Advanced JavaScript

This JavaScript Coding Course will teach you everything you need to master the most widely used front-end language today.

- Covering all the new Object-Oriented features introduced in ES6, this course shows you how to build large-scale web apps
- Build apps that promote scalability, maintainability, and reusability
- Learn popular Object-Oriented programming (OOP) principles and design patterns to build robust apps
- Implement Object-Oriented concepts in a wide range of front-end architectures
- Master JavaScript's OOP features, including the one's provided by ES6 specification
- Identify and apply the most common design patterns such as Singleton, Factory, Observer, Model-View-Controller, and Mediator Patterns
- Understand the SOLID principles and their benefits
- Use the acquired OOP knowledge to build robust and maintainable code
- Design applications using a modular architecture based on SOLID principles

"Any application that can be written in JavaScript will eventually be written in JavaScript."
—Atwood's Law, by Jeff Atwood

Prerequisites

This is an advanced course. You should be at the level of HTML5 / JavaScript programming before you do this course

Intended Audience:

Anybody who wants to understand and remember how to program with JavaScript using the best techniques and the most recent

standards, including OO.

Further Training:

Angular

NodeJS.

Course Material

Provided

Course Contents

DAY 1

A Refresher of Objects

- Object literals
- Object constructors
- Object prototypes
- Using classes

OOP Principles

- OOP principles
- Is JavaScript Object Oriented?
- Abstraction and modeling support
- OOP principles support
- JavaScript OOP versus classical OOP

Working with Encapsulation and Information Hiding

- Encapsulation and information hiding
- Convention-based approach
- Privacy levels using closure
- A meta-closure approach
- Property descriptors
- Information hiding in ES6 classes

DAY 2

Inheriting and Creating Mixins

- Why inheritance?
- Objects and prototypes
- ES6 inheritance
- Controlling inheritance
- Implementing multiple inheritance
- Creating and using mixins

Defining Contracts with Duck Typing

- Managing dynamic typing
- Contracts and interfaces
- Duck typing
- Duck typing and polymorphism

Advanced Object Creation

- Creating objects
- Design patterns and object creation
- Creating a singleton
- An object factory
- The builder pattern
- Comparing factory and builder patterns
- Recycling objects with an object pool

DAY 3

Presenting Data to the User

- Managing user interfaces
- Presentation patterns
- The Model-View-Controller pattern
- The Model-View-Presenter pattern
- The Model-View-ViewModel pattern
- A MV* pattern comparison

Data Binding

- What is data binding?
- Implementing data binding
- The publish/subscribe pattern
- Using proxies

DAY 4

Asynchronous Programming and Promises

- Is JavaScript asynchronous?
- Writing asynchronous code
- Introducing Promises
- Using Generators

Organizing Code

- The global scope
- Creating namespaces
- The module pattern
- Module loading
- ECMAScript 6 modules

DAY 5

SOLID Principles

- Principle of OOP design
- The Single Responsibility Principle
- The Open/Closed Principle
- The Liskov Substitution Principle
- The Interface Segregation Principle
- The Dependency Inversion Principle

Modern Application Architectures

- From scripts to applications
- From old-style to Single Page Applications
- The Zakas/Osmani architecture
- Cross-cutting features and AOP
- Isomorphic applications

Duration and pricing

Pricing Group A

Certificate

Read about our certificates

Bookings

You can download the course registration form on our home page or by clicking here

Brochure

You may download a pdf copy of this page by clicking on the pdf icon at the top of the page.

Questions

Please email us

Schedule

On the calendar below. If your browser doesn’t display the calendar below, please click on this link or try using Google Chrome, alternatively please enquire via our Contact Us page.