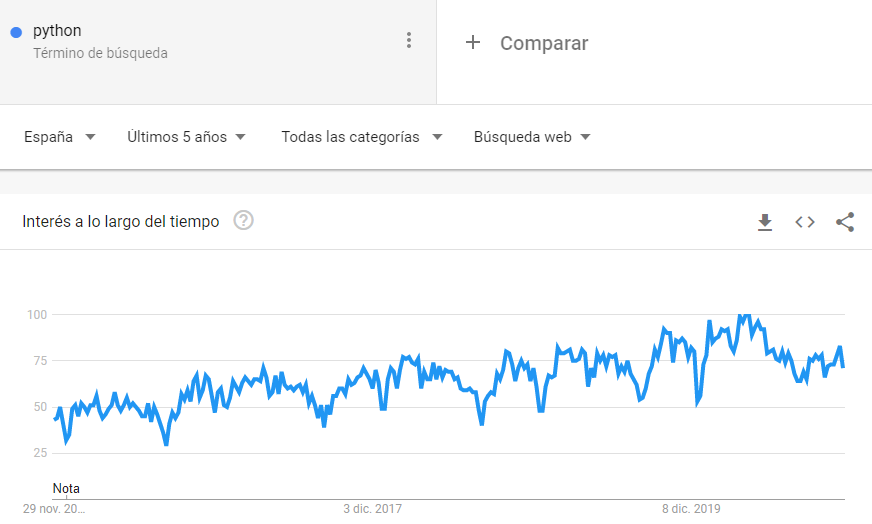
**Entornos de desarrollo integrados**

Utiliza la máquina de Windows 10, o en su defecto tu equipo personal desde casa, para realizar las siguientes tareas. Documenta la respuesta de cada apartado con capturas de pantalla donde sea necesario.

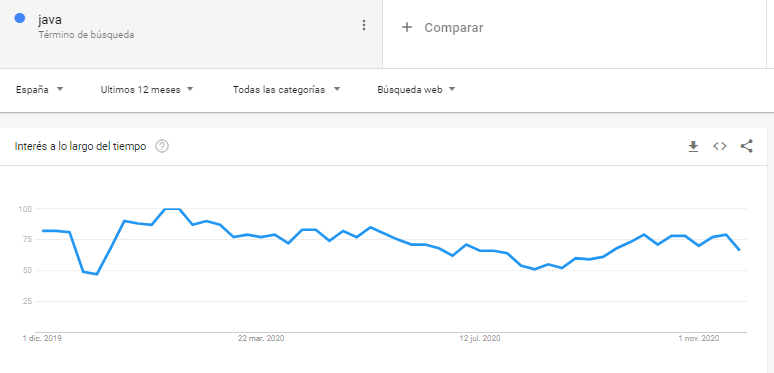
1. Utiliza **Google Trends** para visualizar la importancia de los lenguajes Python, Java, C++ , Ruby, PHP e .NET. Vai modificando la búsqueda para mostrar la gráfica en el último año y en los últimos cinco años.

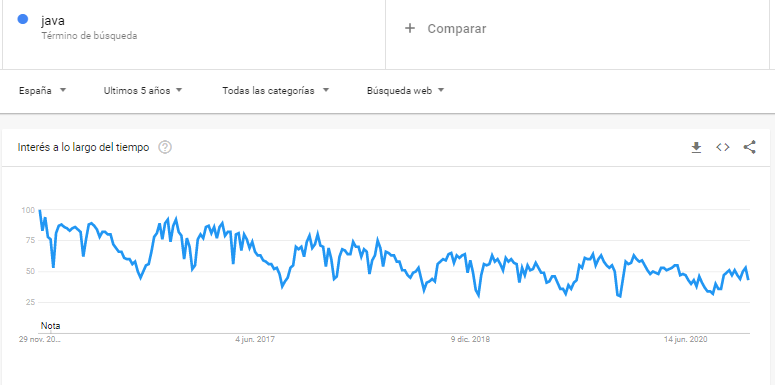
Python



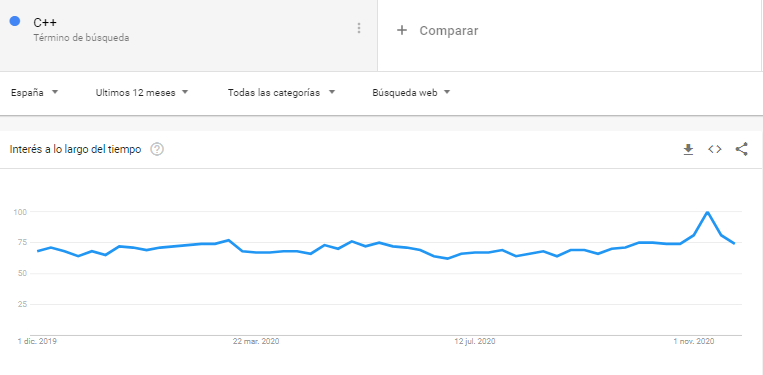


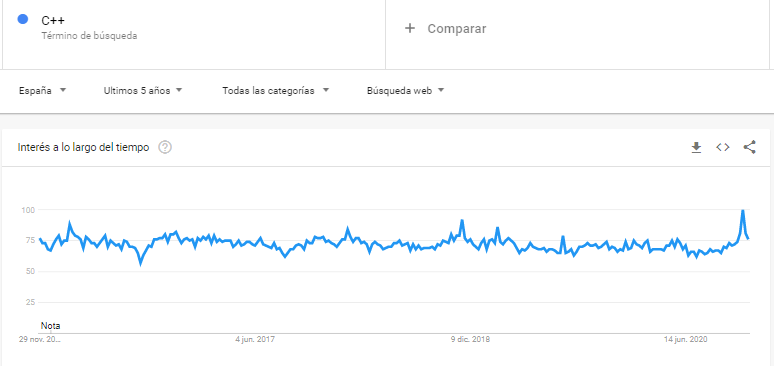
Java



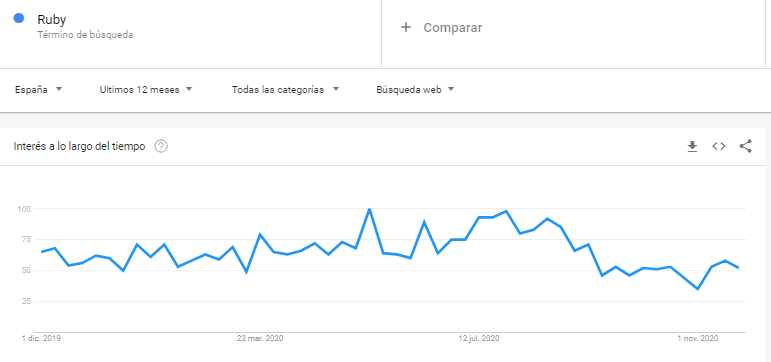


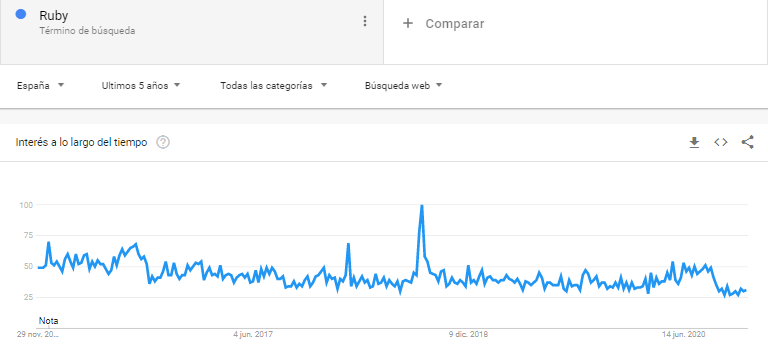
C++





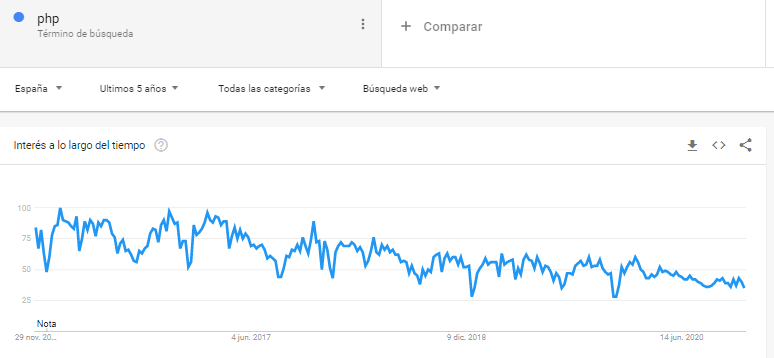
Ruby





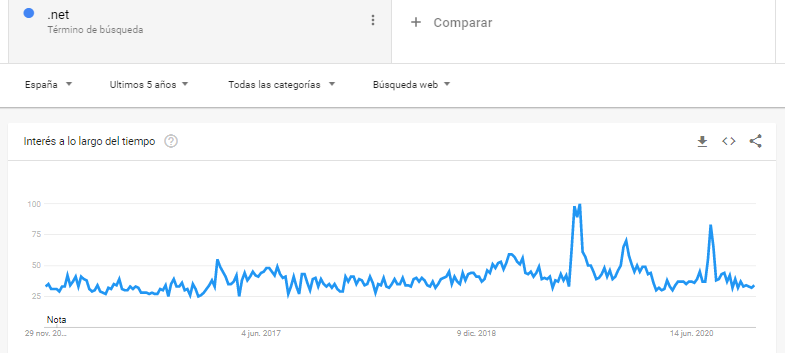
PHP





.NET



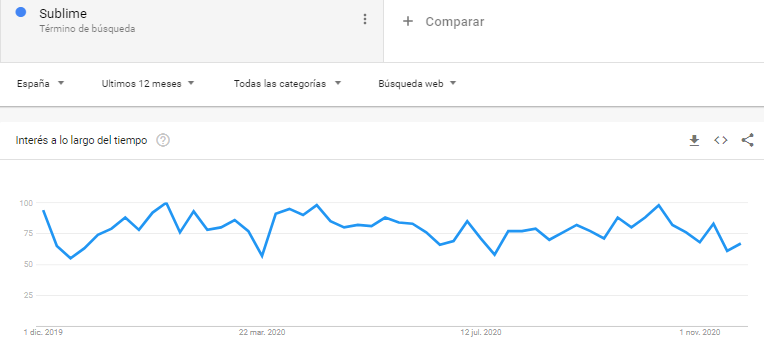


1. Utiliza **Google Trends** para visualizar la importancia de los editores multipropósito Atom, Sublime, Brackets e Visual Studio Code.

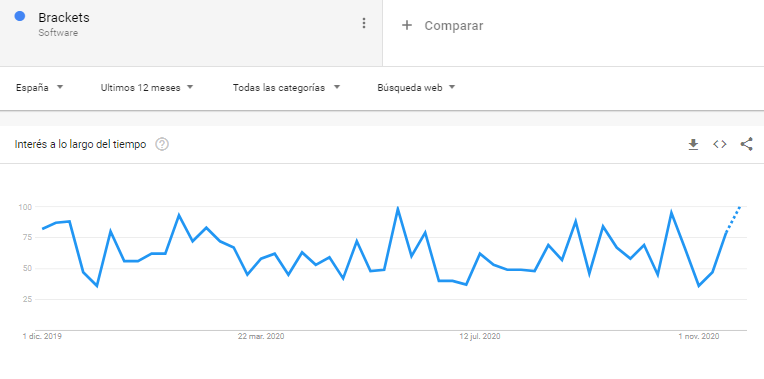
Atom



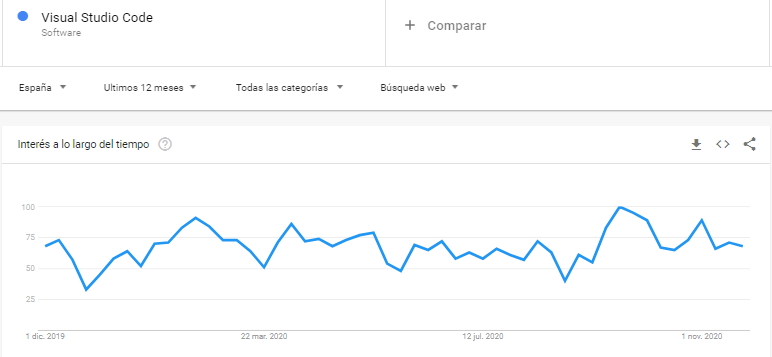
Sublime



Brackets



Visual Studio Code



1. Indica a que categorías pueden pertenecer las siguientes aplicaciones, en base a la siguiente tabla. Razona también la diferencia entre editores multipropósito e IDEs.

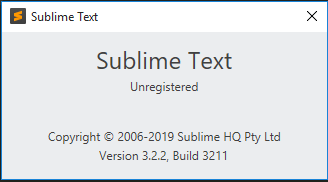
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Eclipse** | **NetBeans** | **IntelliJ** | **VS Code** | **Atom** | **Sublime** | **Vim** | **Emacs** | **Bloc de Notas** | **Gedit** | **TextEditor** | **Brackets** |
| **IDE** | X | X | X |  |  |  |  |  |  |  |  |  |
| **Editor avanzado/sofisticado** |  | X |  | X | X | X |  | X |  |  | X |  |
| **Editor de texto** |  |  |  |  |  |  | X |  | X | X |  |  |

1. ¿Qué características debería aportar como mínimo un IDE para ayudarnos a desarrollar código?

Un IDE debería aportar las siguientes características:

1. Multiplataforma
2. Soporte para diversos lenguajes de programación
3. Integración con Sistemas de Control de Versiones
4. Reconocimiento de Sintaxis
5. Extensiones y Componentes
6. Depurador
7. Importar y Exportar proyectos
8. Instala y configura las siguientes aplicaciones. Toma una captura de pantalla donde se pueda observar su versión desde el propio programa:
   1. Sublime Text
   2. Brackets
   3. Visual Studio Code
   4. Notepad++
   5. Atom

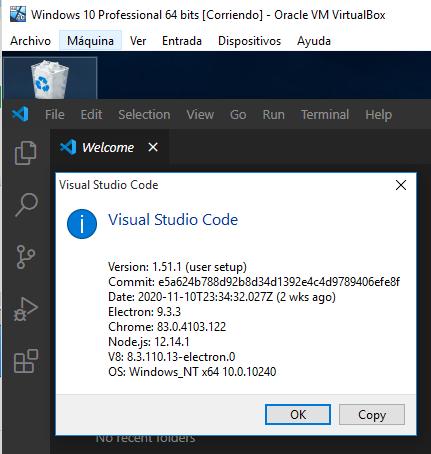
**Sublime Text**



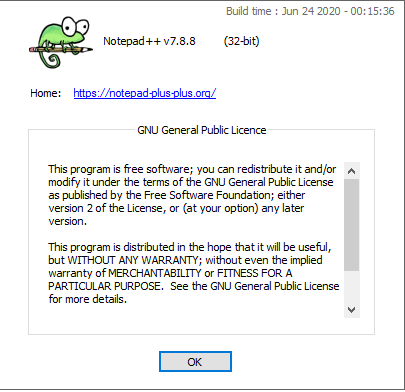
**Brackets**



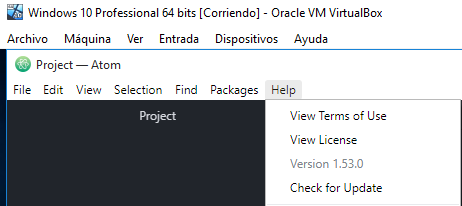
**Visual Studio Code**



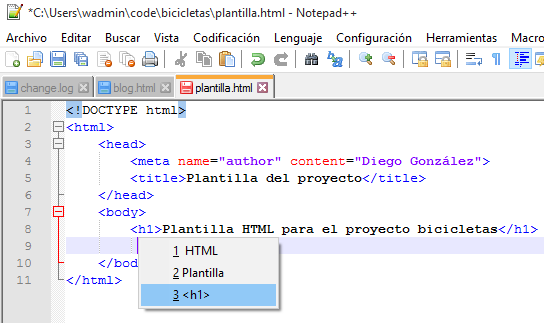
**Notepad++**



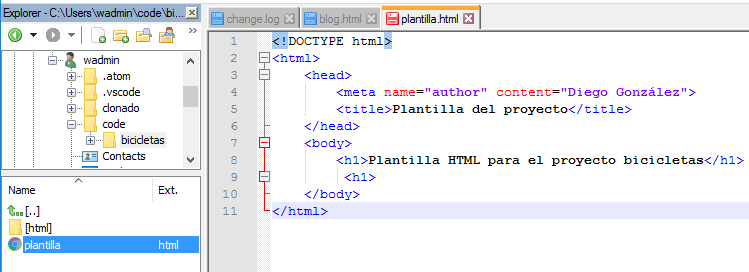
**Atom**



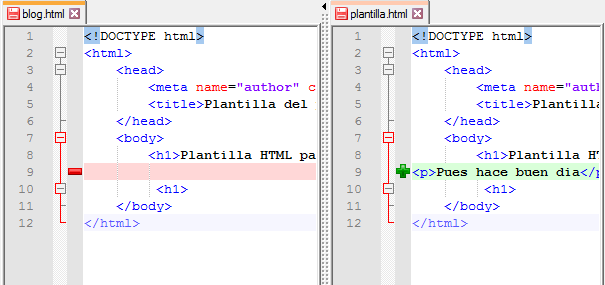
1. Instala los siguientes plugins para Notepad++ (<https://www.manuprieto.es/plugins-esenciales-progamar-notepad/>) explicando su utilidad de forma muy breve. Aporta una captura donde se vea su funcionalidad.
   1. Multiclipboard: Permite almacenar un historial del portapapeles, por lo que se puede copiar varios elementos y luego seleccionar uno para pegar.



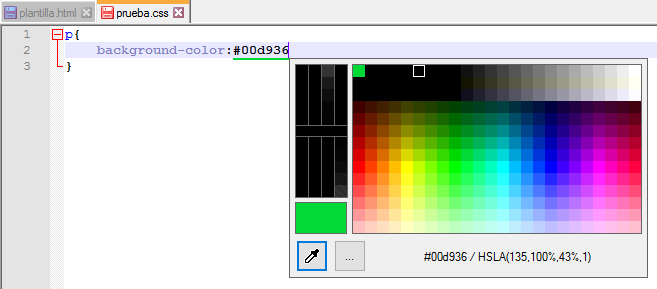
* 1. Explorer: Permite colocar un explorado de archivos para poder buscar y abrir cualquier documento o archivo sin necesidad de buscarlo desde el explorador de Windows.



* 1. Compare: Compara textos de forma puntual, y muestra las diferencias resaltadas.



* 1. QuickColorPicker: En una hoja de estilos CSS muestra un preview del color y dá una posibilidad de abrir una paleta de colores y remplazar el ya existente.



1. Instala los siguientes paquetes para Sublime Text explicando su utilidad de forma muy breve. Aporta una captura donde se vea su funcionalidad. (<https://www.genbeta.com/desarrollo/10-packages-de-sublimetext-para-desarrolladores-web>  
   <https://www.easyappcode.com/10-packages-de-sublime-text-que-todo-desarrollador-web-deberia-conocer> )
   1. SideBarEnhancements: Añade operaciones a la barra lateral.
   2. Emmet: Aumatiza la creación de bloques de etiquetas HTML utilizando abreviaciones.
   3. HTML-CSS-JS Prettify (*para que funcione se debe instalar Node.js en el sistema operativo)*: Formatea el texto de forma automática de los lenguajes: HTML, CSS y JavaScript.
   4. BracketHighlighter: Se encarga de destacar los límites superiores e inferiores de cada pareja de llaves.
   5. Terminal: Permite abrir un terminal situado en la carpeta actual donde se esté trabajando.
   6. Seti UI: Mejora la interfaz de la lista de archivos, donde asocia un icono a cada archivo.
   7. Live Reload: Permite la carga en tiempo real en el navegador cuando se está escribiendo el código.
   8. ColorPicker: Permite seleccionar en la paleta de color un color especifico.
2. Instala las siguientes extensiones para Brackets explicando su utilidad de forma muy breve. Aporta una captura donde se vea a su funcionalidad:

(<https://www.designbombs.com/best-free-brackets-extensions/> )

* 1. Emmet: Permite escribir expresiones similares a CSS en forma de abreviatura que se pueden analizar dinámicamente y producir una salida.
  2. Beautify: Mejora la interfaz cambiando los archivos para que sean más legibles.
  3. Custom Work: Agrega una barra de herramientas que incluye botones para mostrar/ocultar las pestañas de documentos abiertos.
  4. Minifier: Elimina caracteres innecesarios de los archivos
  5. TODO: Permite agregar tags sobre el código
  6. Brackets File Icons: Añade iconos en el directorio/arbol de archivos
  7. Minimap: Incluye funciones como resaltado de sintaxis, función de ocultación automática, transparencias, función de vista previa, zoom y scroll suave
  8. CSSLint

1. Instala los siguientes plugins para Atom explicando su utilidad de forma muy breve. Aporta una captura donde se vea su funcionalidad. (<https://openwebinars.net/blog/los-5-mejores-plugins-para-atom/> )
   1. Emmet
   2. Javascript Snippets
   3. Pigments
   4. Linter
   5. Comment
2. Instala os seguintes plugins para VisualStudio Code explicando a súa utilidade de forma moi breve. Aporta unha captura en onde se vexa a súa funcionalidade. (<https://carontestudio.com/blog/mejores-extensiones-para-visual-studio-code/> )

<https://medium.com/better-programming/how-to-use-vscode-like-a-pro-e120c428f45f>

* 1. Auto-close Tag
  2. Bracket pair colorizer
  3. ESlint
  4. [Open in browser](https://carontestudio.com/blog/mejores-extensiones-para-visual-studio-code/#Open_in_browser)
  5. [CSS Peek](https://carontestudio.com/blog/mejores-extensiones-para-visual-studio-code/#CSS_Peek)
  6. [Prettier](https://carontestudio.com/blog/mejores-extensiones-para-visual-studio-code/#Prettier)
  7. [Color info](https://carontestudio.com/blog/mejores-extensiones-para-visual-studio-code/#Color_info)
  8. [SVG](https://carontestudio.com/blog/mejores-extensiones-para-visual-studio-code/#SVG)
  9. [Vscode-icons](https://carontestudio.com/blog/mejores-extensiones-para-visual-studio-code/#Vscode-icons)
  10. [JS & CSS Minifier (Minify)](https://carontestudio.com/blog/mejores-extensiones-para-visual-studio-code/#JS_CSS_Minifier_Minify)
  11. [TODO Highlight](https://carontestudio.com/blog/mejores-extensiones-para-visual-studio-code/#TODO_Highlight)
  12. File templates for VSCode
  13. Live Reload

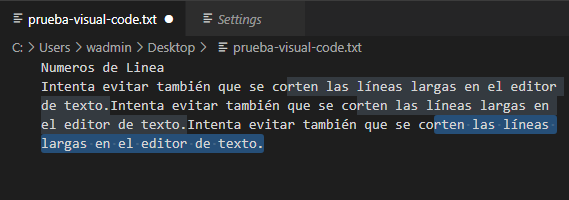
1. Después de ver estos editores, ¿qué características crees que son importantes en el trabajo diario de un programador?

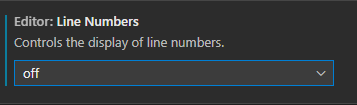
Un editor enfocado a la programación debe incluir características como: herramientas visuales que permiten facilitar el trabajo, herramientas de detección automática del código, ayudas a la escritura (como la coloración, el sangrado, la identificación de elementos…), atajos y herramientas de búsqueda.

También es importante que incluya la posibilidad de añadir plugins internos, herramientas de colaboración y de organización.

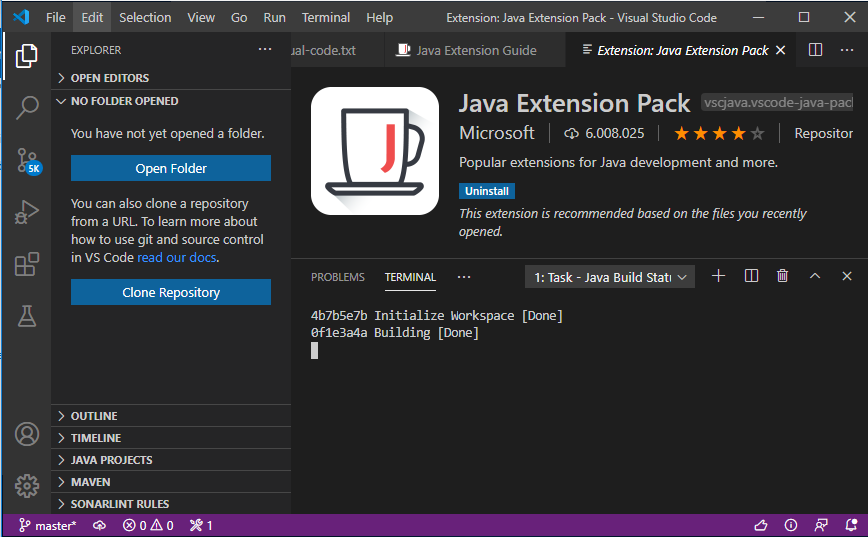
1. Intenta evitar que en VSCode se visualicen los números de línea. Intenta evitar también que se corten las líneas largas en el editor de texto.

Para evitar que se corten las líneas con Alt+Z





1. Configura VSCode para que podamos desarrollar programas con Java. Sigue las recomendaciones de esta página <https://code.visualstudio.com/docs/languages/java> y esta otra <https://code.visualstudio.com/docs/java/java-tutorial> pero teniendo en cuenta que vamos a instalar el **Java Extension Pack**. Activa el standard Mode para poder ejecutar y compilar de un modo oportuno.





(Manito arriba == Standard Mode)