Updated on 27/06/2025

Sign up

Anime.JS Training

3 days (21 hours)

Overview

Anime.js is a lightweight JavaScript library dedicated to the animation of DOM, SVG and JavaScript objects. With this comprehensive training course, you'll learn how to design fluid, expressive and high-performance animations to enrich your web interfaces and deliver dynamic, modern user experiences.

You'll be introduced to the basics of Anime.js, from installation to the first CSS animations, including the manipulation of properties, JS objects and SVG shapes. Each concept is accompanied by concrete examples to get you up to speed quickly.

You'll explore timelines, keyframes, custom easings and complex trajectories to create rich, synchronized animated sequences. Practical workshops will help you structure your animations in a modular and coherent way.

Concrete use cases will enable you to apply the skills you've learned to real-life projects, while complying with performance and accessibility best practices.

On the training date, we will present you with the latest version of the Anime.JS software.

Objectives

- Master the fundamentals of the Anime.js library and how it works with the DOM, JavaScript objects and SVG elements
- Create fluid, sequential or synchronized web animations using timelines, keyframes and motion paths
- Understand and manipulate animation curves to enhance visual impact and user experience
- Structure complex animations in an interactive web project, respecting best practices accessibility and performance

- Integrate Anime.js into modern environments (React, Vue, Angular) and implement modular, maintainable animations.

Target audience

- Web and video game developers
- Front-end developers

Prerequisites

- Basic knowledge of HTML5 and JavaScript

Our Anime.JS training program

Introduction to Anime.js

- What is Anime.js?
- Advantages over CSS / jQuery / GSAP
- Common use cases: loader, site intro, UX interactions
- Integration via CDN vs. npm
- Setting up a basic project (HTML / JS / Anime.js)
- First tests with anime({})

Animation curves

- targets: CSS selectors, arrays, JS objects
- Dynamic manipulation
- CSS: translate, rotate, scale, opacity, etc.
- Colors: RGB, HEX, HSL, with fluid interpolation
- JavaScript objects: numerical properties
- Basic properties: duration, delay, easing, loop
- Changing several properties in one animation
- Callbacks: begin, update, complete

Creating and managing animations

- Notions of linear, easeInOutQuad, spring
- Choosing the right curve for the desired effect
- Using easings.net
- Simulating different effects with anime.js

Timelines

- Building a timeline
- Adding stages with .add()
- Managing relative timing
- Animating several elements together
- Logical organization of a complex animation

Keyframes and dynamic effects

- Definition: array of intermediate steps
- Application to one or more properties
- bounce, zoom + rotation
- Cutting text letter by letter
- Animate each letter separately

Program control

- play(), pause(), restart(), reverse()
- Example of button control
- Analyze and manipulate active animations
- Reacting to an animation in progress in the application

SVG animation and motion path

- Targeting SVG attributes: d, strokeDasharray, strokeDashoffset
- Stroke effects
- Using anime.path()
- Moving an object on an SVG path
- Managing path-oriented rotation

Practical workshops

- Animated SVG circle
- Animated text or logo
- Title with word-by-word appearance
- Background with smooth transition
- Buttons with animated hover
- Mobile menu open/close

Integration with other frameworks

- Lifecycle usage
- Optimization best practices
- Example combined with WebGL

Best practices & performance

- Minimizing the number of active animations
- Scroll triggering
- Disabling animations for certain users
- User feedback vs. decorative animation

Companies concerned

This course is aimed at both individuals and companies, large or small, wishing to train their teams in a new advanced computer technology, or to acquire specific business knowledge or modern methods.

Positioning at training start

Positioning at the start of training complies with Qualiopi quality criteria. As soon as registration is finalized, the learner receives a self-assessment questionnaire which enables us to assess his or her estimated level on different types of technology, as well as his or her expectations and personal objectives with regard to the training to come, within the limits imposed by the selected format. This questionnaire also enables us to anticipate any connection or security difficulties within the company (intra-company or virtual classroom) which could be problematic for the follow-up and smooth running of the training session.

Teaching methods

Practical training: 60% hands-on, 40% theory. Training material distributed in digital format to all participants.

Organization

The course alternates theoretical input from the trainer, supported by examples, with brainstorming sessions and group work.

Validation

At the end of the session, a multiple-choice questionnaire verifies the correct acquisition of skills.

Certification

A certificate will be awarded to each trainee who has completed the entire course.