

➤ Supplier interface

⌘ It is a *functional interface* that provides a **get** method to create something.

➤ To download a particular version of a library: <https://mvnrepository.com/> this can be used.

⌘ Select the version you want to download.

⌘ Then select the build tool for which you need this. (in my case Maven)

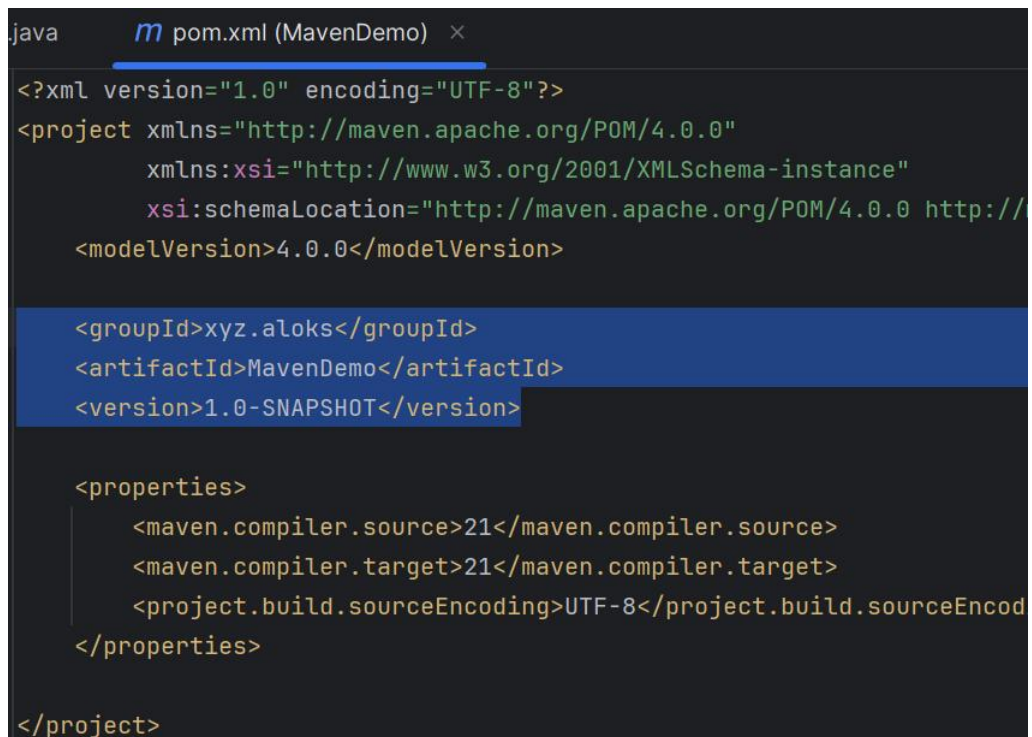
➤ POM: Project Object Model

⌘ Every library contains 3 things:

⌘ Group id (it should be unique across the globe)

⌘ Artifact id

⌘ Version



```
java    m pom.xml (MavenDemo) ×

<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://
    <modelVersion>4.0.0</modelVersion>

    <groupId>xyz.aloks</groupId>
    <artifactId>MavenDemo</artifactId>
    <version>1.0-SNAPSHOT</version>

    <properties>
        <maven.compiler.source>21</maven.compiler.source>
        <maven.compiler.target>21</maven.compiler.target>
        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    </properties>

</project>
```

⌘ When I clicked on **package** (right side of IntelliJ IDEA IDE), one **.jar** file got created.

⌘ MavenDemo-1.0-SNAPSHOT.jar

⌘ It contains the **artifact id** followed by the **version**.

⌘ Create one block **<dependencies></dependencies>** and paste all the dependency you want.

```

<groupId>xyz.aloks</groupId>
<artifactId>MavenDemo</artifactId>
<version>1.0-SNAPSHOT</version>

<dependencies>
  <dependency>
    <groupId>com.mysql</groupId>
    <artifactId>mysql-connector-j</artifactId>
    <version>8.1.0</version>
  </dependency>
</dependencies>

```

```

v External Libraries
  > < 21 > C:\Program Files\Java\jdk-21
  > Maven: com.google.protobuf:protobuf-jav
  > Maven: com.mysql:mysql-connector-j:8.1.
  Scratches and Consoles

```

* Now you can see, the *mysql-connector* got downloaded.

```

v Dependencies
  v com.mysql:mysql-connector-j:8.3.0
    com.google.protobuf:protobuf-java:3.25.1
  v org.hibernate.orm:hibernate-core:7.1.0.Final
    jakarta.persistence:jakarta.persistence-api:3.2.0
    jakarta.transaction:jakarta.transaction-api:2.0.1
    org.jboss.logging:jboss-logging:3.6.1.Final (runtime)
  > org.hibernate.models:hibernate-models:1.0.1 (runtime)
    com.fasterxml:classmate:1.7.0 (runtime)
    net.bytebuddy:byte-buddy:1.17.6 (runtime)
  > jakarta.xml.bind:jakarta.xml.bind-api:4.0.2 (runtime)
  > org.glassfish.jaxb:jaxb-runtime:4.0.5 (runtime)
    jakarta.inject:jakarta.inject-api:2.0.1 (runtime)
    org.antlr:antlr4-runtime:4.13.2 (runtime)

```

* You can see the dependencies here.

* For my-sql, only one **.jar** file got downloaded, but in case of **hibernate**, so may; it depends upon the package; every package needs its own set of dependencies.

^ If you want to add **plugins**, you can add.

^ NOTE:

^ Dependencies : the code needs to run; used at compile/run time

^ Plugins : tools/maven uses to build/manage; used at build time

- ⌘ But one thing is, **Maven** doesn't see the **pom.xml** file, rather it sees the **Effective POM** file.
- ⌘ In intellij: right click on pom.xml file >> Maven >> Show effective pom