

'Java Configuration Runtime' का उपयोग कैसे करें

java.configuration.runtimes को एक संस्करण में लिया जाता है (जो एक JSON) सेटिंग्स में उपयोग करने के लिए, आपको अपने settings.json फ़ाइल में इसे कॉन्फ़िगर करना होगा। यह सेटिंग आपको एक संस्करण को आपके एक प्रोजेक्ट के लिए उपयोग करने के लिए कई विभिन्न संस्करणों को स्पेसिफ़ाई करने की अनुमति देता है। यह विशेष रूप से अलग-अलग प्रोजेक्ट के साथ काम करने के लिए उपयोगी है जो विशिष्ट संस्करणों की आवश्यकता होती हैं। यहां एक स्टेप-बाय-स्टेप गाइड है इसे सेटअप करने के लिए:

java.configuration.runtimes को कॉन्फ़िगर करने के लिए चरण

1. एक सेटिंग्स खोलें:

- एक नए फ़ाइल/परफ़ूम पर: File > Preferences > Settings पर जाएं या Ctrl + , दबाएं।
- एक नए परफ़ूम पर: Code > Preferences > Settings पर जाएं या Cmd + , दबाएं।

2. सेटिंग्स फ़ाइल तक पहुंचें:

- सेटिंग्स एक में, java.configuration.runtimes के लिए खोजें।
- आप "नाम: संस्करण नाम: संस्करण" जैसा विकल्प देखेंगे। "संस्करण नाम संस्करण.ज़िप" (सामान्यतः सेटिंग्स विवरण के नीचे एक लिंक) पर क्लिक करें settings.json फ़ाइल को खोलने के लिए।

3. settings.json को संपादित करें:

- settings.json फ़ाइल में, java.configuration.runtimes एरे को जोड़ें या संपादित करें। इस एरे में वस्तुएं होती हैं, प्रत्येक एक संस्करण को एक संस्करण को पहचानने के लिए प्रतिनिधित्व करता है।
 - प्रत्येक वस्तु आमतौर पर शामिल होती है:
 - name: संस्करण पहचानकर्ता (उदाहरण के लिए, JavaSE-1.8, JavaSE-11, JavaSE-17).
 - path: आपके सिस्टम पर एक इंस्टॉलेशन डायरेक्टरी का पूर्ण पथ।
 - default (वैकल्पिक): इसे true सेट करें ताकि यह अनमैनेज्ड फ़ोल्डर्स (मैवेन या ग्रेडल जैसी बिल्ड टूल्स के बिना प्रोजेक्ट) के लिए डिफ़ॉल्ट एक बन जाए।

यहां एक उदाहरण कॉन्फ़िगरेशन है:

```
{  
  "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. यहाँ पथों की पुष्टि करें:

- path को इस इंस्टॉलेशन के रूट डायरेक्टरी (जहाँ bin फोल्डर java.exe या java है) पर सेट करें।
- यहाँ पर, पथ में फॉरवर्ड स्लैश (/) या बैकस्लैश (\) का उपयोग करें।
- यहाँ पर, उचित फ़ाइल सिस्टम पथ का उपयोग करें (उदाहरण के लिए, /usr/lib/jvm/java-17-openjdk).

5. सहेजें और पुनः लोड करें:

- settings.json फ़ाइल को सहेजें।
- यह फ़ाइल को पुनः शुरू करें या विंडो को पुनः लोड करें (Ctrl + R या Cmd + R) बदलाव लागू करने के लिए।

6. कॉन्फिगरेशन की जांच करें:

- कमांड पैलेट (Ctrl + Shift + P या Cmd + Shift + P) खोलें और कमांड Java: Configure Java Runtime चलाएं।
- यह एक दृश्य खोलता है जिसमें आपके प्रोजेक्ट्स के लिए उपलब्ध यहाँ दिखते हैं। पुष्टि करें कि आपके कॉन्फिगर किए गए रनटाइम “Java-17” टैब के नीचे दिखाई देते हैं।

यह कैसे काम करता है

- **अनमैनेज्ड फ़ोल्डर्स:** बिल्ड टूल्स के बिना प्रोजेक्ट्स (उदाहरण के लिए, साधारण यहाँ फ़ाइलें) के लिए, यहाँ java.configuration.runtimes में स्पेसिफ़ाई किया गया default यहाँ का उपयोग करता है।
- **मैनेज्ड प्रोजेक्ट्स (pom.xml/gradle.build):** बिल्ड टूल्स के साथ प्रोजेक्ट्स के लिए, यहाँ संस्करण बिल्ड कॉन्फिगरेशन (उदाहरण के लिए, pom.xml या build.gradle) द्वारा निर्धारित होता है, लेकिन यहाँ अभी भी यहाँ सूचीबद्ध रनटाइम्स को अनुकूलता के लिए पहचानेगा।
- **लैंगवेज सर्वर:** यहाँ लैंगवेज सर्वर (यहाँ एक्सटेंशन का हिस्सा) को चलाने के लिए 17+ की आवश्यकता होती है। अगर आवश्यक हो, इसे अलग से java.jdt.ls.java.home के साथ स्पेसिफ़ाई किया जा सकता है (उदाहरण के लिए, "java.jdt.ls.java.home": "C:/Program Files/Java/jdk-17.0.6").

कई यहाँ के साथ उदाहरण

अगर आप यहाँ पर हैं और 17 इंस्टॉल किए हैं, तो आपके कॉन्फिगरेशन इस तरह दिख सकता है:

```

{
    "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"
}
]
}

```

टिप्प

- सही name फॉर्मेट (JavaSE-<version>) का उपयोग करें, जो १००० संस्करण (उदाहरण के लिए, १००० 11 के लिए JavaSE-11) के साथ मिलता है।
- अगर आपके पास केवल एक १००० है, तो आप इसे सूचीबद्ध करके १० १००० को इसे स्पेसिफिक रूप से उपयोग करने के लिए सुनिश्चित कर सकते हैं।
- १००० पथ को खोजने के लिए, एक टर्मिनल में java -version चलाएं और इंस्टॉलेशन डायरेक्टरी (उदाहरण के लिए, १००००००० पर where java या १००००-१०००० सिस्टम पर which java, फिर १००० रूट तक वापस ट्रेस करने के लिए) की जांच करें।

यह सेटअप आपको १० १००० में कई १००० संस्करणों के साथ काम करने की सुविधा देता है! मुझे बताएं अगर आपको १००० पथों को खोजने या ट्रॉलशूटिंग करने में मदद चाहिए।