

Java 語言伺服器與 Maven 建構

聽起來你在 Visual Studio Code (VSCode) 遇到了兩個相關的問題：一個是需要至少 Java 21 的投訴，另一個是 VSCode 無法識別你的 Java 代碼中的某些對象定義，即使你的 Maven 建置成功。讓我們一步一步解決這些問題。這個問題出現是因為 VSCode 的 Java 擴展（特別是 Java 語言伺服器）和你的 Maven 建置在 Java Development Kit (JDK) 版本或項目配置上可能不一致。VSCode 表示需要 JDK 21 才能正常運行，而解析錯誤表明它沒有像 Maven 一樣解釋你的代碼。以下是如何修復它的方法：

第 1 步：安裝 JDK 21

由於 VSCode 明確要求 Java 21，你需要確保它已安裝在你的系統上。

- **下載 JDK 21**：從可信來源如 Oracle 的網站 或 Adoptium 賽取。
- **安裝它**：按照安裝說明進行安裝，並記下安裝路徑（例如，Windows 上的 C:\Program Files\Java\jdk-21 或 Linux 上的 /usr/lib/jvm/java-21-openjdk）。
- **驗證**：打開終端並運行 `java -version`。你應該會看到類似以下的內容：

```
java version "21.0.1" ...
```

第 2 步：配置 VSCode 使用 JDK 21

你需要告訴 VSCode 的 Java 擴展使用 JDK 21。以下是如何做到的：

1. 打開 VSCode 設定：

- 進入 File > Preferences > Settings (或按 `Ctrl + ,`)。

2. 設置 JDK 路徑：

- 搜尋 `java.jdt.ls.java.home`。
- 输入你的 JDK 21 安裝路徑（例如，C:\Program Files\Java\jdk-21）。
- 可選地，搜尋 `java.home` 並將其設置為相同的路徑以保持一致。

3. 編輯 `settings.json` (替代方法)：

- 打開命令面板 (`Ctrl + Shift + P`)，輸入 “Preferences: Open Settings (JSON)”，並添加：

```
{  
  "java.jdt.ls.java.home": "C:\\Program Files\\Java\\jdk-21",  
  "java.home": "C:\\Program Files\\Java\\jdk-21"  
}
```

- 根據你的系統調整路徑（Linux/Mac 使用正斜杠 /）。

這樣可以確保 VSCode 中的 Java 語言伺服器使用 JDK 21，滿足最低要求。

第 3 步：在 VSCode 中設置項目的 JDK

為了修復解析問題（例如，找不到對象定義），請確保你的 VSCode 項目也使用 JDK 21：

- 在 `settings.json` 中添加以下內容以指定 JDK 21 為你的項目：

```
"java.configuration.runtimes": [
  {
    "name": "JavaSE-21",
    "path": "C:\\Program Files\\Java\\jdk-21",
    "default": true
  }
]
```

- 這樣可以使 VSCode 的項目運行時與 JDK 21 一致，這應該有助於它正確解析你的代碼。
-

第 4 步：驗證 Maven 的 JDK 配置

由於你的 Maven 建置 (`mvn compile`) 運行正常，它很可能使用了一個兼容的 JDK。讓我們確認並與 VSCode 保持一致：

1. 檢查 Maven 的 JDK：

- 在終端中運行 `mvn -version`。查找 “Java version” 行（例如，`Java version: 21.0.1`）。
- 如果它沒有使用 JDK 21，將 `JAVA_HOME` 環境變量設置為你的 JDK 21 路徑：
 - **Windows** : `set JAVA_HOME=C:\\Program Files\\Java\\jdk-21`
 - **Linux/Mac** : `export JAVA_HOME=/usr/lib/jvm/java-21-openjdk`
- 再次運行 `mvn -version` 以確認。

2. 在 `pom.xml` 中指定 JDK（可選）：

- 如果你想在 Maven 項目中強制使用 JDK 21，將以下內容添加到你的 `pom.xml`：

```
<properties>
  <maven.compiler.source>21</maven.compiler.source>
  <maven.compiler.target>21</maven.compiler.target>
</properties>
```

第 5 步：刷新 VSCode

完成這些更改後，VSCode 需要刷新以應用它們：

- 重新加載 VSCode：
 - 打開命令面板 (Ctrl + Shift + P) 並選擇 “Developer: Reload Window”。
 - 或者，關閉並重新打開 VSCode。
-

第 6 步：確保依賴項已加載

如果 VSCode 仍然找不到對象定義，它可能沒有正確加載你的 Maven 依賴項：

1. 清理工作區：

- 打開命令面板 (Ctrl + Shift + P)。
- 運行 “Java: Clean Java Language Server Workspace”。

2. 強制編譯：

- 從命令面板運行 “Java: Force Java Compilation”。

3. 檢查導入：

- 確保你的 Maven 依賴項已導入，方法是右鍵點擊 VSCode 中的 pom.xml 並選擇 “Update Project Configuration”。
-

故障排除

如果問題仍然存在：
- **檢查日誌**：- 將 "java.trace.server": "verbose" 添加到 settings.json。- 打開輸出面板 (Ctrl + Shift + U) 並選擇 “Java Language Server” 以查看詳細日誌。
- **驗證代碼**：- 確保缺失的對象定義來自你的代碼或 pom.xml 中列出的依賴項，而不是外部問題。

摘要

通過安裝 JDK 21、配置 VSCode 使用它、對齊你的項目和 Maven 設置，並刷新環境，VSCode 應該不再投訴 Java 21 並正確解析你的代碼，與 Maven 的成功編譯匹配。如果你遇到任何問題，請告訴我！