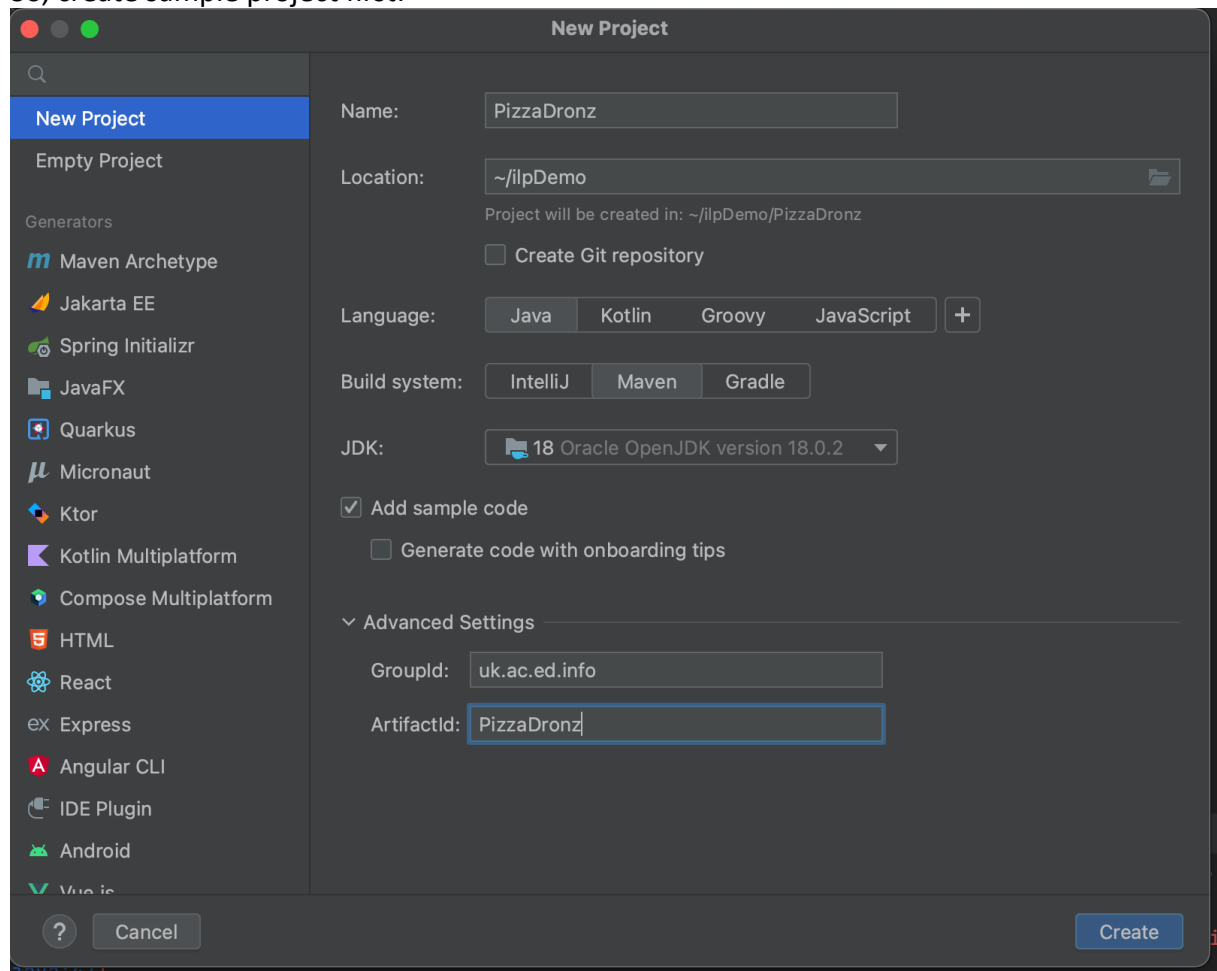


Using IlpDataObjects.jar

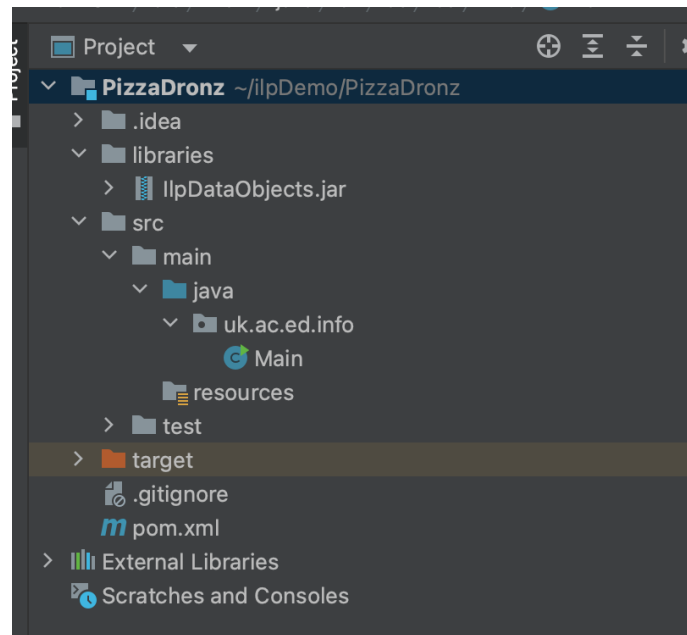
As IlpDataObjects.jar is a jar-file **without (!)** a default entry point it cannot be run directly but must be used from another application (or jar) to be useful.

In addition, as it is a downloaded file, many systems (Windows, Mac) might prevent the jar file from being executed if it remains in the download-folder as this could insert viruses into the system. Therefore, it is important to move the file to a location which is considered “safe”, which is usually the case within a sub-folder in your project.

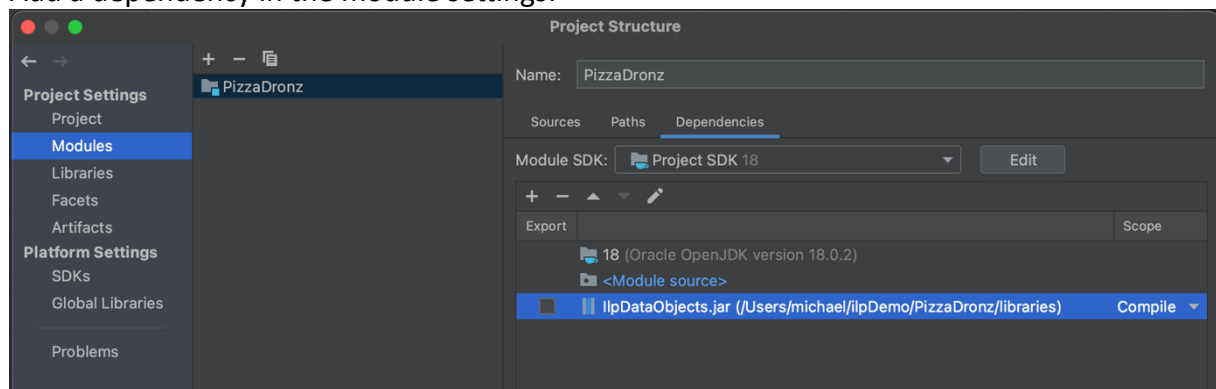
So, create sample project first:



Add the jar file into a sub-directory:

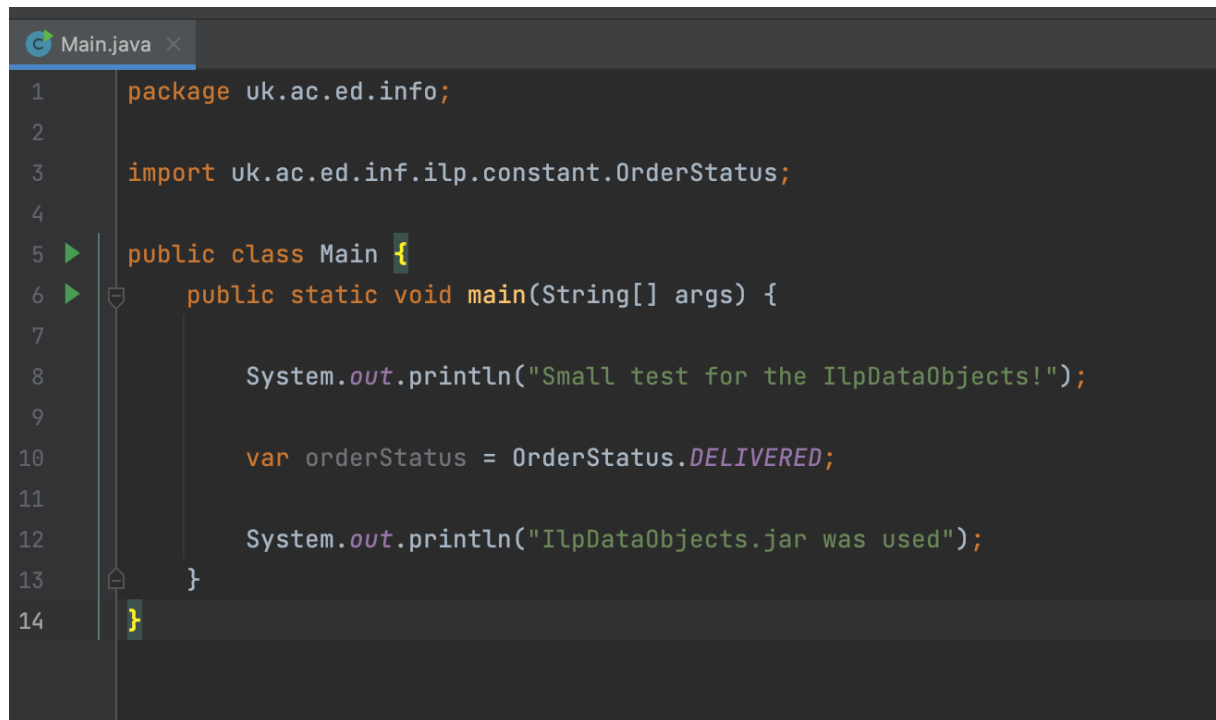


Add a dependency in the module settings:



Normally you would use Maven for these dependencies, yet then we would have to publish IlpDataObjects.jar to the maven-repository, or you would have to update your local cache. Each of these approaches is not too nice, so the local jar is the easiest to use.

Create a small sample and import the packages needed:



```
1 package uk.ac.ed.info;
2
3 import uk.ac.ed.inf.ilp.constant.OrderStatus;
4
5 public class Main {
6     public static void main(String[] args) {
7
8         System.out.println("Small test for the IlpDataObjects!");
9
10        var orderStatus = OrderStatus.DELIVERED;
11
12        System.out.println("IlpDataObjects.jar was used");
13    }
14 }
```

If you run the application (already if it compiles actually) IlpDataObjects.jar was loaded and accessed – so proof was given that the loading works.