

Presento el diseño de un curso de "Java Inicial Asistido con IA" con la idea de actualizar el alcance y contenidos de los cursos que se vienen dando. El objetivo es doble: los alumnos no solo aprenden la sintaxis y la lógica de Java, sino que también se convierten en profesionales más eficientes y modernos al dominar las herramientas de IA que ya son estándar en la industria.

Java desde Cero con Asistente IA: Tu Primer Paso en la Programación Profesional

Objetivos Generales del Curso:

- Comprender los fundamentos de la programación en Java y la lógica computacional.
- Dominar los principios de la Programación Orientada a Objetos (POO).
- Aprender a utilizar la IA (como GitHub Copilot o ChatGPT) como una herramienta de trabajo: para generar código *boilerplate*, explicar conceptos complejos, depurar errores y refactorizar código.
- Desarrollar un proyecto final que integre todos los conocimientos, desde la consola hasta una interfaz básica.
- Desarrollar el pensamiento crítico para no utilizar ciegamente el código de la IA, sino para entenderlo, validarlo y mejorarlo.

Temario Detallado (16 Clases de 4 horas)

Módulo 1: Fundamentos y Primeros Pasos con IA (Clases 1-4)

Clase 1: Introducción al Curso y "Hola, IA"

Teoría: ¿Qué es Java? Historia, características y máquina virtual (JVM). Instalación del JDK y un IDE (IntelliJ IDEA Community o Eclipse). Estructura básica de un programa Java.

Práctica con IA: Configuración de un asistente de IA (ej. extensión GitHub Copilot en el IDE o acceso vía web a ChatGPT). Ejercicio: Usar un prompt de IA para que genere el primer "Hola, Mundo". Analizar el código generado, identificar cada parte (clase, método main, System.out.println). Modificar el mensaje para que el saludo sea personalizado.

Clase 2: Variables, Tipos de Datos y la IA como "Explicadora"

Teoría: Variables en Java. Tipos de datos primitivos (int, double, char, boolean). Declaración, inicialización y ámbito de variables. Operadores aritméticos y de asignación.

Práctica con IA: Ejercicio: Calcular el área de un rectángulo. Uso de IA: Pedir a la IA que explique la diferencia entre int y double con ejemplos de la vida real. Hacer que la IA genere pequeños fragmentos de código con errores de tipo para que los alumnos los depuren manualmente.

Clase 3: Entrada de Usuario y Estructuras Condicionales con IA

Teoría: La clase Scanner para leer datos del teclado. Operadores relacionales. Estructuras if, else if, else.

Práctica con IA: Ejercicio: Crear un simulador de nota de alumno que indique "Aprobado" o "Reprobado". Uso de IA: Usar la IA para generar la estructura de if-else compleja (ej. múltiples rangos de calificaciones con letras A, B, C, D, F). Discutir por qué la IA eligió esa estructura y si se puede optimizar.

Clase 4: Estructuras Repetitivas y la IA como "Generadora de Casos de Prueba"

Teoría: Bucles while, do-while y for. Cuándo usar cada uno. Uso de break y continue.

Práctica con IA: Ejercicio: Crear un juego simple de adivinanza de números (el programa genera un número aleatorio y el usuario debe adivinarlo con pistas). Uso de IA: Pedir a la IA que genere 5 casos de prueba diferentes para el juego (ej. números negativos, letras en lugar de números, etc.) para entender la importancia de probar el software.

Módulo 2: Programación Orientada a Objetos y Depuración con IA (Clases 5-8)

Clase 5: Introducción a la POO: Clases y Objetos con IA

Teoría: Conceptos de clase, objeto, atributo y método. Creación de una clase simple. Instanciación de objetos.

Práctica con IA: Ejercicio: Modelar una clase Coche con atributos (marca, modelo, velocidad) y métodos (acelerar, frenar). Uso de IA: Pedir a la IA que, a partir de una descripción en texto ("Quiero una clase que represente un libro con título, autor y número de páginas"), genere el esqueleto de la clase. Comparar la estructura generada con lo aprendido.

Clase 6: Métodos Constructores, Encapsulamiento y la IA

Teoría: Constructores por defecto y parametrizados. Getters y Setters. Encapsulamiento (modificadores de acceso private / public).

Práctica con IA: Ejercicio: Refactorizar la clase Coche para que sus atributos sean private y crear sus getters y setters. Uso de IA: Usar la IA para generar automáticamente los constructores, getters y setters a partir de los atributos definidos (una función común en los IDE modernos, impulsada por IA).

Clase 7: Arreglos y la IA como "Optimizadora de Código"

Teoría: Declaración, creación e inicialización de arreglos unidimensionales. Recorrer arreglos con bucles.

Práctica con IA: Ejercicio: Crear un programa que guarde las notas de 5 alumnos, calcule el promedio y encuentre la nota más alta. Uso de IA: Mostrar a la IA el código funcional de un alumno y pedirle: "Optimiza este código para que sea más legible y eficiente". Analizar las sugerencias de refactorización.

Clase 8: Depuración Avanzada y Manejo de Errores con IA

Teoría: Tipos de errores: sintaxis, lógica y de ejecución. Introducción a try-catch para manejo de excepciones básicas (ej. InputMismatchException).

Práctica con IA: Ejercicio: Proporcionar a los alumnos un programa con errores lógicos y de ejecución ocultos. Uso de IA: Usar la IA como "compañero de depuración". Pegar el mensaje de error y el fragmento de código problemático a la IA para que explique la causa y sugiera una corrección. Enfocarse en entender por qué ocurría el error, no solo en copiar la solución.

Módulo 3: Herramientas Profesionales y Proyecto Integrador con IA (Clases 9-12)

Clase 9: Git, GitHub y la IA para Escribir Mensajes de Commit

Teoría: Concepto de control de versiones. Introducción a Git: init, add, commit, push. Creación de una cuenta en GitHub.

Práctica con IA: Ejercicio: Subir el proyecto del juego de adivinanza a un repositorio de GitHub. Uso de IA: Aprender a usar la IA para generar mensajes de commit descriptivos a partir de los cambios realizados. Herramientas como GitHub Copilot pueden sugerir estos mensajes automáticamente.

Clase 10: Colecciones en Java: ArrayList y la IA

Teoría: Limitaciones de los arreglos. Introducción al Framework de Colecciones. Uso de ArrayList para manejar listas dinámicas.

Práctica con IA: Ejercicio: Modificar el proyecto de notas para que pueda manejar las notas de un número indeterminado de alumnos usando ArrayList. Uso de IA: Consultar a la IA: "¿Cuál es la diferencia entre ArrayList y LinkedList? Dame un ejemplo de cuándo usar cada uno".

Clase 11: Proyecto Integrador (Parte 1) - Diseño y Configuración con IA

Objetivo: Inicio del proyecto final. Ejemplo: "Sistema de Gestión de una Biblioteca".

Tareas: Definir las clases necesarias (Libro, Usuario, Préstamo). Usar la IA como "arquitecto de software": pedirle que sugiera los atributos y métodos principales para cada clase basándose en una descripción del proyecto. Configurar el repositorio en GitHub con un README.md generado por IA.

Clase 12: Proyecto Integrador (Parte 2) - Lógica de Negocio con IA

Objetivo: Implementar la lógica central del programa.

Tareas: Desarrollar métodos para agregar/eliminar libros, registrar usuarios y prestar/devolver libros. Uso de IA: Para las funcionalidades más complejas (ej. "evitar que un libro sea prestado si no hay copias disponibles"), pedir a la IA que genere la lógica condicional y luego el alumno la adapta e integra a su proyecto.

Módulo 4: Más Allá de la Consola y el Futuro con IA (Clases 13-16)

Clase 13: Introducción a la Programación Visual con Java Swing (Parte 1)

Teoría: Concepto de interfaz gráfica. Introducción a Swing. Componentes básicos: JFrame, JPanel, JButton, JLabel.

Práctica con IA: Ejercicio: Crear una ventana simple con un botón y una etiqueta. Uso de IA: Usar la IA para generar el "código boilerplate" de la ventana y conectar el evento de clic del botón.

Clase 14: Swing y Eventos (Parte 2)

Teoría: Manejo de eventos. Listeners (ej. ActionListener).

Práctica con IA: Ejercicio: Conectar la interfaz gráfica con el proyecto de la biblioteca. Crear una pantalla para buscar libros. Uso de IA: Pedir a la IA que explique cómo funciona un ActionListener con una analogía, y que genere un ejemplo de cómo actualizar una JList con los resultados de la búsqueda.

Clase 15: El Rol de la IA en el Desarrollo Moderno y Ética

Teoría: Más allá del código: IA para generar documentación, pruebas unitarias básicas y sugerencias de arquitectura. Discusión sobre los límites de la IA: sesgos, seguridad, código de baja calidad. La importancia del criterio profesional. Introducción a herramientas más avanzadas como GitHub Copilot Chat.

Práctica con IA: Ejercicio: Usar la IA para generar la documentación JavaDoc para el proyecto de la biblioteca. Usar la IA para escribir una pequeña reflexión sobre el impacto de la IA en el trabajo del programador.

Clase 16: Presentación de Proyectos y Cierre

Actividad: Cada alumno presenta su proyecto final de biblioteca, explicando no solo el código, sino también cómo usaron la IA durante el proceso (qué prompts usaron, qué problemas les ayudó a resolver, y en qué momentos tuvieron que corregirla o mejorarla). Feedback grupal y reflexión final sobre el curso y el futuro del aprendizaje con IA.

Herramientas y Recursos Sugeridos:

- IDE: IntelliJ IDEA Community Edition (recomendado) o Visual Studio Code con extensiones de Java y GitHub Copilot.
- Asistente IA: GitHub Copilot (ideal por su integración), ChatGPT (versión gratuita o de pago), o Google Gemini.
- Control de Versiones: Git y GitHub.
- Documentación: La API de Java y tutoriales.