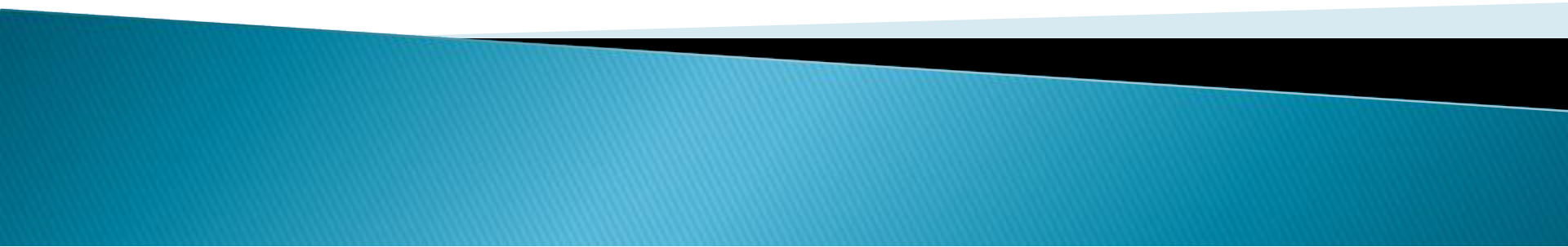


Empaquetado de módulos



Empaquetado en archivos .jar

➤ Para empaquetar un módulo en un .jar debemos hacer uso del comando:

```
jar -c -file=dir_destino/nombre_archivo.jar -C path_modulo .
```

➤ *dir_destino* es el directorio de destino del módulo *path_modulo* el directorio raíz del módulo. El punto “.” indica que se incluya todo el contenido del directorio:

```
jar -c --file=ejecutables/modulo2.jar -C modulo2 .
```

```
jar -c --file=ejecutables/modulo1.jar -C modulo1 .
```

El directorio
ejecutables debe
existir

➤ Para ejecutar modulo1 desde *ejecutables*:

```
java -p . -m modulo1/com.cliente.Test
```

Empaquetado en archivos .jmod

➤ Similar a .jar, si bien debe ser utilizado para librerías nativas en lugar de módulos en general:

➤ *Sintaxis:*

```
jmod create --class-path dir_modulo fichero.jmod
```

➤ **Ejemplo:**

```
jmod create --class-path modulo2 ejecutables/modulo2.jmod
```

```
jmod create --class-path modulo1 ejecutables/modulo1.jmod
```

➤ No se puede utilizar el formato .jmod en la ejecución de módulos

Comando jdeps

➤ Se emplea para obtener información sobre la dependencia de módulos:

```
jdeps -s dir_modulo/nombre_modulo
```

➤ Si depende de algún módulo extra:

```
jdeps --module-path mod_dependiente -s dir_modulo/nombre_modulo
```

➤ Ejemplo:

```
jdeps --module-path ejecutables/modulo2.jar -s ejecutables/modulo1.jar
```

➤ Resulta:

```
modulo1 -> java.base  
modulo1 -> modulo2
```