

What is the directory structure of maven project?

java project directory structure:  
project (root directory)  
|-src  
  |-main  
    |-java (java source directory)  
    |-.java  
    |-resources (non-java source directory)  
      |-.properties|.xml|.json|.yaml  
  |-test  
    |-java  
    |-resources  
|-pom.xml (dependencies/libraries: jars)  
|-target  
  |-classes  
  |-\*.class  
|-project packaged (.jar|.war|.ear)

dependencies:  
\$MAVEN\_HOME/.m2 (local maven repository)  
|-repository

What is a classpath?  
The CLASSPATH is an environment variables that points to the location of an  
1. directory (that contains .class files)  
2. or to a jar (that has .class files packaged inside it)

so that the javac (java compiler) or java (jvm runtime engine) will use this environment variable to locate the classfiles or jar files in compiling or running the application.

Note: The default CLASSPATH is always “.” (present working directory)

What is the sourcecode of an application?  
SourceCode is the code that is required to run application and needs to be shipped to the enduser.  
There are 2 types of sourcecode

1. java sourcecode       = .java files are java sourcecode, that should be compiled to use/execute

2. non-java sourcecode = .properties, .xml, .json, .yaml etc that holds the configuration information that acts as an input for the programs inorder to execute are also called as sourcecode, but since these dont have to be compiled to use, these are called non-java sourcecode.

end of the day both java/non-java sourcecode is required to run the application, which means both must be available under the classpath of the project.

because we package or run the application from the classpath of the project.