

# Runtimes de Java en Visual Studio Code

Para usar `java.configuration.runtimes` en la configuración de Visual Studio Code (VS Code), debes configurarlo en tu archivo `settings.json`. Esta configuración te permite especificar múltiples versiones del Java Development Kit (JDK) que VS Code puede usar para tus proyectos de Java. Es especialmente útil cuando trabajas con diferentes proyectos que requieren versiones específicas de Java. Aquí tienes una guía paso a paso para configurarlo:

## Pasos para configurar `java.configuration.runtimes`

### 1. Abrir la configuración de VS Code:

- En Windows/Linux: Ve a `File > Preferences > Settings` o presiona `Ctrl + ,`.
- En macOS: Ve a `Code > Preferences > Settings` o presiona `Cmd + ,`.

### 2. Acceder al archivo de configuración JSON:

- En la interfaz de configuración, busca `java.configuration.runtimes`.
- Verás una opción como “Java: Configuration: Runtimes”. Haz clic en “Edit in settings.json” (generalmente un enlace debajo de la descripción de la configuración) para abrir el archivo `settings.json`.

### 3. Editar `settings.json`:

- En el archivo `settings.json`, agrega o modifica el array `java.configuration.runtimes`. Este array contiene objetos, cada uno representando una versión de JDK que deseas que VS Code reconozca.
- Cada objeto generalmente incluye:
  - `name`: El identificador de la versión de Java (por ejemplo, `JavaSE-1.8`, `JavaSE-11`, `JavaSE-17`).
  - `path`: La ruta absoluta al directorio de instalación del JDK en tu sistema.
  - `default` (opcional): Establece en `true` para hacer que este sea el JDK predeterminado para carpetas no gestionadas (proyectos sin herramientas de construcción como Maven o Gradle).

Aquí tienes un ejemplo de configuración:

```
{
  "java.configuration.runtimes": [
    {
      "name": "JavaSE-1.8",
      "path": "C:/Program Files/Java/jdk1.8.0_351",
      "default": true
    },
    {
      "name": "JavaSE-11",
```

```

        "path": "C:/Program Files/Java/jdk-11.0.15"
    },
    {
        "name": "JavaSE-17",
        "path": "C:/Program Files/Java/jdk-17.0.6"
    }
]
}

```

#### 4. Verificar las rutas del JDK:

- Asegúrate de que la `path` apunte al directorio raíz de tu instalación del JDK (por ejemplo, donde se encuentra la carpeta `bin` que contiene `java.exe` o `java`).
- En Windows, usa barras inclinadas hacia adelante (/) o barra invertida escapada (\\) en la ruta.
- En macOS/Linux, usa la ruta del sistema de archivos adecuada (por ejemplo, `/usr/lib/jvm/java-17-openjdk`).

#### 5. Guardar y recargar:

- Guarda el archivo `settings.json`.
- Reinicia VS Code o recarga la ventana (`Ctrl + R` o `Cmd + R`) para aplicar los cambios.

#### 6. Verificar la configuración:

- Abre la paleta de comandos (`Ctrl + Shift + P` o `Cmd + Shift + P`) y ejecuta el comando `Java: Configure Java Runtime`.
- Esto abre una vista que muestra los JDK disponibles para tus proyectos. Verifica que los tiempos de ejecución configurados aparezcan en la pestaña “Project JDKs”.

### Cómo funciona

- **Carpetas no gestionadas:** Para proyectos sin herramientas de construcción (por ejemplo, archivos Java planos), VS Code usa el JDK predeterminado especificado en `java.configuration.runtimes`.
- **Proyectos gestionados (Maven/Gradle):** Para proyectos con herramientas de construcción, la versión del JDK se determina por la configuración de construcción (por ejemplo, `pom.xml` o `build.gradle`), pero VS Code seguirá reconociendo los tiempos de ejecución listados aquí para la compatibilidad.
- **Servidor de lenguaje:** El Java Language Server (parte de las extensiones de Java) requiere JDK 17+ para ejecutarse. Puedes especificar esto por separado con `java.jdt.ls.java.home` si es necesario (por ejemplo, `"java.jdt.ls.java.home": "C:/Program Files/Java/jdk-17.0.6"`).

### Ejemplo con múltiples JDKs

Si estás en una Mac y tienes JDK 8 y 17 instalados a través de Homebrew, tu configuración podría verse así:

```

{
  "java.configuration.runtimes": [
    {
      "name": "JavaSE-1.8",
      "path": "/Library/Java/JavaVirtualMachines/jdk1.8.0_351.jdk/Contents/Home",
      "default": true
    },
    {
      "name": "JavaSE-17",
      "path": "/Library/Java/JavaVirtualMachines/jdk-17.0.6.jdk/Contents/Home"
    }
  ]
}

```

## Consejos

- Usa el formato correcto de name (JavaSE-<version>), coincidiendo con la versión de Java (por ejemplo, JavaSE-11 para Java 11).
- Si solo tienes un JDK, aún puedes listarlo para asegurarte de que VS Code lo use explícitamente.
- Para encontrar tu ruta del JDK, ejecuta `java -version` en una terminal y verifica el directorio de instalación (por ejemplo, a través de `where java` en Windows o `which java` en sistemas Unix, luego rastrea hacia atrás hasta la raíz del JDK).

Esta configuración te da la flexibilidad de trabajar con múltiples versiones de Java en VS Code sin problemas. ¡Hazme saber si necesitas ayuda para encontrar tus rutas del JDK o para solucionar problemas!