

Sesión de tutoría técnica donde se abordan conceptos fundamentales de **programación en Java, bases de datos (SQL)**, el uso de **Git** y el desarrollo pdel estudiante en el ámbito tecnológico.

A continuación, se resumen los puntos principales tratados:

1. Programación Avanzada en Java

Gran parte de la sesión se centra en aclarar dudas sobre la estructura y lógica del lenguaje Java:

- **Enumeraciones (Enums):** Se definen como tipos de **datos personalizados e inamovibles** que se utilizan para definir variables finales estáticas, lo que ayuda a evitar errores y dar mayor legitimidad al código,,. Un ejemplo práctico discutido es el estado de un pedido (pendiente, enviado, cancelado), el cual no debe cambiar arbitrariamente,.
- **Uso de this y Ambigüedad:** Se debate sobre la palabra clave this, aclarando que se usa principalmente cuando hay **redundancia o ambigüedad** en los nombres de las variables, aunque algunos docentes la utilizan de forma opcional incluso cuando no es necesaria,,.
- **Herencia y Polimorfismo:** El tutor utiliza una **analogía de Minecraft** para explicar las clases padre e hijas: una clase "Monstruo" (padre) define atributos generales como "ID" y "fuerza", mientras que clases como "Zombi" o "Creeper" (hijas) heredan esos atributos y añaden comportamientos propios mediante el uso de extends y @Override,,.
- **Serialización:** Se menciona que para enviar objetos a través de una red o a una base de datos, es necesario que la clase implemente Serializable,.

2. Bases de Datos (SQL)

Se revisan los fundamentos de SQL, aclarando que, a diferencia de HTML o CSS, este sí se considera un **lenguaje de programación** para manejar bases de datos,.

- **Relacional vs. No Relacional:** Se discute la diferencia entre bases de datos SQL (relacionales), que requieren una estructura rígida de tablas y campos obligatorios, y las NoSQL (como MongoDB), que permiten mayor flexibilidad.
- **Consultas y Filtros:** Se repasan comandos esenciales como SELECT, WHERE, AND, OR y el uso de LIKE para buscar patrones en cadenas de texto,,. También se mencionan conceptos avanzados como JOIN, GROUP BY y el uso de HAVING para trabajar con funciones agregadas como la media

3. Control de Versiones con Git y GitHub

El tutor guía a la estudiante en la configuración de su entorno de trabajo profesional:

- **Flujo de Trabajo:** Se explica cómo iniciar un repositorio local (git init), preparar archivos (git add), registrar cambios (git commit) y subirlos al servidor remoto en GitHub (git push),,,.
- **Colaboración:** Se detalla el proceso para invitar a colaboradores en GitHub para que otros puedan ver y trabajar en el código remoto,.
- **Buenas Prácticas:** El tutor recomienda trabajar siempre en el disco duro del ordenador y no en discos externos para evitar errores de repositorio,.

4. Contexto Académico y Motivación

El tutor la anima a ser **autodidacta**, aprender mecanografía para ser más eficiente y ver la programación como una herramienta para destacar en un mercado competitivo.