mechanics.



Who Should Attend

Developers, DevOps, System Administrators.



Prerequisites

- Basic Linux commands.
- Basic Programming Concepts (control flows, conditionals, scripting).
- Docker, Kubernetes Knowledge.



Course Topics

- Introduction- Introduction to Istio as a ServiceMesh solution, Installing and configuring Istio on Kubernetes, Istio Architecture, Installing a sample application into Istio service mesh.
- Istio Traffic Management- ingress and egress Gateway, Applying Routing rules using VirtualService, Controlling traffic policy using DestinationRule, Canary Deployment using VirtualService and DestinationRule, Blue Green Deployment Using VirtualService, Deploying new Version To Production with DarkLaunch, Making sure Our App is fault Tolerant, Fault Injections (Delays, Aborts, timeouts), Handling Failures, Recover using Retries and Circuit Breakers, Accessing external services using ServiceEntry.
- Istio Security- Service Identity, Istio Secure Naming concept, Service Authentication using Peer Authentication, User Authentication using Jwt, Authorization and Access Management, Protecting the cluster with Istio egress gateway and Kubernetes network policy.
- Monitoring and Observability: Understanding Istio extensibility model, Kiali, Grafana, Prometheus, Jaeger.



kubernetes

Kubernetes Essentials

2 Days



Overview

Kubernetes is a popular framework for managing container deployments on a cluster. In this course, the students will learn the concepts of Kubernetes (e.g., Pods, ReplicaSets, Deployments, Services) as well as deployment patterns and best practices. The course includes extensive hands-on exercises will be crucial to the understanding of the topics.



Who Should Attend

Developers, DevOps, System Administrators.



Prerequisites

Basic Linux commands, Basic Programming Concepts (control flows, conditionals, scripting).



Course Topics

- Introduction to Kubernetes: Kubernetes architecture, Masters And nodes, Kubernetes components, Interacting with Kubernetes api, Declarative and Desired State, Reconciling state.
- Kubernetes Basic Workloads - Pod: An overview of images and containers, Pods, Labels & Selectors, Namespaces.
- Kubernetes Controllers: ReplicaSet, Deployment, DaemonSet, StatefulSet, CronJob.
- Networking And Services: How services work, Load balancing, Virtual service IP, Talking to services, Exposing Services, Service Endpoints, Headless services.
- Volumes: Kubernetes volumes vs. Docker volumes, Type Of Volumes, persistentVolumeClaim, StorageClass, Dynamic vs Static Volume Provisioning.
- Secrets And ConfigMaps: Managing configuration using configmaps, Exposing configmaps to containers, Configuration using Secrets, Retrieving and Decoding a Secret, Difference between configmap and secret.
- Kubernetes Security: Authenticating with TLS certificates, Authenticating with tokens, Understanding Service Accounts, Pods And Service Accounts, Users vs Service Accounts, Authorization in kubernetes Using Role-based access control, Binding a role to the service account, Roles vs Cluster Roles.
- HealthChecks: Container Probes, Liveness and Readiness.

The course includes a virtual lab for each section.



ANSIBLE

Modern Provisioning Using Ansible

3 Days



Overview

This course is designed to provide you the tools for:

- Working with Ansible.
- Understand how Ansible can be used for Deployment use cases in Amdocs..



Who Should Attend

The course is designed for DevOps, Infrastructure, or Development teams seeking to obtain a solid foundation with Ansible.



Prerequisites

- Basic Unix commands.
- Basic Programming Concepts (control flows, conditionals, scripting).



Course Topics

- Linux Fundamentals: Users and Permissions, Sudoers, Package Managers, Yum package manager, Systemd, Hands-on session.
- Inventory and ad-hoc commands: Static inventory, Dynamic inventory, Using Ansible and Using Ad-hoc Modules, Inventory variables, Host variables, Group variables, Hands-on session.
- Playbooks: Anatomy of a Playbook, YAML Syntax, Anatomy of a Task, Module and its Arguments, Testing your playbook (dry run), Popular Modules, Amdocs yum_secure module, Hands-on session.
- Advanced Playbooks: Variables, Play variables, Variable files, Extra variables, Using variables in playbooks, Basic variables, Lists, Dictionaries, Compound variables, Registering output of tasks in variables, Facts, Conditions, Loops, Handlers, Hands-on session.
- Structuring Best Practices: Structuring playbooks, Includes, Variable includes, Play includes, Task includes, Dynamic vs Static includes, Hands-on session.
- Roles: Introduction to roles, Building Simple roles, Invoking Roles from playbooks, Ansible Galaxy, Using Ansible Galaxy Roles in Playbooks, Hands-on session.
- Intro to Python: Introduction to Python, Python Packages and resources, Hands-on session.
- Custom Modules: Custom Modules, Custom Modules Parameters, Ansible Python Support for Modules, Communicating Back to Ansible, Testing custom modules, Hands-on session.



Jenkins

Building CI/CD with Jenkins

3 Days



Overview

Jenkins is an open-source and probably the most popular continuous integration tool that serves as an orchestrator for building testing and shipping a product.

On this course, we will be learning how to build test and deploy CI/CD with Jenkins 2 as an orchestration tool together with groovy as a programming language enabling us to create pipeline as a code.



Who Should Attend

Developers, System Administrators.



Prerequisites

None.



Course Topics

- Introduction to Jenkins.
- Jenkins components.
- Jenkins dashboard.
- Master and slaves.
- Job types.
- Anatomy of the Job.
- Jenkins plugins.
- Testing and Continuous Integration.
- understanding Jenkins file.
- Maven example.
- Building Jenkins pipelines using groovy.
- using Jenkins with docker.
- Archiving Artifacts.
- Email Integration.
- Slack Integration.
- Using Jenkins with Artifactory.
- Adding Sonarqube.
- Blue Ocean.
- Jenkins security.
- Troubleshooting Build Failures.
- Adding tests.



AWS Architecting

5 Days



Overview

The Cloud is probably the most important advancement in designing highly available, fault-tolerant and scalable web or data processing applications. AWS is the premiere cloud provider today. This course teaches how to utilize the services offered by AWS in order to design and implement the most efficient, cost-effective, easy to code, and easy to maintain systems that will run on AWS infrastructure.

We will cover both IASS and PASS topics teaching both how to port on-prem applications to the cloud and use AWS as an IASS provider but also how to use AWS as a PASS provider which should cut costs and investment.



Who Should Attend

Anyone with IT experience. In fact, attendees often have diverse backgrounds: Infrastructure, IT, programmers, database administrators, system administrators, DevOps people, team leaders, architects, finance administrators, managers, and more. Experience in IT is a requirement and a plus.



Prerequisites

Experience in IT in some fields is required.



Course Topics

- AWS overview.
- Types of applications: pure cloud, hybrid, on-prem.
- Principles of designing highly available, fault-tolerant, scalable systems.
- IASS services intro: regions, availability zones, best practices. Infrastructure services: VPC, subnets, security groups, nacls, IGW, Elastic IPs, ELB, Standard patterns- multiple availability zones, load balancers on entry, load balancers between, layers, separation of subnets, multiple VPCs, bastion hosts, multiple accounts.
- Identity and secure access services: IAM, users, groups, roles, best practices, interfacing other identity systems.
- EC2: machine types, AMIs, EBS, Pricing, Monitoring (Cloud-Watch), Autoscaling.
- Storage and mass data access services: S3, Glacier, Storage Gateway, Snow family, EFS, FSx, AWS Backup, Cloud Front, Security and encryption, Route53., Other offerings.
- Application services: SQS, SNS, Elastic Transcoder, Workspaces, Other offerings.
- Database services: RDS, Dynamo DB, Database Migration Service (DMS), Aurora, Elasti Cache, Redshift, Other offerings.
- High-level services: Elastic Beanstalk, OpsWorks, Cloud Formation, Other offerings, Networking services, PrivateLink, Direct Connect, Transit Gateway, Other offerings.
- Developer and DevOps services: CodeCommit, CodeBuild, CodePipeline, CodeDeploy, CodeStar, Other offerings.
- Container and Serverless services: Container Registry, EKS, ECS, Fargate, API Gateway, Lambda, How to combine with API gateway, Kinesis, S3, DynamoDb, Step Functions, Other offerings.
- Conclusions: Cloud best practices, Keeping up with AWS.



Scala Programming

3 Days



Overview

Scala is a type-safe programming language that runs on top of the JVM. Scala was tagged as the "long time replacement for Java". Scala is both object-oriented and functional, thus allowing developers to easily express themselves using powerful tools without losing performance.

By the end of the course, the students will know the Scala language, how to use it, applying correct patterns, and will also have extensive hands-on experience which is crucial when learning a new language.



Who Should Attend

Java developers, Team Leaders, Project Managers.



Prerequisites

Java Knowledge.



Course Topics

- Introduction to Scala: Why another Language?, The Features of Scala, Scala's Extensibility, Scala vs. Java.
- Basic Syntax (for Java Developers): Basic syntax compared to Java, Basic Class syntax, Visibility Rules.
- Object-Oriented Programming in Scala: Everything is an Object, Objects and Companion Objects, No Operators (just Functions), Introducing Parameterized Types, Traits & Mixins.
- Functions: High Order Functions, Lambdas and Closures.
Collections: Mutable vs. Immutable Collections, Comparing and Integrating with Java Collections, Functional API, Persisted Collections, Sequences.
- Pattern Matching: Case Classes, Pattern Matching, Exception Handling.
- For Expressions.
- Implicits: Implicit methods, Implicit classes, Implicit parameters.
- More Topics: Tail Recursion, Futures, and Promises.
- Testing Frameworks.
- Introduction to Functional Programming: What is Functional Programming, Abstractions, Overview Monoids, Functors, and Monads.



Advanced Scala

4 Days



Overview

Scala is a type-safe programming language that runs on top of the JVM. Scala is tagged as the “long time replacement for Java”. Scala is both object-oriented and functional, thus allowing developers to easily express themselves using powerful tools without losing performance.

In this course, we'll focus on the advanced syntax features that Scala offers. We'll also dive into Functional Programming idioms and how to implement them in Scala.



Who Should Attend

Scala developers, Team Leaders.



Prerequisites

Good Scala Knowledge.



Course Topics

- Recap: For Comprehensions, Implicits, Parameterized Types and Variance. Functional Abstractions: The TypeClass Pattern, Equality, Monoids, Functors + Composition, Lambda Types, Kind Projector Plugin, Monads.
- Immutability: The Benefits of Immutability, Introducing Lenses , Lenses Composition, Partial Lenses, Monocle (3rd Party), The Global Variable Problem, Introducing the State Patten, State as a Monad, State & Lenses.
- Effects: Problems with Effects and FP, Monads Composition Problems, Monad Transformers, Trampolines and Recursion, The Free Monad, Extensible Effects, Eff (3rd party).
- Generic Programming: Introducing Shapeless, Working with HList, Generic[_], CoProducts, Typeclass Derivation.
- Effects: The Problem with Multiple Effects, Overview of MonadTransformers, The Free Monad, Introducing Eff, Composing Effects.
- The Scala Ecosystem: Typelevel Projects, Cats, Testing Frameworks, Writing FP Micro-Services.



Advanced GoLang

3 Days



Overview

This advanced GoLang course is project-based: building a shared "game" server that students can use to interact with each other via defined APIs. There is also a strong "code reading" portion, whereby students can check their understanding of Go by reading and modifying Opensource Go Projects. This course will also explore concurrency (by diving into more concurrency patterns) and the standard library, as well as provide coding activities using popular Go libraries/frameworks (such as logging and microservices).



Prerequisites

Skill level:

- Familiarity with Docker.
- Basic Familiarity with Go.
- Experience with Kubernetes in a Production environment.
- Nice to have: Managed DB & servers.

The course requires minimum machine configuration and setting (will be provided separately).



Course Topics

- Why Go?
- Setting up Go environment: \$GOPATH & \$PATH.
- Go Syntax overview (Refresher). Flow control, switch statements, ranges, array, slices...
- Introduction to Goroutines: sync.waitgroup, GOMAXPROCS, Go scheduler.
- Go runtime: Garbage collector, understanding Go's Scheduler.
- Channels and select statement.
- Go build and Go run: Binaries (GOOS & GOARCH / CGO_ENABLED).
- Object oriented using Go.
- xUnit style tests using built-in testing package.
- Using testify/assert: Packaging & imports.
- Directory layout: Working with third-party packages - Go modules, higher order functions.
- Context package
- os: Awesome 3rd party packages (flag, validator, viper, etc.)
- Building web services in Golang.
- Using gorilla/mux for ReSTful APIs: Performance tuning, instrumentation & analysis.
- Profiling.
- Tracing: HTTP tracing.
- Escape analysis: go build -gcflags "-m -m", understanding stack and heap memory, interop.
- Package plugin.
- Writing your own K8s operator in Go using Operator Framework.



Python Intro

3 Days



Overview

Be able to create python 3 programs which

- Parse & process text
- Interact with the user and the local system

Create well-structured, reusable and easy to maintain programs by

- Being aware of python coding standards and idioms
- Including automated testing in your programs
- Adopting pythonic project structure



Who Should Attend

Developers, QA & Devops with previous programming experience



Environment

Python 3 on Windows, Linux or mac
A working python installation



Course Topics

- Hello world - the anatomy of a simple python program
- Running python scripts: Command-line, repl, jupyter
- Variables and Data types



Course Topics

- Control structures (conditions, loops)
- Functions & scopes
- Importing packages and files
- Parsing text with python
- Lists and tuples
- Dictionaries
- Exceptions - Handling
- Exceptions - Generating
- The python ecosystem
 - Editors & Development environments
 - Python style: PEP8 & linters
 - Python vs. other languages, when to use Python
 - Installing and managing packages using pip
 - Using virtual environments
 - Python versions
 - Installing packages using pip
 - Python at scale: Structuring a project
- Interacting with the user
 - Parsing command-line arguments
 - Reading input
 - Formatting output
 - Logging
- Interacting with the OS
 - Working with date and time
 - Reading and writing files. Text and CSV
- Interacting with other systems
 - Getting data from the network using the requests library
- Object Oriented python (overview)



Advanced Python

3 Days



Overview

Take your Python skills to the next level
Learn about the new and exciting features of Python 3.9



Who Should Attend

Developers, QA & Devops with previous programming experience



Environment

Python 3 on Windows, Linux or mac
A working python installation



Course Topics

This course contains a mix of advanced topics. It can be customized to include and exclude any of those

- Object Oriented Python
- TDD & BDD with python. Using pytest & pytest.BDD
- Creating web services (Flask, FastAPI, jinja2)
- Functional programming using comprehensions & lambdas, Iterators and generators, itertools
- Asynchronous python - Threads and multiprocessing, async/await
- Python and the cloud. AWS S3, firebase & google storage
- Intro to Data Science: Jupyter, numpy and pandas. Matplotlib
- Overview of Machine Learning with PyTorch. The basic ML workflow
- Working with databases (using sqlite, ORMs, mongodb)
- Embedded Python - Python on arduino & raspberry pi (ESP-32 / raspberry pi/zero/pico)
- Building extensions to python (SWIG)
- Building larger python projects (pyinstaller, packages, docker)
- Web scraping using BeautifulSoup & MechanicalSoup
- Creating desktop apps using TkInter, PyQt
- Advanced scripting (interfacing with the OS, task management, system monitoring, command-line arguments...)



Pro Python for DevOps and QA

2 Days



Overview

This course will help raise the bar from “hacking” with python, to professional-level code you and your peers can maintain.



Who Should Attend

DevOps and QA people with basic knowledge of python. Software developers who want to polish their knowledge of system-level programming, and Automated testing using Python.



Prerequisites

Working knowledge of Python (2 or 3).



Course Topics

- Setting up Python: Packaging and Isolation using pip & virtualenv, Editing python. Visual Studio Code, pylint, Debugging using VSC, Coding style and PEP-8, Documenting python code using pydoc and doctest.
- Writing Tests: Automating tests and the test pyramid, Writing our first test, The Setup / Execute / Verify / Teardown pattern, Measuring tests coverage.
- Using pytest: pytest vs unittest, Test discovery & Structuring unit tests, Selective running of test using marks, Testing exceptions, Python project structure.
- TDD - Test Driven Development: The Fail-Implement-Pass-Refactor Cycle, Writing clean tests using pytest Fixtures.
- Isolating system dependencies: Dependency injection, Mocking, Stubbing, Patching.
- ATDD/BDD: Writing acceptance criteria, Definition of Done, Code Reviews, Gherkin, Automating gherkin test specs using a BDD framework pytest.BDD, Cross-framework tests coverage.



Java Programming

5 Days



Overview

Java is one of the most popular programming languages. It is an Object-Oriented, portable, fast, and rich language. Java provides many services out of the box (e.g, memory management, security) which helps developers be more productive. The course is intended for anyone who wants to enter the 'Java World'. The course includes many examples and hands-on exercises through which the material is demonstrated and practiced. The course is based on Java 13.



Who Should Attend

Developers that want to learn the Java language.



Prerequisites

Familiarity with Object-Oriented concepts.



Course Topics

- Introduction to Java.
- Basic Syntax.
- Expressions.
- Using Arrays.
- Java Classes.
- Basic Design Patterns.
- Exceptions & Assertions.
- IDE, Classpath & JARs.
- Working with Common Classes.
- The Java Collection Framework & Generics.
- Java IO.
- Multi-Threading.
- Networking.



Extreme Java for Experts

3 Days



Overview

Although Java is considered a relatively easy to use language, it has many sophisticated mechanisms and delicate points that are in many cases not fully utilized or even known to developers. Proper use of these mechanisms and "under the hood" structures greatly enhances code optimization and fine-tuning.

This advanced course focuses on these fine points and internal mechanisms and provides the "behind the scene" understanding of core Java libraries and the JVM internals. The knowledge gained in the course will significantly improve developers' ability to write more efficient and robust code.

The course includes many examples and hands-on exercises through which the material is demonstrated and practiced.



Who Should Attend

JavaSE/EE developers, team leaders, and Architects.



Prerequisites

2 years experience in Java Programming.



Course Topics

- Multi-Threading and the JMM: Introduction, The Java Memory Model, Core Java threading, functionality, Advanced Synchronization Mechanisms (java.util.concurrent), Fork/Join, Working with CompletableFuture, Third-party Concurrency Models (Akka vs. Vertx.io).
- Memory Management & GC: Introduction, The GC Anatomy and Algorithms.
- Reference Objects, Buffers & Native Memory.
- Generics, Streams & Collections: Understanding generics in and out, Deep Dive into Lambdas, Stream API, Overview on Third-Party Collections.
- Classloading & Agents: Understanding Class Loaders in Java, Agents, and Instrumentation.
- Performance & Monitoring: The JIT compiler and the HotSpot JVM, String Handling, Exceptions, JVM Tools (VisualVM & JMC, jstat & jmap), Analyzing Heap Dumps, Profiling Java Applications.



Akka Fundamentals

2 days



Overview

Writing multi-threaded code is hard. Akka is an open-sourced framework that enables you to easily and safely develop concurrent, distributed and fault-tolerant applications in a lightweight manner. This 2-day course includes extensive hands-on for a deep understanding of the material.



Who Should Attend

Java developers, team leaders and project managers.



Prerequisites

Familiarity with the Java language.



Course Topics

- Introduction to Akka: The Java Memory Model, The Problem in Writing Concurrent Applications, Understanding the Actor Model, The ActorSystem, Creating Actors, The ActorRef, Sending Messages, Immutability.
- Flows: Waiting for Responses, Futures, Composing Futures, Actor Lifecycle, Timeouts, Forwarding, Routers, FSM.
- Supervisor Hierarchies: Creating Child Actors, Supervisor Strategies, Actor Paths, Exception Handling and Failures, DeathWatch.
- Threading Models: Mailbox Types, Dispatcher Types, Priorities.
- Advanced Concepts: Actors with Stash, 'Becoming' (State Pattern), Scheduling, Persistence.
- Events: The EventStream, Handling Logging, Creating Events, Classifiers.
- Akka Typed overview, Protocols, Supervision & Persistence revisited, Transactors and STM, Testing typed behavior.
- Clustering: Remote Actors, Serialization Protocols, Clustering, Distributed Pub-Sub.



Spring Cloud

Microservices with Spring Cloud

2 Days



Overview

Spring Framework is a popular and effective open-source framework for Enterprise Java developers. Spring provides the infrastructure for modular and maintainable application development. Spring Cloud builds on Spring and Spring Boot to provide essential infrastructure and common pattern solutions for Microservices and Cloud-based applications. This course covers Microservices and REST and provides hands-on experience with Spring Boot and Spring Cloud.



Who Should Attend

Java developers who want to use Spring Microservices application development, Java developers who want to leverage the readymade solutions of Spring Cloud.



Prerequisites

Experience with Java Development, Experience with Developing Spring Applications.



Course Topics

- Spring DI and Boot overview.
- Microservices Overview: Intro, Advantages, Challenges, and Costs.
- Spring Cloud: Centralized Configuration, Service Discovery (Eureka), Circuit-Breakers (Hystrix), Load Balancing (Ribbon), Gateway (Zuul), Declarative Clients (Feign).



elasticsearch

Elasticsearch

3 Days



Overview

The purpose of this course is to provide knowledge of how to work with and operate Elasticsearch. We will have an overview of Elasticsearch and the Elastic stack, learn how to index/query data, how to model data, and how modeling affects performance. We will also learn about Elasticsearch's internals and how to tailor a cluster to fit your specific requirements.



Who Should Attend

Developers / Tech Leaders / DevOps Engineers



Prerequisites

Basic understanding of HTTP, JSON, and Databases.



Course Topics

- Introduction: Elasticsearch, Elastic stack.
- First Steps:
 - Installing Elasticsearch & Kibana locally.
 - Running Elasticsearch & Kibana on Docker.
 - Architecture.
 - Sharding.
 - Replication.
 - First queries.
 - Curl.
 - Postman.

- Working with Documents.
 - Creating & deleting indices.
 - Indexing.
 - Fetching queries by ID.
 - Updating documents.
 - Deleting documents.
 - How Elasticsearch reads/writes data.
 - Document versioning.
 - Batch processing.
- Mapping
 - Inverted indices & lucene.
 - Data types.
 - Explicit mapping.
 - Updating existing mapping.
 - The reindex API.
 - Dynamic mapping.
 - Top words & stemming.
 - Analyzers.
- Searching Basics.
 - URI searching.
 - Query DSL.
 - Understanding the search mechanism.
 - Reading results.
 - Query context.
 - Full text / term-level queries.
- Searching Advanced .
 - Term level queries in depth.
 - Full text queries in depth.
 - Boolean queries.
 - Queries join.
 - Controlling results, aggregations.
- Optional - Cluster sizing, understanding the requirements, single shared testing, Node saturation, cluster sizing.

Intro to Data Science and Machine Learning in Python

3 Days



Overview

Data science and machine learning are becoming increasingly mainstream in software products. As a developer, if you want to get into that playing field, come and get a taste of the skills needed, and start your journey on that path. During this training you will:

- Analyze, visualize and derive insights from data using python
- Create a basic, end-to-end, machine learning model
- Understand the full workflow of Data Science and Machine Learning, from exploration to production

The course is hands-on and the participants are expected to take part in coding throughout the course



Who Should Attend

Software Developers



Prerequisites

Python programming experience

No programming experience with Python? Consider taking our Python intro course



Course Topics

- The data exploration cycle
- Jupyter notebook & Google colab
- Numpy. Working with arrays
- Pandas. Working with dataframes
- Data import & cleaning, filling the gaps
- Hands on: analyzing weather data
- Basic visualization with Matplotlib
- AI, Machine Learning and Deep Learning concepts
- The ML Workflow
- Regression systems
- Backpropagation & Gradient Descent
- Pytorch. Working with tensors
- Features engineering. Scaling and one-hot encoding
- Artificial Neural networks
- Hands on: Working through an end-to-end machine learning flow
- Overview of common architectures - ANNs, CNN, RNN, GANs
- From ML to production - Overview of the ML ecosystem



MongoDB

3 Days



Overview

MongoDB training introduces a NoSQL database that is cross-platform, document-oriented, that provides high performance, high availability, and easy scalability.

In the course, we will Understand MongoDB and NoSQL domain, create MongoDB clusters for different kinds of applications, understand MongoDB Architecture, design and model Applications for MongoDB, and cover basic operations, monitoring and troubleshooting.



Who Should Attend

Developers, Architects, DBAs



Prerequisites

Familiarity with Linux command line, minimal knowledge of a modern programming language, database, or data-warehouse concepts



Course Topics

NoSQL

MongoDB Concepts

- Overview
- Advantages
- Environment
- Data Modelling
- Create and drop Databases
- Create and drop Collections
- Datatypes
- Insert, query, update and delete documents
- Projections
- Limit Records
- Sort Records
- Indexing
- Aggregation
- Replication

Operation

- Sharding
- Deployments
- Backups

Advanced topic

- Relationships
- Database References
- Covered Queries
- Analyzing Queries
- Atomic Operations
- Advanced Indexing
- Indexing Limitations
- Text Search



Elasticsearch and Kibana

4 Days



Overview

This Elasticsearch training intends to provide a solid foundation in search and information retrieval. It starts with fundamental concepts and follows with internals, best practices, and key features. Each topic is followed by a hands-on lab. At the end of the training, the attendee will have a deep understanding of how Elasticsearch works, will be able to reliably analyze data, and will be ready to build search applications and present visualization using Kibana



Who Should Attend

- Architects
- DBAs
- BI developers and analysts



Prerequisites

Basic knowledge of database concepts and data presentation tools



Course Topics

Upon completion of the course, participants will be able to:

- Describe and design an ELK environment
- Create Elasticsearch cluster
- Control and monitor the Elasticsearch cluster
- Use Kibana for visualizationscosystem



Course Topics

DAY 1

Module 1 - Introduction to the Elastic Stack

- Elasticsearch
- Kibana
- Logstash
- Beats
- Security
- Alerting
- Monitoring

Module 2 - Introduction to Elasticsearch

- The Story of Elasticsearch
- Documents
- Indexes
- Indexing Data
- Searching Data

Module 3 - The Search API

- Introduction to the Search API
- URI Searches
- Request Body Searches
- The match Query
- The match phrase Query
- The range Query
- The bool Query
- Source Filtering

Continue...



elasticsearch



kibana

Elasticsearch and Kibana

Continued

4 Days



Course Topics - Continued

DAY 2

Module 4 - Text Analysis

- What is Analysis?
- Building an Inverted Index
- Analyzers
- Custom Analyzers
- Character Filters
- Tokenizers
- Token Filters
- Defining Analyzers
- Synonyms
- How to Choose an Analyzer
- Segments

Module 5 - More Search Features

- Filters
- Term Filters
- The match_phrase_prefix Query
- The multi_match_Query
- Fuzziness
- Highlighting
- More Like This

DAY 3

Module 6 - Working with Search Results

- The Anatomy of a Search
- Relevance
- Boosting Relevance
- DFS Query-then-fetch
- Sorting Results
- Doc Values and Fielddata
- Pagination
- Scroll Searches
- Choosing a Search Type

DAY 4

Module 7 - Aggregations

- What are Aggregations
- Types of Aggregations
- Buckets and Metrics
- Common Metrics Aggregations
- The range Aggregation
- The date_range Aggregation
- The terms Aggregation
- Nesting Buckets

Module 8 - More Aggregations

- Global Aggregation
- The missing Aggregation
- Histograms
- Date Histograms
- Percentiles
- Top Hits
- Significant Terms
- Sorting Buckets



AgileSparks

Challenging your comfort zone

Technical Course Catalog

USA

Email: usa@agilesparks.com
Phone: +1-617-564-0455

Europe

Email: academy@agilesparks.com
Phone: +972-53-222-1473

India

Email: asi@agilesparks.com
Phone: +91-98237-87888