

Preguntas evaluativas anteriores: Dar una respuesta común a todas las anteriores, solo aquellos puntos que se quieran destacar. ¿Qué IDEs son mejores? ¿qué plugin son mejores? ¿Qué funcionalidades son más interesantes? etc....

Preguntas evaluativas (para responder en grupo):

1. ¿Qué características comunes comparten los IDEs en términos de edición de código, depuración y control de versiones.

Edición de código

- Autocompletado de código, haciéndote sugerencias.
- Resaltado de sintaxis
- Ambos IDEs te permiten refactorizar el código
- Los dos te permiten trabajar con linters y formateadores, aunque en el caso de VSC hay que usar extensiones (ESLint y Prettier) e IntelliJ lo tiene integrado.
- VSC permite bastantes lenguajes de programación instalando las extensiones correspondientes, IntelliJ también acepta muchos lenguajes, aunque acepta mejor lenguajes como Java, Kotlin y Python.

Depuración

- Ambos traen integrados herramientas para depurar que incluyen la posibilidad de añadir puntos de ruptura, ver cómo cambian el estado de las variables línea por línea...

Control de versiones

- Los dos funcionan perfectamente con Git permitiendo acciones como clonar repositorios, hacer commits, push, pull, etc. directamente desde los IDEs.
- Además permiten ejecutar comandos de Git directamente desde el terminal.

2. ¿Qué diferencias notaron en la forma en que los IDEs manejan módulos, personalización y generación de ejecutables?

Manejo de módulos

- IntelliJ IDEA tiene un sistema para proyectos con múltiples módulos, especialmente en Java/Kotlin
- VSCode no tiene un sistema tan estructurado.

Personalización

- IntelliJ ofrece personalización más limitada, pero optimizada para lenguajes específicos.
- Visual Code Studio se puede personalizar completamente.

Generación de ejecutables

- IntelliJ IDEA permite generar ejecutables de forma nativa
- Visual Studio Code requiere herramientas externas o configuraciones manuales.

3. ¿Cuál de los IDEs elegidos consideran que es más adecuado para proyectos de desarrollo específicos y por qué?

Es mejor los IDEs propietarios porque tienen herramientas específicas para un lenguaje en concreto. Según el proyecto que necesites realizar. Si necesitas trabajar un proyecto en python, sería mejor utilizar pycharm, si necesitas un proyecto en java o kotlin, sería más adecuado en IntelliJ IDEA.

4. ¿Qué IDE recomendarían a un equipo de desarrollo que trabaja en proyectos de Python, Java o Kotlin y por qué?

IntelliJ IDEA es el más adecuado para Kotlin y Java debido a las herramientas y frameworks que contiene específicamente para ellos, aunque para Python no está tan especializado como otros IDEs.

Tabla comparativa

<u>Características</u>	<u>Visual Studio Code</u>	<u>IntelliJ Idea</u>
Edición de Código	Resaltado de sintaxis, autocompletado, refactorización.	Resaltado de sintaxis, autocompletado, refactorización.
Depuración	Depuración básica con extensiones.	Herramientas de depuración avanzadas, integración profunda con lenguajes.
Control de Versiones	Integración con Git a través de extensiones.	Integración nativa con Git, SVN, Mercurial.
Manejo de módulos	Soporte básico para proyectos con múltiples carpetas.	Personalización limitada pero optimizada para lenguajes específicos.
Personalización	Alta personalización mediante extensiones.	Personalización limitada pero optimizada para lenguajes específicos.
Generación de ejecutables	Requiere configuraciones externas o extensiones.	Generación nativa de ejecutables (JAR, WAR, etc.)
Soporte de Lenguajes	Amplio soporte mediante extensiones (Python, JS, etc.)	Soporte nativo para Java, Kotlin y otros lenguajes que usan JVM.
Ideal para	Proyectos ligeros, desarrollo web, Python, JavaScript.	Proyectos grandes en Java, Kotlin y otros ecosistemas JVM.
Integración con herramientas externas	Requiere configuración manual para herramientas externas.	Integración automática con herramientas como Maven, Gradle, Spring.
Velocidad y uso de recursos	Ligero y rápido, ideal para proyectos pequeños.	Más pesado, adecuado para proyectos grandes y complejos.
Precio	Gratuito.	De pago, aunque tiene una versión gratuita (Community).