

TypeScript - Producir tus desarrollos JavaScript

Curso práctico de 2 días - 14h

Ref.: TSR - Precio 2025: 1 110€ sin IVA

Descubrirás el lenguaje TypeScript, la superposición de JavaScript para el desarrollo Front-End. Aprenderás sobre tipado, clases y herencia para la programación orientada a objetos. Serás más productivo y podrás crear interfaces fáciles de mantener.

OBJETIVOS PEDAGÓGICOS

Al término de la formación, el alumno podrá:

- Familiarización con las herramientas de desarrollo
- Creación de funciones y matrices
- Dominar la herencia con TypeScript
- Ensamblar códigos para el servidor

PROGRAMA

última actualización: 03/2024

1) El principio de TypeScript

- ¿Por qué creó Microsoft esta alternativa?
- La superposición JavaScript.
- La comparación entre los 2 JavaScript y TypeScript.
- La elección del editor, los plug-ins y la instalación.
- El ecosistema de herramientas.
- Configuración del entorno de desarrollo.
- Depuración de la transpilación.

Trabajo práctico : Instalación de VS Code y configuración de plug-ins.

2) Configuración del compilador.

- El archivo «tsconfig.json».
- Las opciones del compilador.
- Creación de una compilación («build»).
- Estrategias de proyectos de configuración múltiple.
- Inclusión y exclusión de recursos.
- Creación de un flujo de trabajo personalizado.

Trabajo práctico : Manipulación del compilador de TypeScript y diseño de una compilación.

3) TypeScript Basics & Basic Types

- Tipos TypeScript frente a tipos JavaScript.
- ¿Cuándo y qué digitar?
- Expresión estática de tipos.
- Asignación e inferencia.
- Tuplas, enums, alias (tipos personalizados).
- Situaciones especiales de tipado (never, void...).
- Funciones y matrices.

Trabajo práctico : Creación de variables de clase con tipado estándar y especial.

4) JavaScript de próxima generación

- Impacto en la sintaxis de JavaScript.

PARTICIPANTES

Desarrolladores y gestores de proyectos web.

REQUISITOS PREVIOS

Programadores con conocimientos básicos de JavaScript.

COMPETENCIAS DEL FORMADOR

Los expertos que imparten la formación son especialistas en las materias tratadas. Han sido validados por nuestros equipos pedagógicos, tanto en el plano de los conocimientos profesionales como en el de la pedagogía, para cada curso que imparten. Cuentan al menos con entre cinco y diez años de experiencia en su área y ocupan o han ocupado puestos de responsabilidad en empresas.

MODALIDADES DE EVALUACIÓN

El formador evalúa los progresos pedagógicos del participante a lo largo de toda la formación mediante preguntas de opción múltiple, escenificaciones de situaciones, trabajos prácticos, etc. El participante también completará una prueba de posicionamiento previo y posterior para validar las competencias adquiridas.

MEDIOS PEDAGÓGICOS Y TÉCNICOS

- Los medios pedagógicos y los métodos de enseñanza utilizados son principalmente: ayudas audiovisuales, documentación y soporte de cursos, ejercicios prácticos de aplicación y ejercicios corregidos para los cursillos prácticos, estudios de casos o presentación de casos reales para los seminarios de formación.
- Al final de cada cursillo o seminario, ORSYS facilita a los participantes un cuestionario de evaluación del curso que analizarán luego nuestros equipos pedagógicos.
- Al final de la formación se entrega una hoja de presencia por cada media jornada de presencia, así como un certificado de fin de formación si el alumno ha asistido a la totalidad de la sesión.

MODALIDADES Y PLAZOS DE ACCESO

La inscripción debe estar finalizada 24 horas antes del inicio de la formación.

ACCESIBILIDAD DE LAS PERSONAS CON DISCAPACIDAD

¿Tiene alguna necesidad específica de accesibilidad? Póngase en contacto con la Sra. FOSSE, interlocutora sobre discapacidad, en la siguiente dirección psh-accueil@orsys.fr para estudiar de la mejor forma posible su solicitud y su viabilidad.

- Automatización de la documentación.
- Utilizar TypeScript con código JavaScript estándar.
- Incluir/generar archivos de definición para compatibilidad.
- Ventajas de los «getter,setter» implícitos.
- Programación anticipada, reducción de la deuda técnica.
- Módulos y espacios de nombres.

Trabajo práctico : Creación de un programa con TypeScript en combinación con JavaScript.

5) Mejorar la POO con TypeScript

- Recordatorios de patrones de POO: Singleton, Factory...
- El uso de interfaces TypeScript «Duck Typing».
- Clases e interfaces.
- El modificador de clase: «public, private, readonly...»
- Los métodos: «private, protected, overrides...»
- Herencia simple y múltiple.

Trabajo práctico : Diseño de herencia múltiple dentro de clases previamente creadas.

6) Tipos avanzados

- Gestión de tipos personalizados.
- Tipos genéricos e inferencias.
- Clases genéricas.
- Restricciones a los tipos genéricos.

Trabajo práctico : Desarrollo e implantación de tipos personalizados.

7) Decoradores

- Decoradores y Metadatos.
- Comprender las «factories» de decoradores.
- Decoradores de clases, propiedades, métodos y parámetros.
- Creación de decoradores personalizados.
- Implementación de abstracciones útiles.

Trabajo práctico : Implementación de un decorador de clases.

8) TypeScript en la práctica

- Uso con Node.js.
- Entrega de código JavaScript compatible con TypeScript.

FECHAS

Contacto