

EJERCICIO GUIADO. JAVA: POO. REUTILIZACIÓN DE CÓDIGO

Reutilización de código

La reutilización de código consiste en aprovechar lo que ya tenemos programado (así como aprovechar lo que han programado otros) para así poder finalizar los proyectos en menos tiempo.

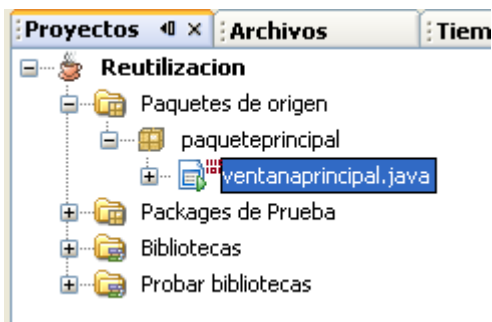
La filosofía de esto es: *¿por qué programar lo que ya está programado?*

A lo largo de la vida profesional de un programador, este se va haciendo con un conjunto de Clases que son capaces de generar objetos que pueden ser usados en los nuevos proyectos que el programador realice. Cuantas más clases colecciona, más herramientas dispondrá para crear los nuevos proyectos.

En esta hoja guiada veremos algunos ejemplos de como podemos aprovechar una Clase ya creada en un nuevo proyecto.

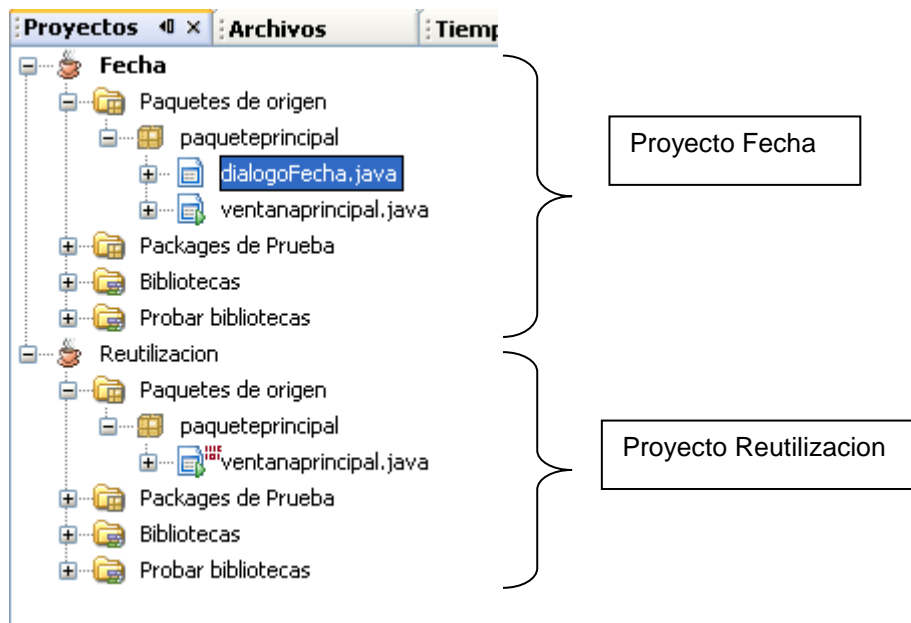
EJERCICIO GUIADO 1

1. Crea un nuevo proyecto. El nombre del proyecto será *Reutilizacion*. Contendrá un paquete llamado *paqueteprincipal* y dentro de él introducirás un JFrame al que llamaremos *ventanaprincipal*:



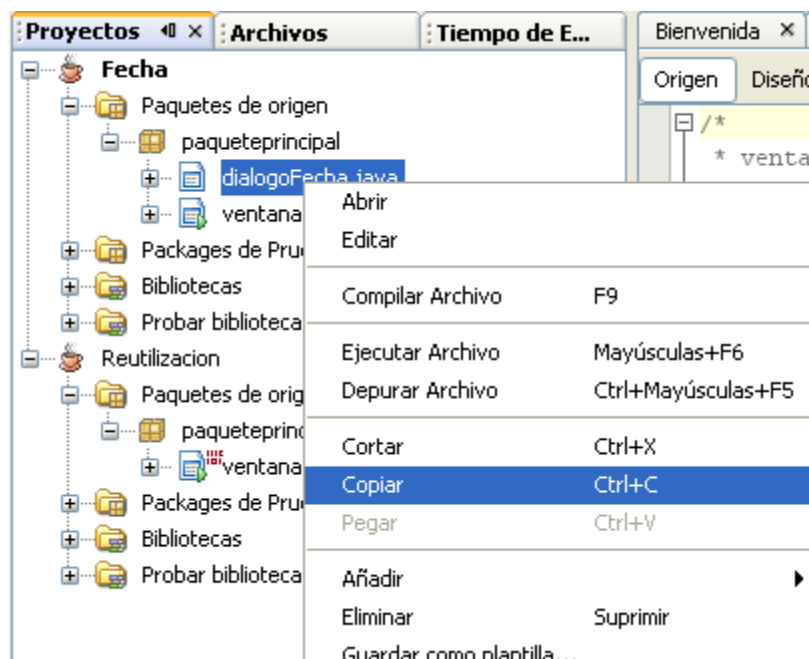
2. Supongamos que en este proyecto queremos utilizar cuadros de diálogo para introducir fechas. Podemos aprovechar la clase *dialogoFecha* creada en la hoja guiada anterior. Para ello, abre el proyecto llamado *Fecha* que hicimos en la hoja anterior.

Si despliegas el proyecto *Fecha*, tu ventana de proyectos tendrá un aspecto similar al siguiente:

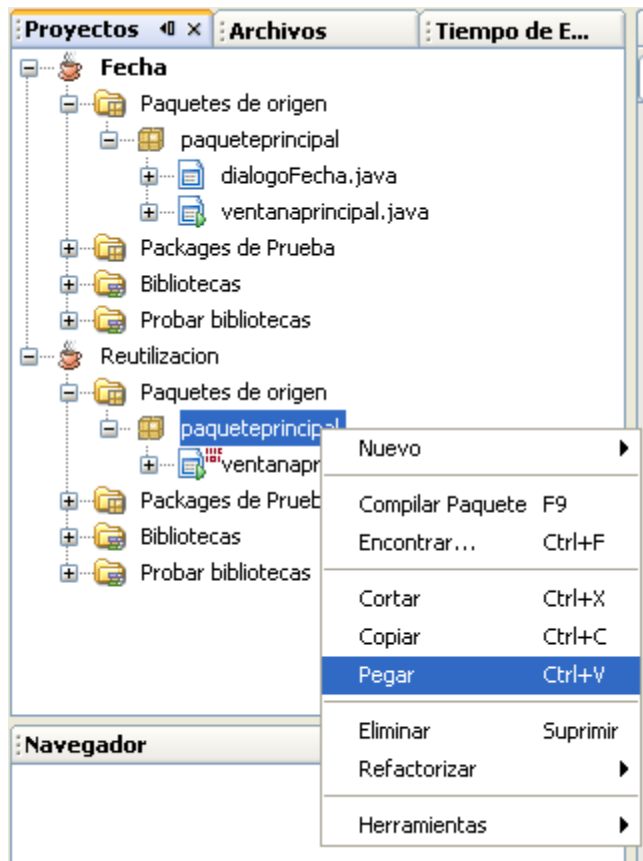


Observarás que el proyecto *Fecha* incluye la clase *dialogoFecha* dentro de su paquete principal.

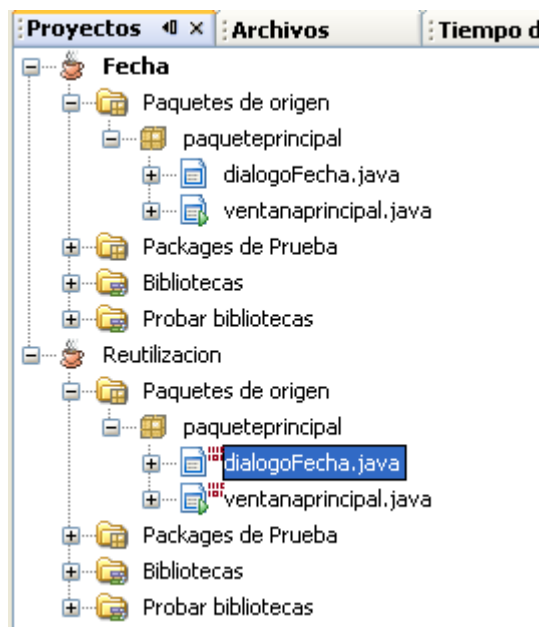
3. Para poder aprovechar la clase *dialogoFecha* en el proyecto *Reutilización*, solo hay que copiar el fichero de la clase desde un proyecto a otro. Así pues Copia *dialogoFecha*...



4. Y luego Pega en el paquete principal del proyecto *Reutilizacion*.

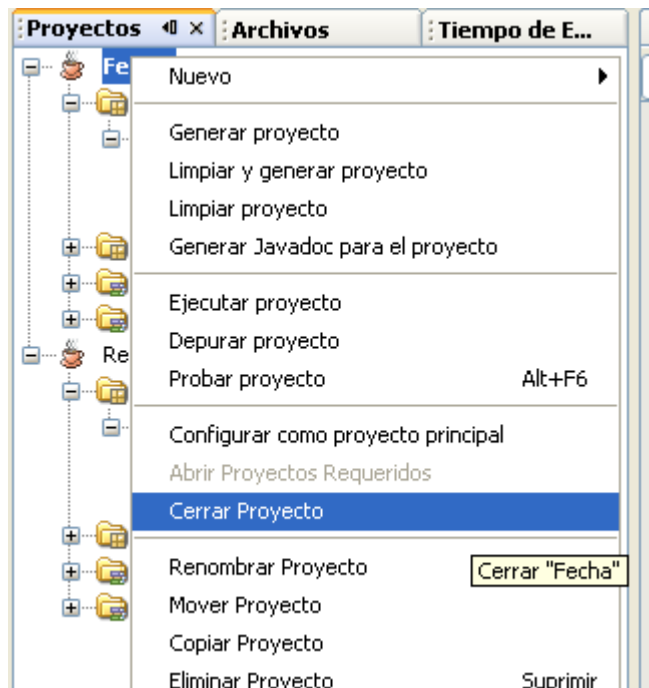


4. Observarás como se añade una copia de *dialogoFecha* al paquete principal del proyecto *Reutilizacion*.

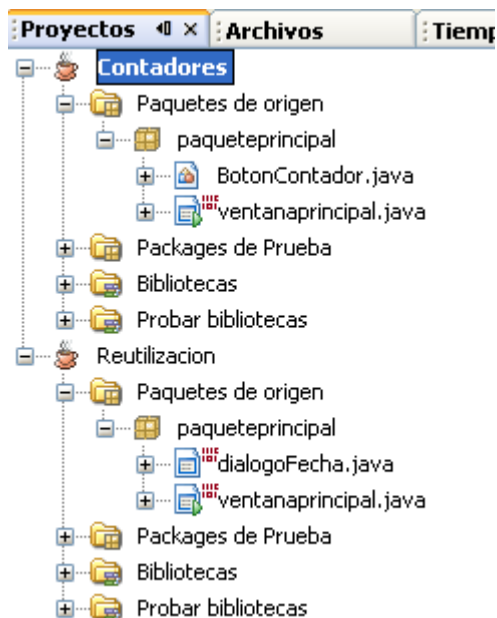


A partir de ahora, podrás crear objetos de tipo *dialogoFecha* dentro del proyecto *Reutilizacion* y aprovechar sus características, sin tener que programar ningún código adicional.

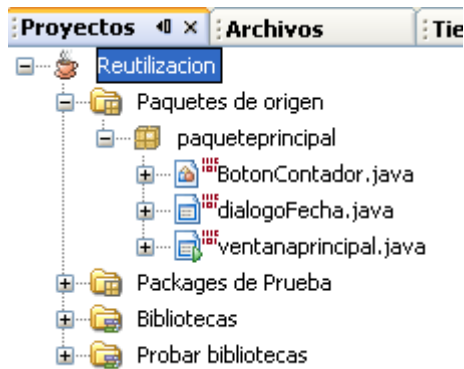
5. Cierra el paquete *Fecha* para despejar un poco la ventana de Proyectos (Botón derecho cerrar proyecto)



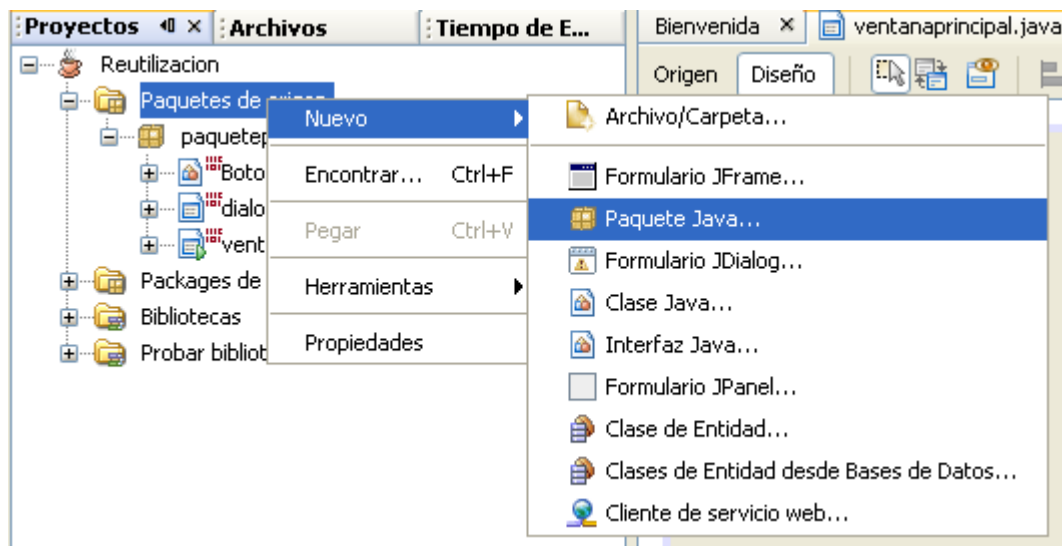
6. Supongamos que necesitamos también la clase *BotonContador* creada en la hoja guiada nº 31. Solo tendremos que abrir el proyecto llamado *Contadores*, que se programó en dicha hoja...



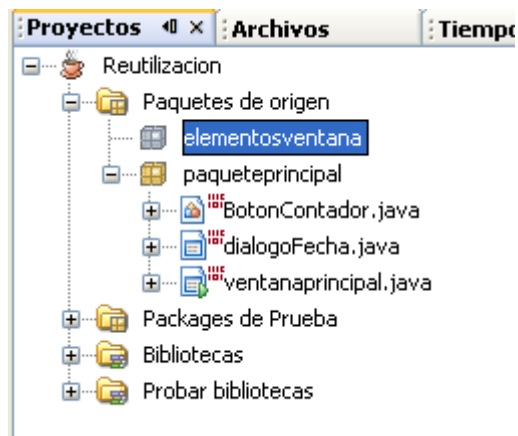
7. Y luego copiar la clase *BotonContador* desde *Contadores* a *Reutilizacion*. Hágalo y luego cierre la clase *Contadores*. Su ventana de proyectos debería quedar así:



8. A partir de este momento, puede crear botones contadores y cuadros de diálogo de introducción de fechas en el proyecto *Reutilizacion* sin tener que haber programado ni una línea de código.
9. Cuando tenemos proyectos de gran envergadura, es normal tener gran cantidad de ficheros de clases. Se hace necesario entonces organizarlos de alguna manera. Esto se hace a través de la creación de paquetes.
10. Para ver esta idea en nuestro proyecto *Reutilizacion*, crearemos un nuevo paquete (botón derecho sobre *Paquetes de origen* y luego *Nuevo* → *Paquete Java*)



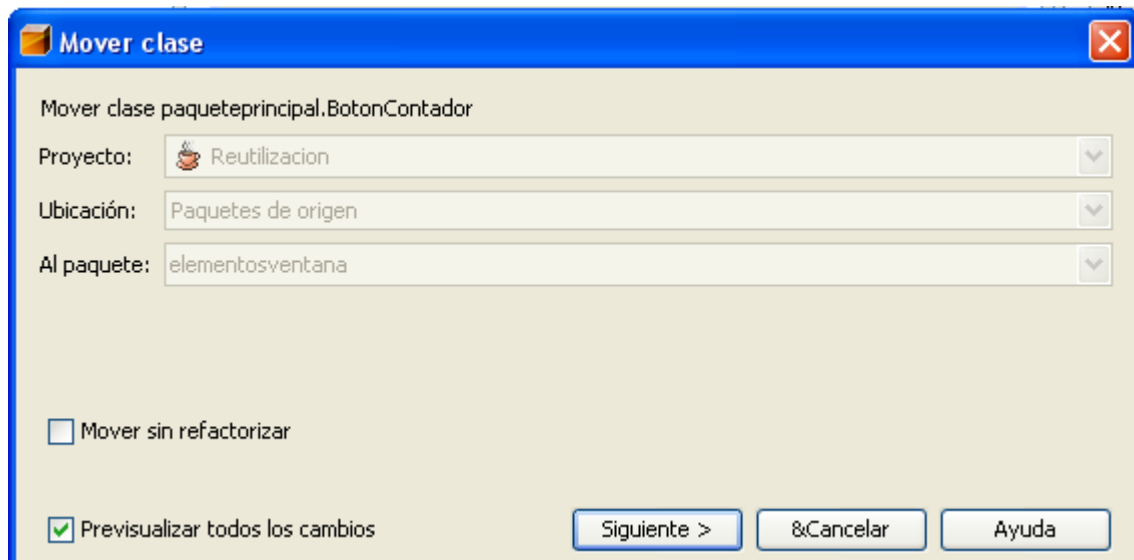
11. El nombre del paquete será *elementosventana*. Su proyecto quedará así:



12. En el nuevo paquete que hemos creado, introduciremos las clases *BotonContador* y *dialogoFecha*. De esta forma, tendremos organizadas nuestras clases. Por un lado la clase de la ventana principal, en el *paqueteprincipal*, y por otro lado, los elementos de las ventanas de creación propia en el paquete *elementosventana*.

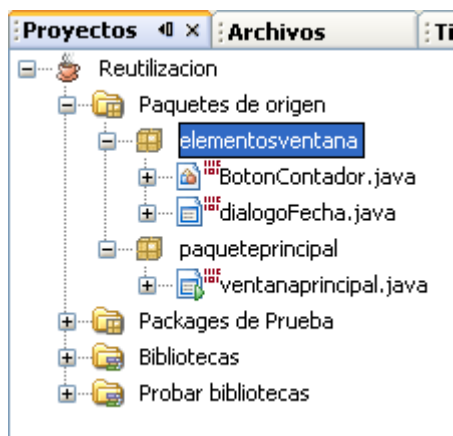
Para introducir las clases en el paquete *elementosventana* puede hacerlo cortando y pegando.

Cuando intente pegar una clase en otro paquete aparecerá un cuadro de opciones como el siguiente:



Desactiva la opción *Previsualizar todos los cambios* y pulsa *Siguiente*.

13. Una vez movidas las dos clases al paquete *elementosventana*, su proyecto debe quedar así:



Ahora tiene mejor clasificadas sus clases en su proyecto.

14. Guarde el proyecto, pero no lo cierre, ya que se usará a continuación.

EJERCICIO GUIADO 2

En el ejercicio guiado 1 hemos visto como podemos añadir a nuestro proyecto clases que se hayan programado en otros proyectos, de forma que no tengamos que volverlas a programar.

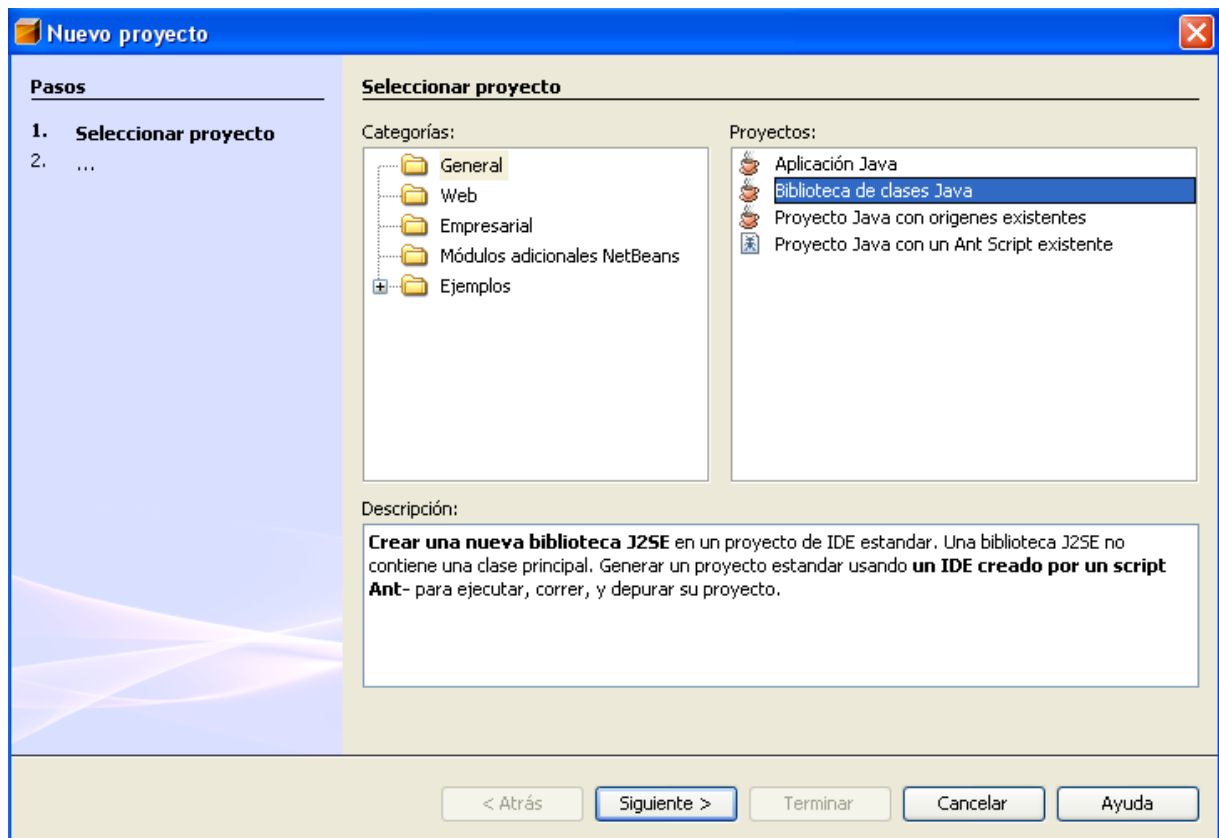
El problema que plantea lo anterior, es que tenemos que acordarnos de donde fue programada la clase que quiero añadir al proyecto. Es decir, tengo que acordarme del proyecto que contiene la clase.

Sería muy interesante que nuestras clases las tengamos reunidas en un solo proyecto, para no tener que estar luego buscándolas por distintos proyectos.

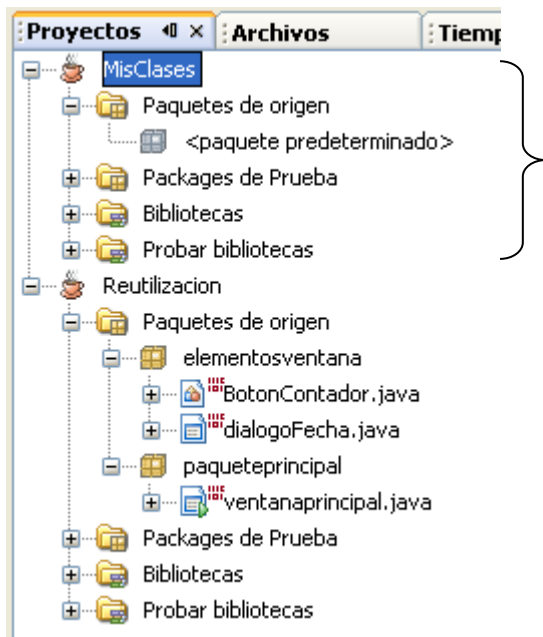
Una Biblioteca en Java es un proyecto que contiene una serie de paquetes que a su vez contienen distintas clases que son normalmente usadas en nuestros proyectos. Al estar reunidas estas clases en un mismo sitio, son más fáciles de encontrar.

Veremos en este ejercicio guiado como crear una Biblioteca.

1. Para crear una Biblioteca activa la opción *Archivo* → *Nuevo Proyecto*, y en la ventana de opciones que aparece elige *General* y *Biblioteca de clases java*.

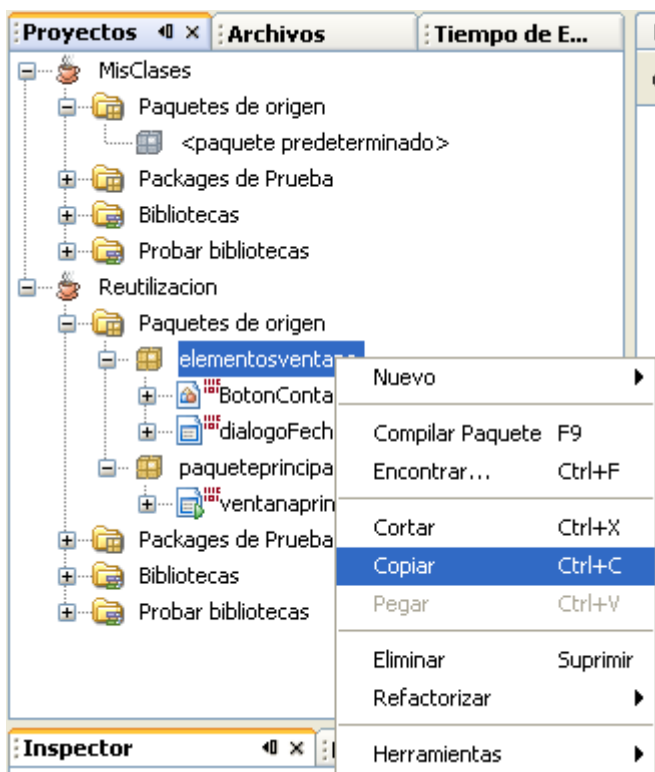


2. Pulsa el botón *Siguiente* y asigna a la biblioteca el nombre *MisClases*. Pulsa *Terminar*.

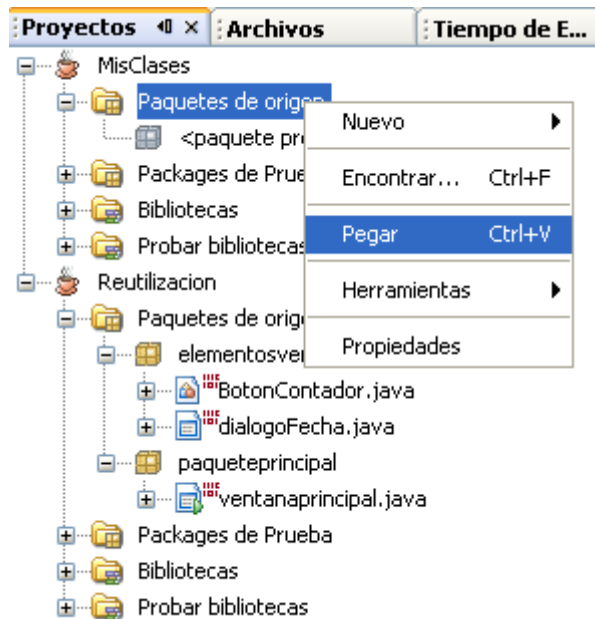


Observarás que ha aparecido un nuevo proyecto, llamado *MisClases*.

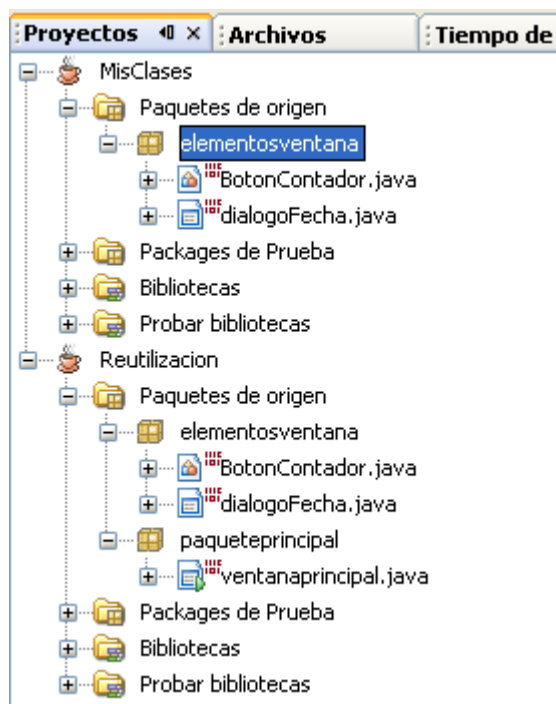
3. A este nuevo proyecto le agregaremos el paquete *elementosventana* que tenemos en el proyecto *Reutilizacion*. Solo hay que copiar....



4. Y luego Pegar...



5. Y el resultado es que hemos añadido un paquete a la biblioteca *MisClases*:



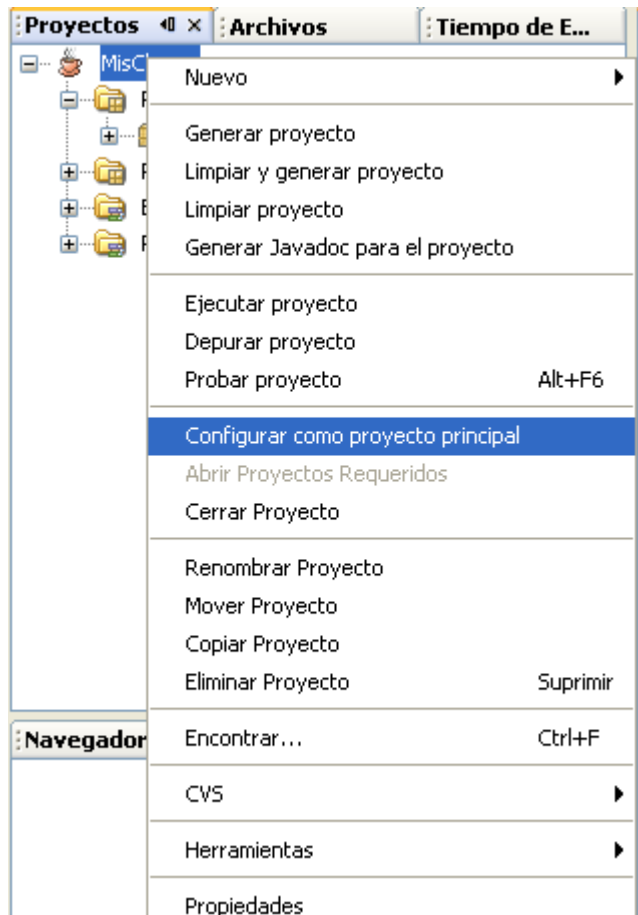
6. Se podría agregar más paquetes de clases a la biblioteca. A medida que vayamos programando clases útiles para nuestro trabajo, deberíamos incluirlas en la biblioteca.

Se consigue así tener todas las clases reunidas en un solo proyecto.

7. Cierre el proyecto *Reutilizacion*, pero no el proyecto *MisClases* porque lo usaremos a continuación.

EJERCICIO GUIADO 3

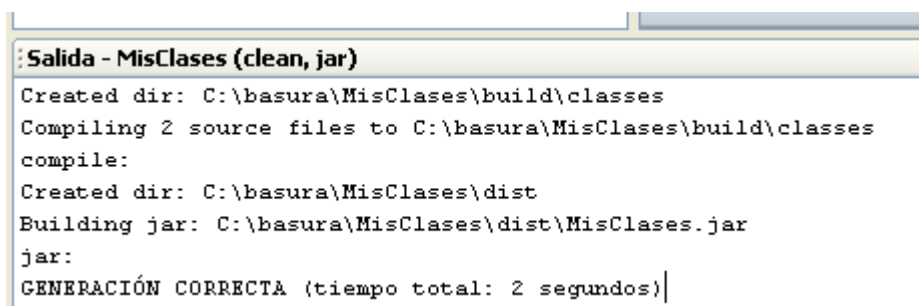
1. Para facilitar la inclusión de una biblioteca en un proyecto, se suele compactar la biblioteca en un fichero de tipo JAR. Para conseguir esto, configura el proyecto *MisClases* como proyecto principal.



2. A continuación pulsa el botón de la barra de herramientas *Limpiar y Construir proyecto principal*:



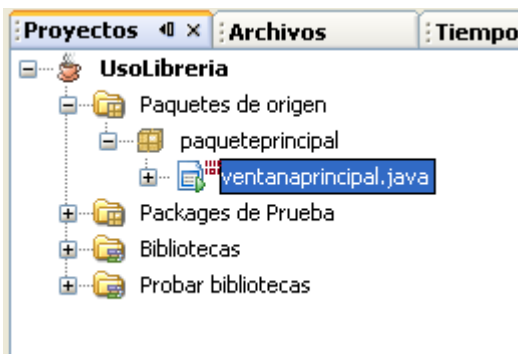
3. Al pulsar este botón, NetBeans compilará las distintas clases y buscará errores en ellas. Si no encuentra ninguno mostrará un mensaje en la parte inferior de la pantalla donde se indicará que la generación ha sido correcta:



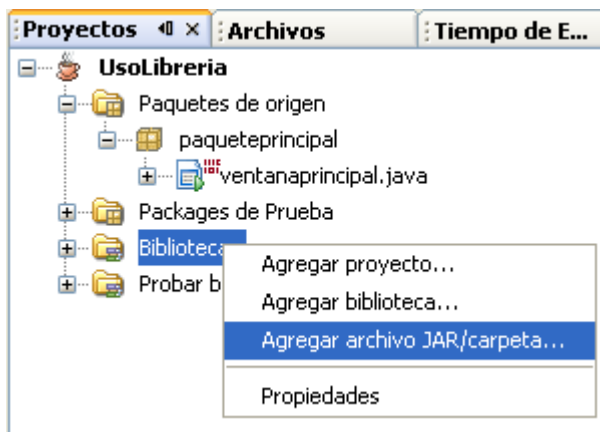
4. Si todo ha ido bien, se debe haber generado un fichero llamado `MisClases.JAR` que contiene toda la librería. Este fichero se encuentra en una carpeta llamada `DIST` que a su vez se encuentra dentro de la carpeta del proyecto.
5. Entra en MiPC y busca la carpeta `DIST`, que se encontrará dentro de la carpeta del proyecto `MisClases`. Dentro de esa carpeta verás el fichero `MisClases.JAR`. Copia el fichero `MisClases.JAR` al escritorio.

Un programador ordenado guardará todos sus ficheros JAR de biblioteca en una carpeta que tendrá a mano. Para este ejemplo, hemos copiado el fichero `MisClases.JAR` en el escritorio.

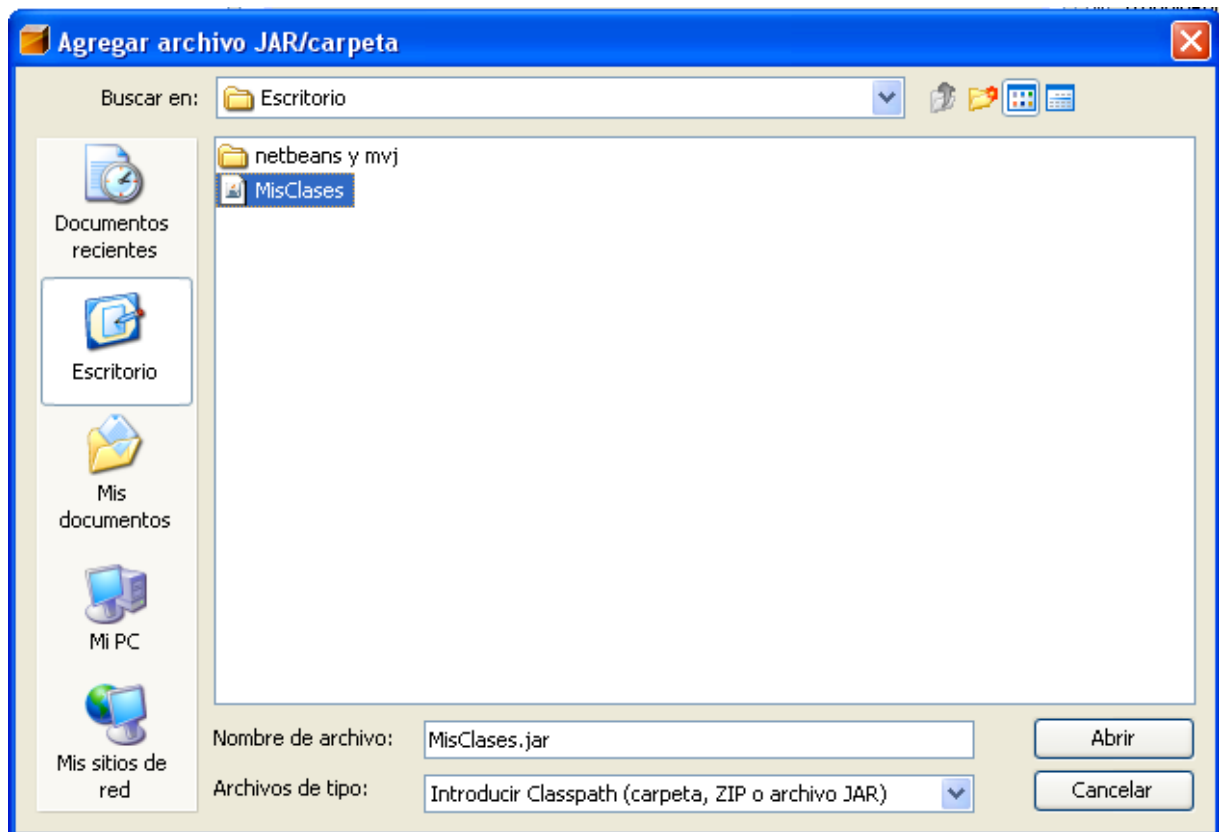
6. Dentro del NetBeans, cierre el proyecto *MisClases*.
7. Crea un nuevo proyecto, que se llamará *UsoLibreria*. Dentro de él añade un *paquete principal* y dentro del paquete principal un `JFrame` llamado *ventanaprincipal*. El aspecto de su ventana de proyecto será el siguiente:



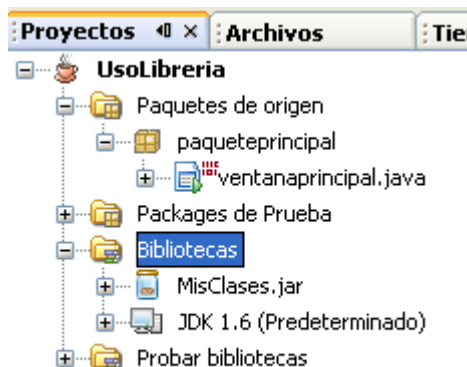
8. Imagine que queremos usar en este proyecto la librería anteriormente creada y compactada. Para ello, haz clic con el botón derecho sobre el elemento del proyecto llamado *Bibliotecas* y activa la opción *Agregar Archivo JAR / Carpeta*.



9. Busca en el escritorio el fichero *MisClases.JAR* y ábrelo.



10. En este momento, la librería de clases llamada *MisClases* ha sido añadida a tu proyecto, y esto significa que podrás usar cualquier clase que contenga *MisClases*. Es decir, podrás usar la clase *BotonContador* y *dialogoFecha* para crear tus objetos, así como cualquier otra clase que estuviera incluida en *MisClases*.



11. Resumiendo:

- Un programador crea muchas clases propias a lo largo de su trabajo.
- Estas clases pueden ser usadas en otros proyectos.
- Es conveniente reunir las clases en paquetes, y reunir paquetes en bibliotecas (también llamadas librerías)
- Las bibliotecas se pueden compactar en un solo fichero .JAR
- Al incluir un fichero de biblioteca en un proyecto, puedes usar las clases que estén contenidas en dicha biblioteca.

CONCLUSIÓN

En NetBeans puedes abrir varios proyectos y copiar los ficheros de clases de uno a otro.

Si un proyecto tiene muchas clases, interesa clasificarlas en paquetes.

Es interesante reunir las clases que más uses en un solo proyecto de tipo biblioteca, para poder acceder a ellas fácilmente.

Una biblioteca se puede compactar en un solo fichero de tipo .JAR

Al asociar un fichero de biblioteca .JAR a un proyecto, hacemos que en dicho proyecto se puedan usar las clases contenidas en dicha biblioteca.

Todas estas ideas hacen efectiva la *reutilización de código*, es decir, el aprovechar lo ya programado, para no tener que volverlo a programar.