Troubleshooting rJava Errors - A Guide
Spatial Data Analysis in R - James Tsalah

The Problem: rJava notoriously gets confused by multiple factors regarding the interface between R & Java, so you are almost guaranteed to get some troubling errors at first run.

There are three main issues you can run in to:
1. You don't have Java installed on your computer. Download the 64 bit version!
   - **Do not just go to https://www.java.com/en/download/ this just downloads the 32 bit version of java we do not want.**
   - Instead go to the manual downloads page @ https://www.java.com/en/download/manual.jsp

   Download Windows Offline (64 bit), as the other two versions are 32 bit.

   - If you are on Mac or Linux find the appropriate 64 bit download of Java

2. The Java version you downloaded is not the same version as your RStudio architecture (by default 64 bit).
   - RStudio typically downloads an 64 bit application, but Java by default installs as an 32 bit application! You can see where this is going…
   - rJava cannot interface your R session with Java because they have incompatible application architectures by default (64 bit vs 32 bit).
   - Refer to the installation tips to get the 64 bit version of Java above.

3. You have the correct 64 bit Java version installed, but rJava still isn't working! This is likely because rJava is quite bad at finding your Java installation…
    - Navigate to your Program Files folder and locate Java, note that it can be in either your Program Files or Program Files (x86) folder.
        - C:\Program Files\Java
        - C:\Program Files (x86)\Java
    - Now find the folder within Java which houses the version of Java you have
        - Note: jre-1.8 is the current version of Java as of 5/3/24
        - C:\Program Files\Java\jre-1.8
        - C:\Program Files (x86)\Java\jre-1.8
    - You can now specify this location to rJava with this line of code! Replace the Java installation path with what is correct for your machine.
        - Sys.setenv(JAVA_HOME='C:/Program Files/Java/jre-1.8')
    - Finally we can load in rJava successfully!
        - require(rJava)