

ExecuTrain

Impulsamos tu talento tecnológico

MICROSOFT	RED HAT	VIRTUALIZACIÓN
CIBERSEGURIDAD	DESARROLLO	OFFICE
BIG DATA	BLOCK CHAIN	BASES DE DATOS
GESTIÓN DE SERVICIOS IT	CLOUD COMPUTING	METODOLOGÍAS EN PROYECTOS
SISTEMAS OPERATIVOS	Y MÁS...	

www.executrain.com.mx

¿Por qué ExecuTrain?

ExecuTrain es un proveedor de entrenamiento corporativo a nivel internacional y líder mundial en la capacitación empresarial. Contamos con más de 30 años de Experiencia y con más de 75 mil personas capacitadas a nivel Nacional.

Te guiamos en la definición de tus requerimientos de capacitación, en las diferentes etapas:

- ✓ Detección de necesidades, evaluación de conocimientos, plan de capacitación y seguimiento posterior para elegir el plan de capacitación como tú lo necesitas.
- ✓ El **más amplio catálogo de cursos**, desde un nivel básico hasta los niveles de conocimientos más especializados.
- ✓ En ExecuTrain el material y la **metodología están diseñados por expertos en aprendizaje humano**. Lo que te garantiza un mejor conocimiento en menor tiempo.
- ✓ Tú puedes confiar y estar seguro del aprendizaje porque nuestro **staff de instructores es de primer nivel**, algunos de los cuales son consultores en reconocidas empresas.
- ✓ No pierdas tu tiempo, los cursos están diseñados para un aprendizaje práctico.

Nuestro compromiso es que tú aprendas, si no quedas satisfecho con los resultados del programa, podrás volver a tomar los cursos hasta tu entera satisfacción o la devolución de tu dinero.

Modalidad de Servicio



Cursos en Fecha Calendario

Súmate a nuestros grupos en fechas públicas.



Cursos Privados

On site, en nuestras instalaciones o en línea con instructor en vivo.



Autoestudio con soporte de instructor

Cursos en modalidad autoestudio, con acceso 24/7 a la plataforma de estudio, con soporte de instructor y foros de ayuda

ExecuTrain
Impulsamos tu talento tecnológico

Duración: 8 horas

Claude / Claude Code para Desarrolladores

La inteligencia artificial está transformando el desarrollo de software. Herramientas agentivas como Claude Code permiten explorar repositorios, analizar código, planificar soluciones, implementar cambios, diagnosticar errores y automatizar tareas técnicas. Sin embargo, aprovechar estas capacidades profesionalmente requiere mucho más que saber escribir instrucciones: exige un método de trabajo que garantice control, calidad, seguridad y resultados verificables.

Al finalizar el taller, el participante será capaz de utilizar Claude Code como asistente agentivo dentro de un flujo profesional de desarrollo de software, aplicando prácticas estructuradas para comprender proyectos, planificar e implementar cambios, diagnosticar defectos, automatizar verificaciones, revisar código y establecer controles de seguridad, con evidencia verificable y supervisión humana.

Perfil del Público

- Desarrolladores de software interesados en integrar inteligencia artificial en sus actividades de programación, mantenimiento y evolución de aplicaciones.
- Ingenieros de software que trabajan con repositorios, pruebas, depuración, revisión de código y control de versiones.
- Profesionales que desarrollan soluciones con tecnologías como C, .NET/C#, Android/Kotlin e iOS/Swift.
- Líderes técnicos y arquitectos de software que buscan establecer prácticas de desarrollo asistido por IA.
- Integrantes de equipos DevOps, QA e ingeniería que participan en procesos de integración y validación de cambios.
- Equipos y organizaciones que desean incorporar herramientas agentivas mediante procesos controlados, seguros y reproducibles.

Requisitos Previos

- ✓ Conocimientos básicos de programación y capacidad para comprender código fuente.
- ✓ Familiaridad con la estructura general de proyectos de software.
- ✓ Conocimientos básicos de Git, incluyendo repositorios, ramas, commits y Pull Requests.
- ✓ Experiencia básica con terminales y ejecución de comandos.
- ✓ Conocimiento general de compilación, depuración y pruebas de software.
- ✓ Visual Studio Code y la extensión oficial de Claude Code instalados.
- ✓ Acceso autorizado a Claude Code mediante una suscripción o cuenta compatible.
- ✓ Git, las herramientas de desarrollo correspondientes a su tecnología y el repositorio de laboratorio disponibles en el equipo.

Módulos

Módulo 1. Fundamentos de Claude Code y ecosistema

Reconocer las capacidades de Claude Code, su entorno de trabajo y los mecanismos de control necesarios para utilizarlo de forma segura.

- Introducción a Claude Oriented Development (CODE).
- Ecosistema Claude y diferencias entre sus herramientas.
- Claude Code como herramienta de desarrollo agentiva.
- Entorno de trabajo con VS Code, extensión oficial y repositorio local.
- Capacidades de lectura, edición y ejecución de comandos.
- Modos de permisos, aprobaciones y límites de autonomía.
- Introducción al workflow CODE de cinco fases.
- **Misión práctica:** Primera exploración de un repositorio en modo de solo lectura, con verificación de la información obtenida.

Módulo 2. Comprensión del repositorio y contexto útil

Explorar y comprender repositorios existentes, proporcionando contexto preciso y verificable a Claude Code.

- Estrategias de exploración de repositorios desconocidos.
- Estructura de proyectos, dependencias y puntos de entrada.
- Identificación de modelos, lógica de negocio y flujos de ejecución.
- Técnicas de prompting para análisis de código.
- Contexto de sesión y referencias de archivos.
- Configuración de instrucciones persistentes mediante CLAUDE.md.
- Administración del contexto y reglas del proyecto.
- Verificación de respuestas mediante evidencia del repositorio.
- Misión práctica: Construcción de un mapa verificado del proyecto y creación de su archivo CLAUDE.md.

Módulo 3. Planificación y desarrollo asistido

Transformar requerimientos en planes técnicos verificables e implementar cambios con alcance controlado.

- Identificación y resolución de requerimientos ambiguos.
- Elaboración de briefs técnicos.
- Objetivos, alcance, restricciones y no-objetivos.
- Criterios de aceptación y estrategias de verificación.
- Planificación y revisión de propuestas de implementación.
- Análisis de impacto, dependencias y cambios multiarchivo.
- Prevención de desviaciones de alcance (scope creep).
- Revisión de diffs y verificación inicial.
- Misión práctica: Implementación de una nueva funcionalidad de reportes a partir de un requerimiento, con planificación y revisión de cambios.

Módulo 4. Debugging y corrección controlada

Diagnosticar defectos mediante evidencia, identificar sus causas y aplicar correcciones mínimas verificables.

- Metodología de debugging asistido por IA.
- Reproducción de defectos e identificación de síntomas.
- Formulación y contraste de hipótesis.
- Análisis de código, logs, variables y ejecución.
- Depuración con VS Code y herramientas del stack.

- Corrección mínima y evaluación de efectos secundarios.
- Pruebas de regresión y registro de evidencia.
- Misión práctica: Investigación de un defecto reproducible, confirmación de su causa y aplicación de una corrección controlada.

Módulo 5. Refactoring y Automated Testing

Mejorar la calidad interna del código y construir pruebas automatizadas que permitan verificar su comportamiento.

- Diferencias entre refactoring, corrección de defectos y desarrollo de funcionalidades.
- Identificación de code smells y responsabilidades mezcladas.
- Refactoring incremental asistido por Claude Code.
- Generación y revisión crítica de pruebas unitarias.
- Pruebas de casos normales, límites, errores y regresión.
- Fixtures y datos de prueba deterministas.
- Ejecución de builds, tests y herramientas de verificación.
- Interpretación de resultados y evidencia reproducible.
- Misión práctica: Construcción de una red de pruebas automatizadas y ejecución de un refactoring controlado.

Módulo 6. Git Workflow, Code Review y Pull Requests

Gestionar cambios de código con trazabilidad y preparar entregas revisables mediante Git y Claude Code.

- Git como mecanismo de trazabilidad.
- Inspección de cambios con git status y git diff.
- Ramas de trabajo y preparación de commits.
- Generación y revisión de mensajes de commit.
- Elaboración de Pull Requests con resumen, pruebas y riesgos.
- Code Review asistido por inteligencia artificial.
- Revisión automatizada y aprobación humana.
- Criterios para preparar cambios listos para integración.
- Misión práctica: Organización de cambios en Git y preparación de un Pull Request técnicamente revisable.

Módulo 7. Seguridad, gobernanza y estándar CODE

Identificar riesgos del desarrollo agentivo y aplicar controles técnicos y organizacionales para un uso responsable de Claude Code.

- Riesgos del desarrollo de software asistido por IA.
- Protección de código propietario, secretos y datos sensibles.
- Permisos, ejecución de comandos y sandboxing.
- Prompt injection y tratamiento de contenido no confiable.
- Model Context Protocol (MCP) y límites de confianza.
- Hooks, plugins y automatizaciones.
- Matriz de decisión: permitir, restringir o rechazar.
- Escalamiento de cambios sensibles.
- Definición de un estándar mínimo CODE.
- Misión práctica: Aplicación de controles de seguridad al proyecto y evaluación de instrucciones provenientes de contenido no confiable.