

## Añadir Y Eliminar Módulos



<b>Autor</b>	Manuel Ávila Dugo
<b>Fecha</b>	09/11/2024
<b>Curso</b>	1ºDAM

# ÍNDICE

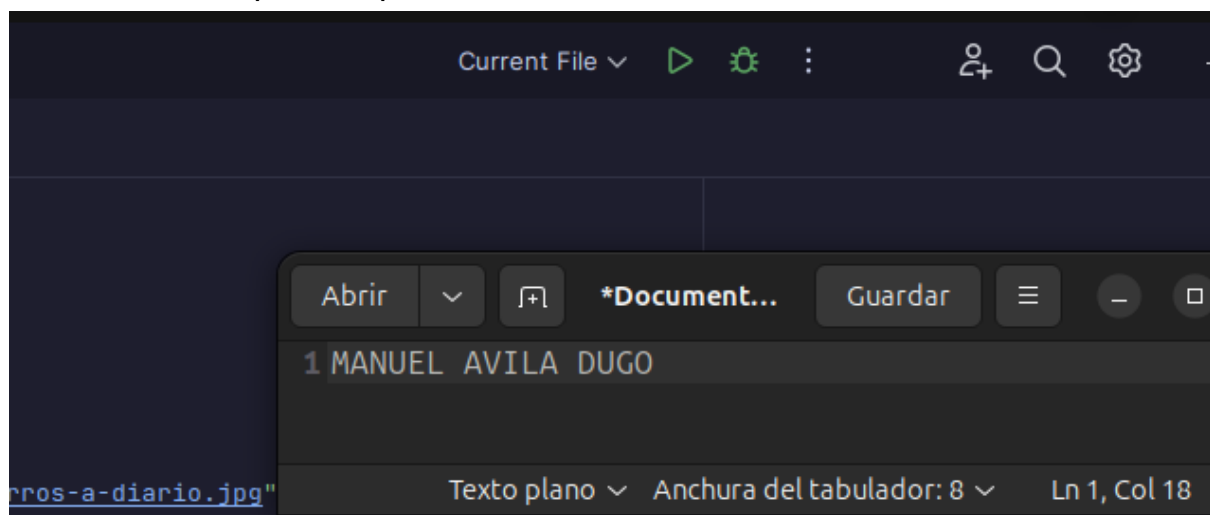
<b>1. Como instalar y modificar módulos en el IDE Intellij:</b>	<b>3</b>
1.1 Inicio:	3
1.2 Buscamos El Apartado de Project Structure:	4
<b>1.3 Instalación De Un Módulo:</b>	<b>5</b>
1.4 Deshabilitar El Módulo:	6
<b>2.Comparativa de los IDES:</b>	<b>6</b>

# 1. Como instalar y modificar módulos en el IDE IntelliJ:

Vamos a aprender a como instalar y modificar módulos en el IDE de intellij, pero para ello previamente tenemos que tener instalado este IDE instalado en nuestro ordenador y lo veremos de manera sencilla.

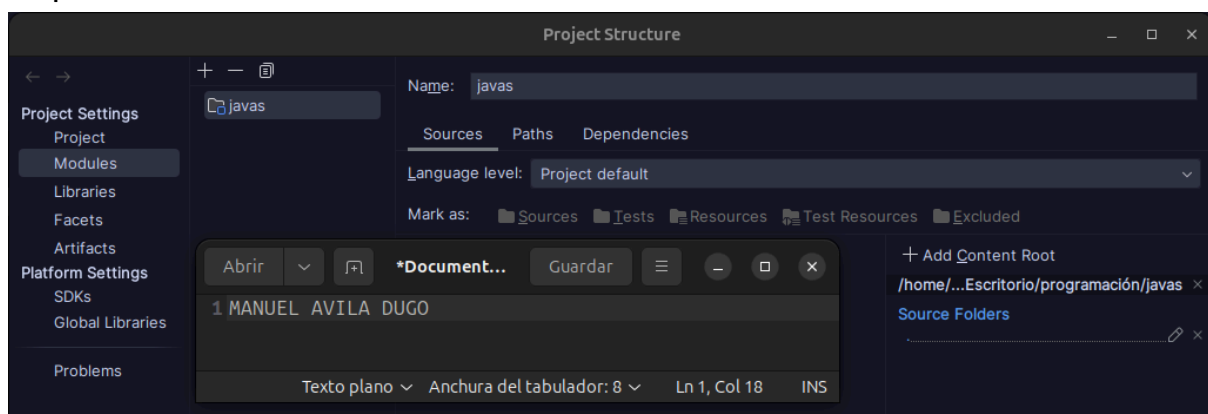
## 1.1 Inicio:

Para comenzar entraremos en nuestro IDE intellij y en la pantalla que se nos abre al principio vamos a buscar un icono de un engranaje, el cual se encuentra en la parte superior a la derecha.



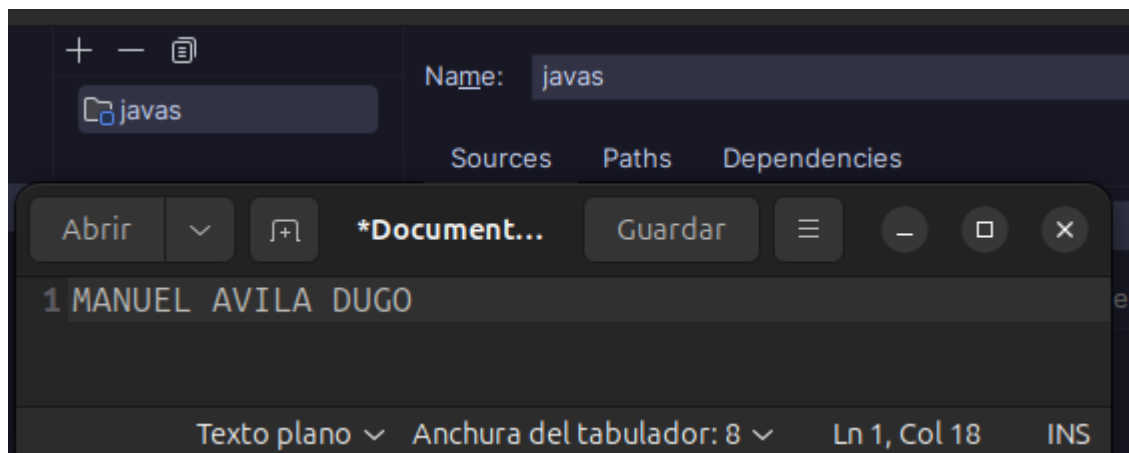
## 1.2 Buscamos El Apartado de Project Structure:

Ahora debemos de hacer click en el icono del engranaje y a continuación nos saldrá un pequeño menú en el que tendremos que buscar el apartado en el que ponga project structure y hacemos click sobre él, así ya podremos empezar a instalarlos.

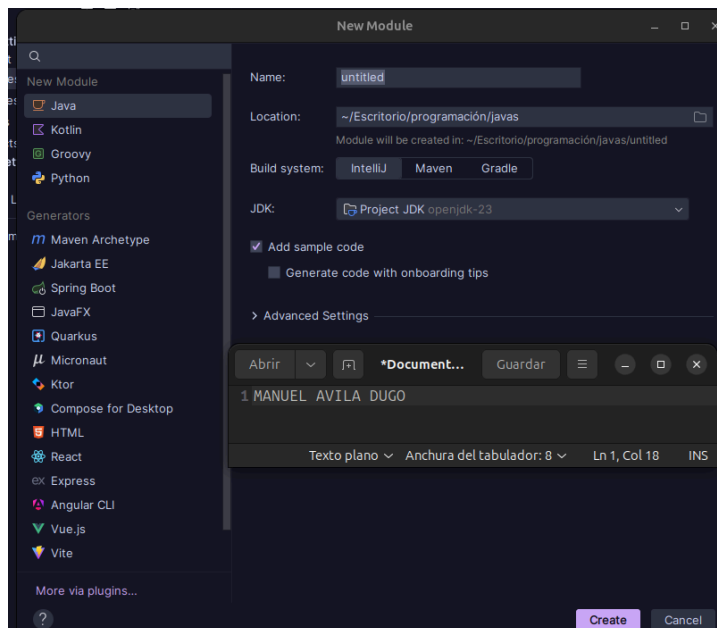


## 1.3 Instalación De Un Módulo:

Ahora en la parte de arriba del menú vas a ver un símbolo + el cual sirve para añadir el módulo que queramos, solamente debemos de hacer click y darle a new module.

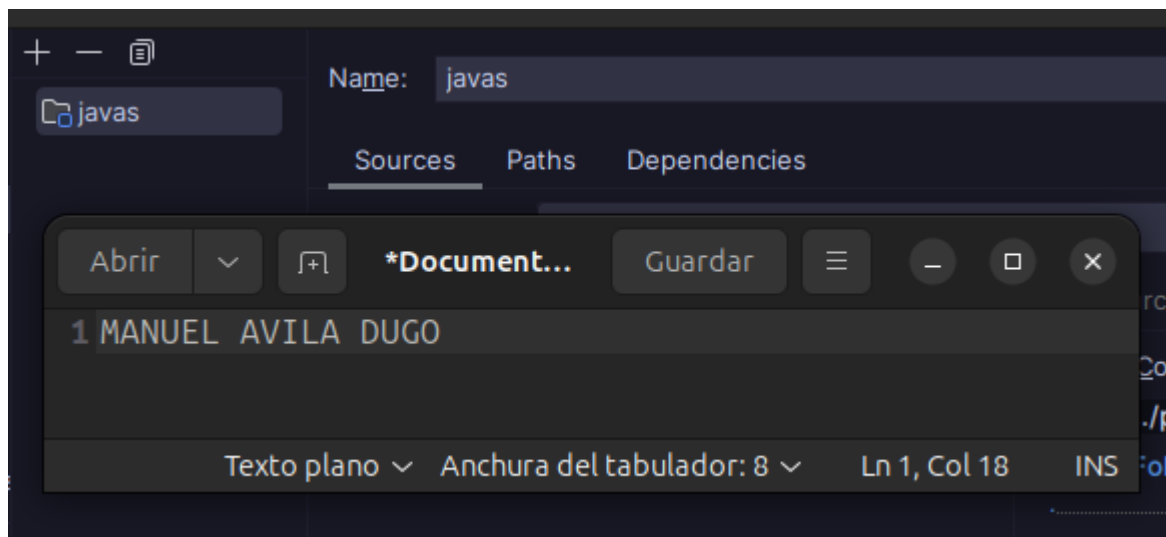


Y al clicar ahí podemos elegir el lenguaje en el que hacemos el módulo como por ejemplo java, Kotlin, etc...



## 1.4 Deshabilitar El Módulo:

Para eliminar el módulo hay que darle al símbolo - y ahí te dejaba eliminar el módulo seleccionado, o también dando click derecho en el módulo que deseas también te dará la opción de borrarlo.



## 2.Comparativa de los IDES:

Característica	IntelliJ IDEA	VS CODE
<b>Ventajas</b>	Excelente integración con herramientas de construcción y gestión de módulos para entornos JVM, interfaz gráfica completa para la gestión de módulos y dependencias, y gran soporte para proyectos de múltiples módulos.	Enfoque liviano y flexible para diversos lenguajes; las extensiones permiten personalización, y el terminal facilita la ejecución de comandos personalizados de cada gestor de paquetes.
<b>Desventajas</b>	Puede consumir más recursos en proyectos grandes y complejos debido a sus características avanzadas; limitado a ciertos lenguajes sin extensiones.	La gestión de módulos puede ser más manual y depende de las extensiones, lo que podría resultar en una menor integración y en menos automatización en proyectos complejos.
<b>Integración de repositorios</b>	IntelliJ IDEA permite vinculación con repositorios personalizados y locales, facilitando la gestión de dependencias específicas	VS Code depende en gran medida de la línea de comandos o extensiones para manejar repositorios personalizados, lo que

	para proyectos grandes.	puede complicar la administración avanzada.
<b>Soporte para proyectos múltiples</b>	Manejo avanzado de módulos en proyectos de múltiples capas, permitiendo agregar subproyectos y manejar dependencias entre ellos. La sincronización de dependencias puede ajustarse para cada módulo de forma independiente.	VS Code permite gestionar proyectos múltiples a través de carpetas de trabajo ("workspaces") y configuraciones en archivos <code>.json</code> . Aunque ofrece flexibilidad, la gestión de dependencias en proyectos complejos puede ser más manual.

En resumen:

**-IntelliJ IDEA:** es una opción ideal para desarrolladores que trabajan principalmente en el ecosistema Java o JVM y necesitan una interfaz gráfica avanzada para la gestión de dependencias. Su integración nativa con herramientas como Maven y Gradle permite una gestión de módulos más centralizada y automática.

**-Visual Studio Code:** es una herramienta más ligera y versátil, especialmente útil para proyectos en múltiples lenguajes y donde la flexibilidad es clave. Sin embargo, la gestión de módulos en VS Code depende en gran medida de la línea de comandos y de las extensiones, lo que puede requerir más configuración manual en comparación con IntelliJ.