

MANUAL EJERCICIO 1: GENERAR NÚMEROS ALEATORIOS ORDENADOS

1. Ejecución de la aplicación Aleatorios

- **Descripción:** La aplicación aleatorios genera una lista de 40 números enteros aleatorios entre 0 y 100.
- **Pasos:**

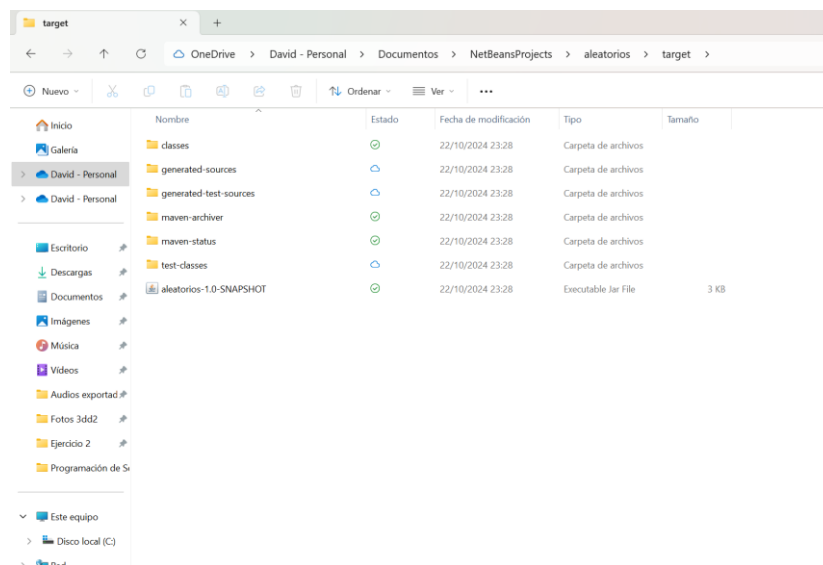
1.1 Crear el archivo JAR:

Para ello, en el apartado de proyectos de NetBeans, seleccionaríamos con el botón derecho la aplicación y pulsaríamos Clean and Build.

Pero antes de esto, debemos agregar la siguiente configuración en el archivo POM.xml del proyecto para crear un JAR ejecutable:

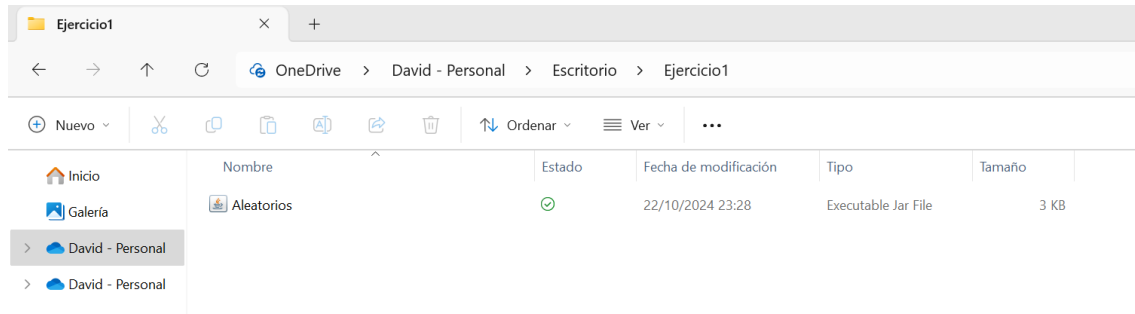
```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.
<modelVersion>4.0.0</modelVersion>
<groupId>com.mycompany</groupId>
<artifactId>aleatorios</artifactId>
<version>1.0-SNAPSHOT</version>
<packaging>jar</packaging>
<properties>
<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
<maven.compiler.release>21</maven.compiler.release>
<exec.mainClass>com.mycompany.aleatorios.Main</exec.mainClass>
</properties>
<build>
<plugins>
<plugin>
<!-- Build an executable JAR -->
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-jar-plugin</artifactId>
<version>3.1.0</version>
<configuration>
<archive>
<manifest>
<mainClass>com.mycompany.aleatorios.Main</mainClass>
</manifest>
</archive>
</configuration>
</plugin>
</plugins>
</build>
</project>
```

Esto generará el archivo .jar en la carpeta target del proyecto:

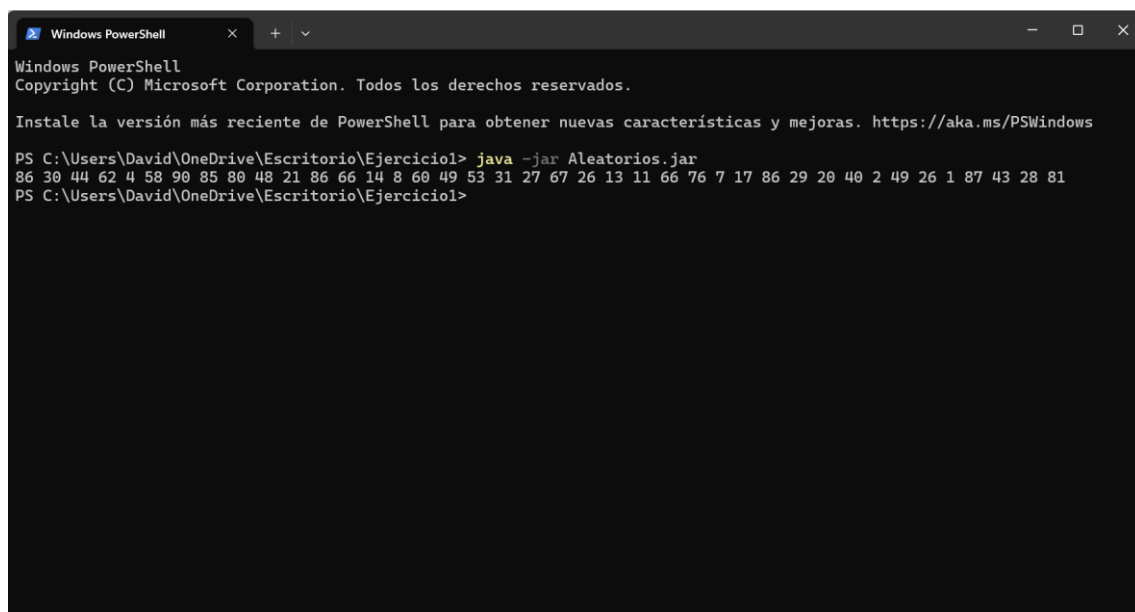


1.2 Ejecutar la aplicación:

Para ejecutar la aplicación, copiamos el archivo aleatorios.jar en la carpeta ejercicio 1.



Desde esa carpeta abrimos la terminal y comprobamos el funcionamiento de la aplicación con la siguiente línea de comandos: `java -jar aleatorios.jar`



De esta manera obtenemos una lista de 40 números aleatorios entre el 1 y el 100 separados por espacios.

2. Ejecución de la Aplicación ordenarNumeros

- **Descripción:** La aplicación ordenarNumeros lee una lista de números de la entrada estándar, los ordena y los imprime.
- **Pasos:**

2.1 Crear el archivo JAR:

Para ello, en el apartado de proyectos de NetBeans, seleccionaríamos con el botón derecho la aplicación y pulsaríamos Clean and Build.

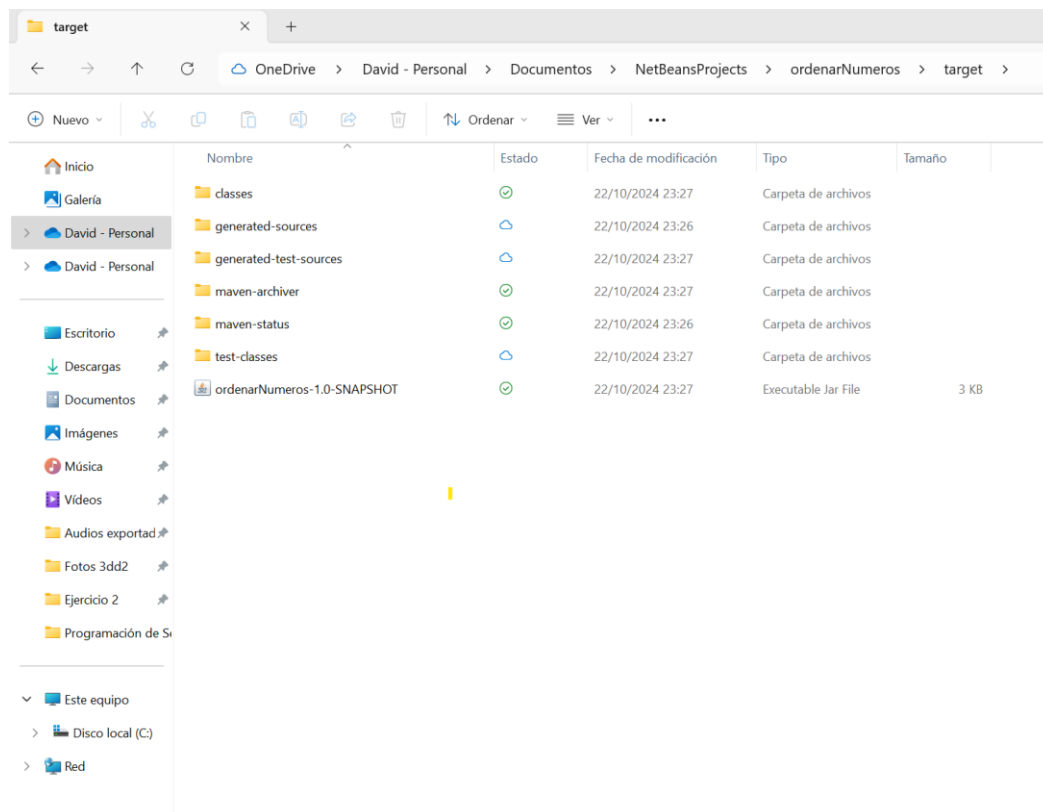
Pero antes de esto, debemos agregar la siguiente configuración en el archivo POM.xml del proyecto para crear un JAR ejecutable:

```

<?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.
<modelVersion>4.0.0</modelVersion>
<groupId>com.mycompany</groupId>
<artifactId>ordenarNumeros</artifactId>
<version>1.0-SNAPSHOT</version>
<packaging>jar</packaging>
<properties>
<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
<maven.compiler.release>21</maven.compiler.release>
<exec.mainClass>com.mycompany.ordenarNumeros.Main</exec.mainClass>
</properties>
<build>
<plugins>
<plugin>
<!-- Build an executable JAR -->
<groupId>org.apache.maven.plugins</groupId>
<artifactId>maven-jar-plugin</artifactId>
<version>3.1.0</version>
<configuration>
<archive>
<manifest>
<mainClass>com.mycompany.ordenarNumeros.Main</mainClass>
</manifest>
</archive>
</configuration>
</plugin>
</plugins>
</build>
</project>

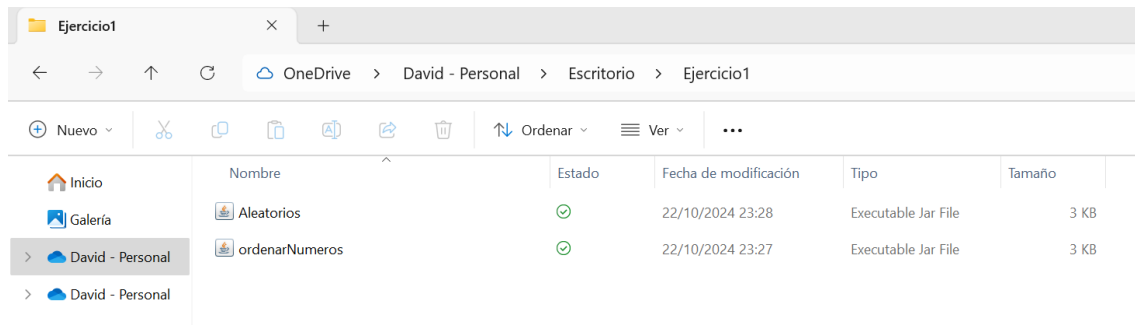
```

Esto generará el archivo .jar en la carpeta target del proyecto:



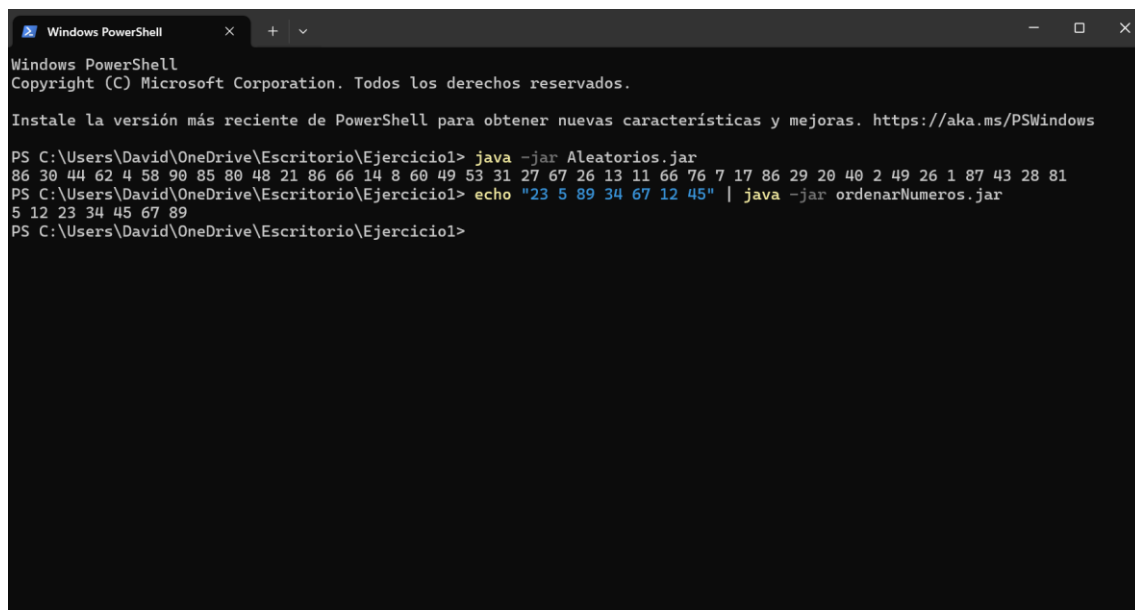
2.1 Ejecutar la aplicación:

Para ejecutar la aplicación, copiamos el archivo ordenarNumeros.jar en la carpeta ejercicio 1.



Desde esa carpeta abrimos la terminal y comprobamos el funcionamiento de la aplicación dando valores de números por la entrada estándar y usando el operador `|` para redirigir esta entrada con la aplicación ordenarNumeros con la siguiente línea de comandos:

```
echo "23 5 89 34 67 12 45" | java -jar ordenarNumeros.jar
```



De esta manera obtenemos en la salida los números dados por la entrada ordenados de manera ascendente.

3. Redirigir la salida de aleatorios a la entrada de ordenarnumeros

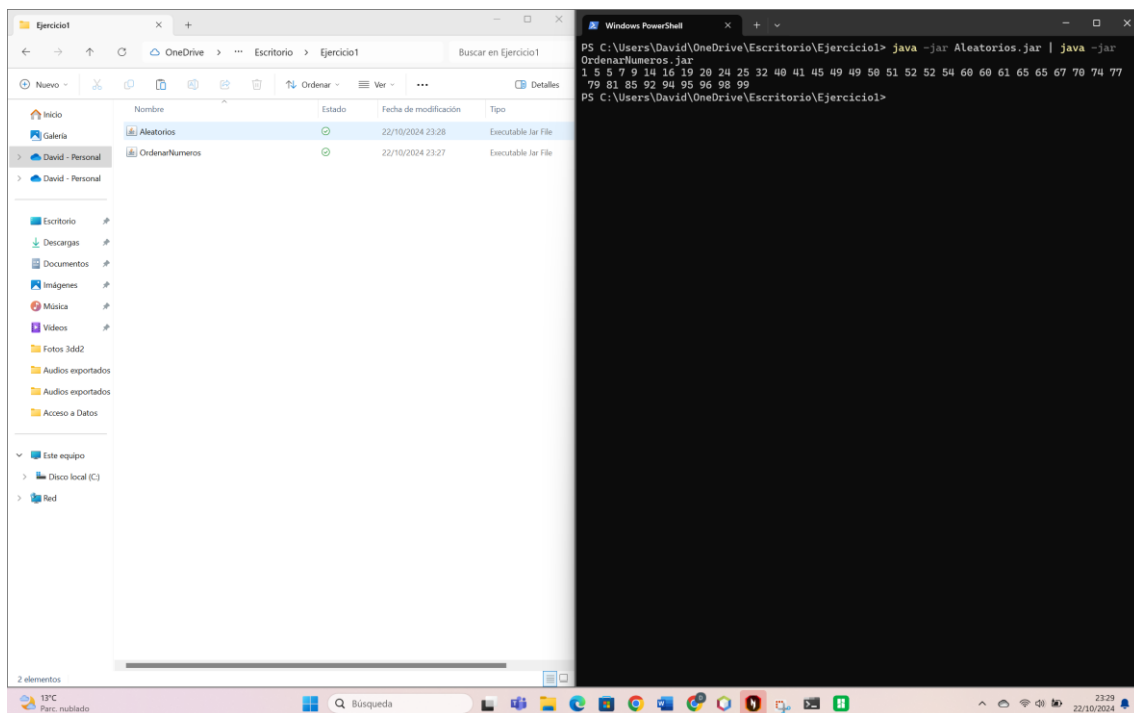
Abrimos la terminal en la carpeta Ejercicio 1 y usamos el operador | para redirigir la salida de Aleatorios como entrada de ordenarNumeros.

Ejecutar ambos JARs con la tubería con el siguiente código:

```
java -jar aleatorios.jar | java -jar ordenarnumeros.jar
```

Explicación:

- java -jar aleatorios.jar genera una lista de números aleatorios.
- | (pipe) redirige esa lista como entrada a java -jar ordenarnumeros.jar, que ordena y muestra los números.



Estas aplicaciones trabajando conjuntamente nos muestran una lista de 40 números ordenados de menor a mayor.

4. Conclusiones

- Las aplicaciones Aleatorios y ordenarNumeros pueden trabajar de forma independiente o de manera conjunta usando una tubería para generar y ordenar números.
- Esto permite que Aleatorios actúe como un generador de datos y ordenarNumeros como un procesador, una técnica útil en el procesamiento de datos en línea de comandos.