

ThreeJS training with WebGL

5 days (35 hours)

Presentation

[Angular](#) is the benchmark typescript framework for well-structured projects, benefiting from all the framework's tools (ngTools, Angular Cli). Along with React, it is one of two web frameworks developed by GAFAM and widely used and maintained by the open source community.

For its part, Three JS is a JavaScript library (also available in typescript) that enables the creation of 3D scenes in a Web browser. This library is widely used in the WebGL world to create interactive websites (Sketchfab, [Orano Innovation](#)), but also for industrial products (3D simulator in SAAS access, virtual navigation in digital building clones).

In this training course, you'll discover how to create a Sketchfab yourself, enabling you to visualize and navigate in a 3D scene. You'll learn the fundamentals of 3D with ThreeJS (scene graph, mesh, materials, vertices, facets, camera matrices, view frustum). These fundamentals will be useful for numerous 3D libraries and SDKs, including [Matterport](#) for creating digital clones, for example.

Finally, you'll discover an Open Source software framework for CAD/CAM, CAE and 3D modeling: OpenCascade. We'll open with two current examples of Opencascade in use on the Web: Node-Occ (Node Native Recompilation) and OpencascadeJS (WebAssembly).

Objectives

- Discover ThreeJS features
- How to develop a new Web application with Angular and Three JS
- Designing Angular applications with JavaScript and TypeScript 4.8
- Build a scene graph including objects, materials and a camera
- How to define the positions and rotations of scene elements and camera characteristics

- Differentiate between the advantages/disadvantages of rendering the scene in the browser vs. server-side rendering and use cases

Target audience

- Web Developer
- Lead Developer
- Architect
- Project Manager

Prerequisites

Knowledge of JavaScript or TypeScript, HTML5/CSS3, Angular

ThreeJS with WebGL training program

Day 1 - Introduction

- **WebGL and ThreeJS:** its history
- TypeScript: typing JavaScript
- ECMAScript, DOM and BOM
- Canvas support

WebGL basics (Fundamentals I)

- A 3D Object and its definition
- Scene definition
- Update coordinates (position, size, rotation)
- The different types of cameras
- Testing examples: online editing on JSFiddle

Angular basics

- Integrating Angular
 - Components, modules, services
- Property binding, event binding
- The forms
- The different types of directives

- The router and route hierarchy (Example with ui-router and lazy loading)
- Testing examples: online editing on JSFiddle

Day 2 - WebGL basics (Fundamentals II)

- Meshes and geometries
- Different types of materials and illuminations
- Exporting geometries or scenes
- Collision detection
- Testing examples: online editing on JSFiddle

Day 3 - Creating a SKETCHFAB (Day 1)

- Preparing the Angular application shell that hosts the canvas
- Set up interfaces (import, export, change camera, center scene on object)
- Binding the canvas with the ThreeJS library
- Write the loading of the various elements (camera, vertices, faces, lights, materials) in the 3D scene

Day 4 - Creating a SKETCHFAB (Day 2)

- Writing the 3D object loader in ThreeJS
- Setting the scene
- Binding Angular interfaces with API calls in ThreeJS
- Group trial and demonstration
- Import and visualize various 3D models

Day 5 - Introducing OpenCascade

- A look at two different geometric solid models (BREP, CSG) and the advantages and differences of each.
- Introducing OpenCascade and its OpenSource version Node-Occ
- Example of CSG server-side rendering and benefits
- Comparison with a WebAssembly version of OpenCascade

Companies concerned

This training 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 on entry to training

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 of proficiency in different types of technology, as well as his or her expectations and personal objectives for 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 course: 60% Practical, 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.

Sanction

A certificate will be issued to each trainee who completes the course.