

JavaScript

Avanzado

IFCD89. JavaScript avanzado.

Sku: PS_IFCD89

Horas: 36

Formato: HTML

OBJETIVOS

- Introducir las tecnologías, conceptos y competencias que se adquirirán a lo largo del desarrollo del curso para asegurar una adecuada codificación y despliegue de aplicaciones JavaScript.
- Estudiar los conceptos que permitirán experimentar y mejorar la legibilidad de la codificación desarrollada.
- Estudiar los conceptos que permitirán experimentar y mejorar la funcionalidad y la practicidad de su codificación desarrollada.
- Estudiar las ventajas obtenidas de la modularización de los desarrollos, de la utilización de un gestor de paquetes de código, de la programación asíncrona, de la eliminación del código innecesario y de la atención a las nuevas especificaciones publicadas por las actuales organizaciones internacionales de estandarización.
- Estudiar patrones y principios orientados a la mejora de los diseños desarrollados.
- Capacitar a los alumnos para la realización de pruebas unitarias del código desarrollado de cara a familiarizarse a cumplir los mínimos requisitos de calidad exigidos en la industria actual.
- Revisar las buenas prácticas que faciliten la detección, el control y la gestión de errores en la codificación
-

CONTENIDOS

- **Unidad 1. Introducción.**
 - Conocimientos / Capacidades cognitivas y prácticas
 - Potenciación de las generaciones de codificación limpia y funcional en desarrolladores con experiencia anterior en JavaScript.
 - Estructuración con facilidad de un mejor código y en módulos diferenciados.
 - Exploración de patrones y prácticas vigentes de diseño en la industria actual y cómo usarlos.

- Aprendizaje para la realización de pruebas unitarias.
 - Control y gestión adecuado de errores en JavaScript.
 - Habilidades de gestión. personales y sociales
 - Concienciación de la importancia de conocer de forma más especializada el lenguaje JavaScript.
 - Conocimiento de las ventajas que tiene la ampliación y desarrollo del manejo del lenguaje JavaScript para su desarrollo profesional.
 - **Unidad 2. Conceptos iniciales para la generación de código limpio (Clean code) con JavaScript.**
 - Conocimientos / Capacidades cognitivas y prácticas
 - Variables.
 - Qué es Lexical Scope (static scope). Uso de la instrucción "let". 7
 - Funciones.
 - Todo sobre la palabra clave "this".
 - Objetos and Clases.
 - Prevención de errores y formato homogéneo del código gracias a la herramienta Linter.
 - Actividad de evaluación de los conocimientos adquiridos por el alumno.
 - Habilidades de gestión. personales y sociales
 - Concienciación de la importancia de conocer de forma más especializada el lenguaje JavaScript.
 - Conocimiento de las ventajas que tiene la ampliación y desarrollo del manejo del lenguaje JavaScript para su desarrollo profesional.
 - **Unidad 3. Conceptos iniciales de generación de código funcional con JavaScript.**
 - Conocimientos / Capacidades cognitivas y prácticas
 - ¿Qué la programación funcional?
 - Las Funciones Puras en JavaScript.
 - El correcto uso del método Object.assign y el operador Spread.
 - Optimización de bucles en JavaScript.
 - Reducción de arrays.
 - Filtrado en arrays.
 - Actividad de evaluación de los conocimientos adquiridos por el alumno
 - Habilidades de gestión. personales y sociales
 - Concienciación de la importancia de conocer de forma más especializada el lenguaje JavaScript.
 - Conocimiento de las ventajas que tiene la ampliación y desarrollo del manejo del lenguaje JavaScript para su desarrollo profesional.
 - **Unidad 4. Módulos en JavaScript.**
 - Conocimientos / Capacidades cognitivas y prácticas
 - Reutilización de código con patrón Módulo (Module Pattern).
 - Utilización del gestor de paquetes Node Package Manager (NPM).
 - El proyecto CommonJS.
 - La API de definición asíncrona de módulos (AMD).
 - Módulos de ES6.
-

- Eliminación de código inactivo (Tree-Shaking) con Webpack.
 - Actividad de evaluación de los conocimientos adquiridos por el alumno.
 - Habilidades de gestión. personales y sociales
 - Concienciación de la importancia de conocer de forma más especializada el lenguaje JavaScript.
 - Conocimiento de las ventajas que tiene la ampliación y desarrollo del manejo del lenguaje JavaScript para su desarrollo profesional.
 - **Unidad 5. Patrones y principios de diseño.**
 - Conocimientos / Capacidades cognitivas y prácticas
 - Qué es el principio de responsabilidad única (Single Responsibility Principle).
 - El principio Open/Closed.
 - El principio de sustitución Liskov.
 - El principio de segregación de la interfaz.
 - El principio de Inversión de Dependencias (Dependency Inversion).
 - El patrón Singleton.
 - El patrón Observer.
 - El patrón Estrategia (Strategy).
 - Actividad de evaluación de los conocimientos adquiridos por el alumno.
 - Habilidades de gestión. personales y sociales
 - Concienciación de la importancia de conocer de forma más especializada el lenguaje JavaScript.
 - Conocimiento de las ventajas que tiene la ampliación y desarrollo del manejo del lenguaje JavaScript para su desarrollo profesional.
 - **Unidad 6. Pruebas del código generado.**
 - Conocimientos / Capacidades cognitivas y prácticas
 - Qué son y qué no son y la importancia de las pruebas unitarias.
 - Instalación y uso del framework de testing Jasmine.
 - Desarrollo guiado por pruebas de software, o Test-Driven Development (TDD).
 - Ejercicio guionizado para consolidar formación del alumno.
 - Habilidades de gestión. personales y sociales
 - Concienciación de la importancia de conocer de forma más especializada el lenguaje JavaScript.
 - Conocimiento de las ventajas que tiene la ampliación y desarrollo del manejo del lenguaje JavaScript para su desarrollo profesional.
 - **Unidad 7. Manejo de errores.**
 - Conocimientos / Capacidades cognitivas y prácticas
 - Estudio de buenas prácticas.
 - Detección de errores en operaciones asíncronas (Objeto Promise).
 - Estrategias y recomendaciones en el manejo de errores.
 - Generación de logs con JavaScript.
 - Actividad de evaluación de los conocimientos adquiridos por el alumno.
 - Habilidades de gestión. personales y sociales
-

- Concienciación de la importancia de conocer de forma más especializada el lenguaje JavaScript.
- Conocimiento de las ventajas que tiene la ampliación y desarrollo del manejo del lenguaje JavaScript para su desarrollo profesional.