



Curso JavaScript Essentials 2





CAS Training



Duración
50 horas



Modalidad
Aula Virtual



**Learning by
doing**



**Curso
Oficial**

Acerca de:

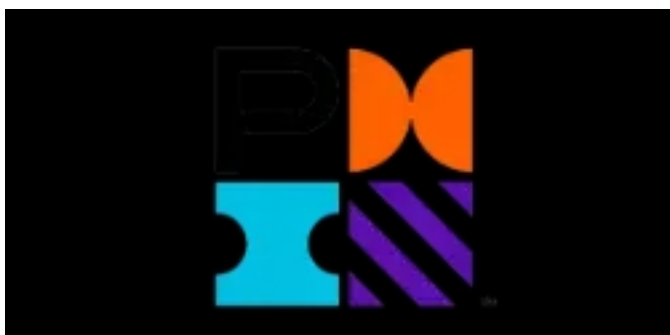
Consulta nuestro calendario Agenda tu examen de certificación

Encuentra tus cursos y certificaciones oficiales

SearchSearch



Certificación ITIL® 4 Foundation





Objetivos:

- Diseñar, desarrollar, depurar, ejecutar y refactorizar programas informáticos escritos en JavaScript.
- Aprender sobre la programación orientada a objetos (OOP).
- Analizar y modelar problemas de la vida real y resolverlos utilizando los enfoques procedimental y OOP.
- Aprender sobre arquitectura de soluciones.
- Desarrollar un pensamiento algorítmico y analítico.
- Conocer las mejores prácticas en la programación.
- Crear, desarrollar y mejorar tu propia cartera de programación para aumentar tu valor en el mercado laboral.
- Prepararte para el examen de certificación JSA – Certified Associate JavaScript Programmer.

Requisitos:

- No existen requisitos previos para tomar este curso.
- Sin embargo, es recomendable haber obtenido la certificación JSE – Certified Entry-Level JavaScript Programmer.

Material del curso:

Documentación oficial para el **curso JavaScript Essentials 2**.

Perfil del docente:

- Formador certificado por OpenEDG.
- Más de 5 años de experiencia profesional.
- Más de 4 años de experiencia docente.
- Profesional activo en empresas del sector IT.

Metodología:

- “Learning by doing” se centra en un contexto real y concreto, buscando un aprendizaje en equipo para la resolución de problemas en el sector empresarial.
- Aulas con grupos reducidos para que el profesional adquiriera la mejor atención por parte de nuestros instructores profesionales.
- El programa de estudios como partners oficiales es confeccionado por nuestro equipo de formación y revisado por las marcas de referencia en el sector.
- La impartición de las clases podrá ser realizada tanto en modalidad Presencial como Virtual.



Examen y Certificación:

Preparación para el examen de certificación:

Examen de certificación JSA – Certified Associate JavaScript Programmer



Contenidos:

Módulo 1: Classless objects

- Creating individual objects (basic way to create object using literals)
- Object properties (adding, modifying and deleting properties, nested properties)
- Dot notation vs. bracket notation (dot notation as the primary way to refer to object fields, use of bracket notation to allow the use of multi-word and computed keys)
- Property existence test and property enumeration (test for the presence of a field in an object, using the for ... in to pass the keys of an object, the Object.key method as an alternative to for ... in)
- Comparing objects (the idea of object references, reference comparison vs. field comparison - deep comparison)
- Coping objects (coping references, cloning, merging - Object.assign method; spread operator and shallow cloning, deep cloning concept)
- Methods (function as an object property, define methods in the object body and add methods to existing objects, using this keyword inside of methods)
- Getters and setters (methods treated as properties - defining and using getters and setters)
- Property and object configuration (modifying attributes of objects and fields; using methods Object.defineProperty, Object.preventExtensions, Object.seal and Object.freeze)
- Different ways to create classless objects (factories, constructor function and new operator, Object.create method)
- Prototypes (prototype-based inheritance, object property __proto__, constructor function prototype property, use of setPrototypeOf method)

Módulo 2: Classes and class-based approach

- Class declaration (normal class declaration - keyword class; class body - constructor, properties, methods; class as first-class citizen - storing classes in variables and class expression)
- Creating objects from a class (creation of an object based on a class; looking for a connection between an object and a source class - instanceof operator)
- Properties (properties definition inside of class methods - constructor and regular methods; direct declaration inside of the class body)
- Getters and setters (defining and using getters and setters)
- Inheritance (class inheritance - extends keyword; shadowing methods and properties; using the super keyword in the constructor and in methods)
- Static members (the concept of static members; methods and properties associated with a class instead of an object - defining and using)
- Classes vs constructors (similarity of classes to constructors; conversion of a class into an adequate constructor function and vice versa)

Módulo 3: Built-in objects

- Number constructor (number constructor - creating Number objects from data of different types, including various string formats; converting numbers into different string formats; static properties and methods of the Number constructor, i.e. properties defining the basic ranges)
- String constructor (String constructor; string as an array of characters; case conversion methods; splitting the string; searching for and replacing substrings; padding and trimming; comparison of strings)
- Date constructor (creating Date object - constructor; time zones and local time handling; getting current time; timestamp and using it to measure the time of code execution; time specification; operating on



individual date and time components)

- Array – basics (basic methods for managing an Array type collection - creating, merging, adding and removing items, passing through, slice method, splice method; use of spread operator; destructuring assignment)
- Array – advanced methods (the functional approach: find, every, some, filter, sort, map and reduce)
- Set type collection (concept of set data structure and Set object; basic methods and properties of Set objects - constructor, add, has, delete, clear, size; walking through elements - for ... of, iterators; spread operator)
- Map type collection (concept of map data structure and Map object; basic methods and properties of Map objects - constructor, set, get, has, delete, clear, size; walking through elements - for ... of, iterators; spread operator)
- Object as a data structure (using the object as a regular dictionary-type data structure; items management, walking through, spread operator)
- Using the JSON object (basic concept of JSON /JavaScript Object Notation/ format; converting data to JSON - JSON.stringify method; parsing JSON format and obtaining data back - JSON.parse method)
- Using the Math object (basic methods: ceil, floor, round, random, min, max, abs, pow, log, trigonometric functions)
- Using the RegExp object (basic rules for creating regular expressions; abbreviated notation of a RegExp object declaration; using methods of RegExp and String objects for efficient searching of patterns in text: test, exec, match, search, replace)
- Extending built-in types (using prototypes to extend built-in types - adding new properties and methods)

Módulo 4: Advanced function usage

- Extended function parameters handling (usage of default parameter values, rest parameter and spread operator; simulating named parameters)
- Closure and IIFE (the use of closure - execution environment of the function; Immediately Invoked Function Expression - IIFE)
- Forwarding calls (this keyword and functions; apply, call and bind methods that allow to call functions with the indicated context)
- Decorating functions (functions as first-class citizens; passing functions as arguments and returning as results, higher order functions; decorating functions as adding new functionality using a wrapper function)
- Generators and iterators (creating and using generators; the idea of iterable objects; generators as an element of iterable objects, iterators)
- Handling asynchronous events using callback functions (concept of asynchronous programming; use of callback functions to handle asynchronous events)



CAS TRAINING

UN ESPACIO PARA CRECER

cas-training.com

