



APRENDERAPROGRAMAR.COM

¿QUÉ ES UN PROYECTO  
JAVA? ORGANIZACIÓN DE  
ARCHIVOS .JAVA, .CLASS Y  
OTROS. CÓDIGO FUENTE Y  
CÓDIGO MÁQUINA.  
(CU00615B)

Sección: Cursos

Categoría: Curso “Aprender programación Java desde cero”

Fecha revisión: 2029

**Resumen:** Entrega nº15 curso Aprender programación Java desde cero.

Autor: Alex Rodríguez

## ¿QUÉ ES UN PROYECTO JAVA? ORGANIZACIÓN DE ARCHIVOS .JAVA, .CLASS Y OTROS

Un proyecto Java podemos considerarlo como una serie de carpetas ordenadas y organizadas de acuerdo con **una lógica para mantener organizado el código**. Un proyecto suele constar de archivos .java, archivos .class y documentación.

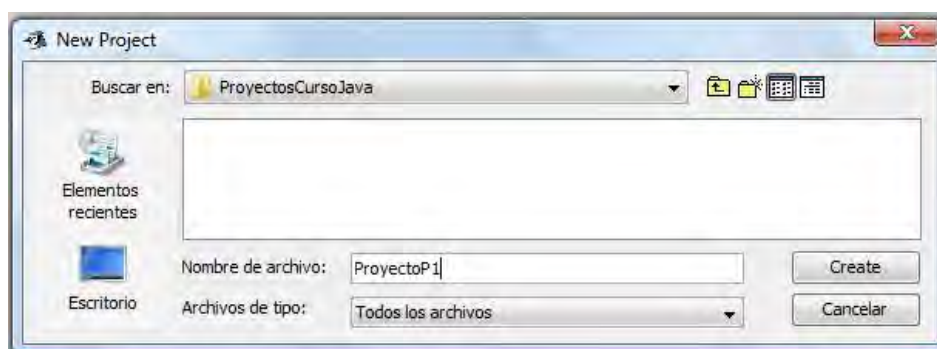


Los archivos .java contienen el código fuente (entendible por humanos) que en un momento dado podemos modificar con un editor de textos y suelen encontrarse en carpetas de nombre src (source). Los archivos .class contienen el bytecode (no entendible por humanos pero sí por la máquina virtual Java) y suelen encontrarse en carpetas de nombre bin (binary).

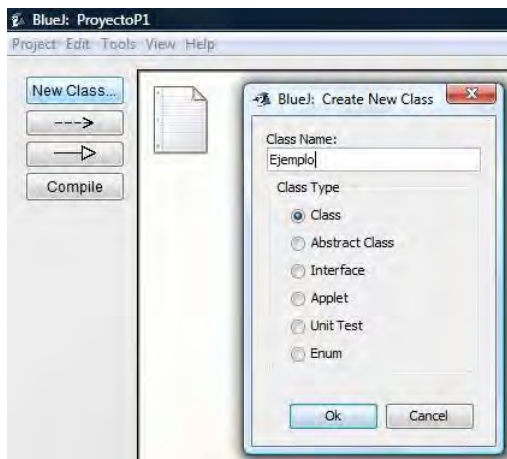
La organización de los archivos en carpetas y la presencia de otros adicionales depende del entorno de desarrollo que utilicemos. Además, Java introduce un esquema organizativo a través de paquetes (packages) que comentaremos más adelante.

Para trabajar con proyectos en la mayoría de entornos, incluido BlueJ, debemos tenerlos en un soporte grabable accesible (por ejemplo en una carpeta de nuestro disco duro). No es válido por tanto un cd, dvd, unidad remota restringida o carpeta del disco duro con restricciones. El motivo es que los entornos de desarrollo trabajan grabando información en la carpeta del proyecto que se encuentre activo. Por tanto un soporte no escribible no es adecuado. Si queremos trabajar con un proyecto contenido en un cd o dvd, primero lo copiaremos a nuestro disco duro y después lo abriremos para trabajar con él.

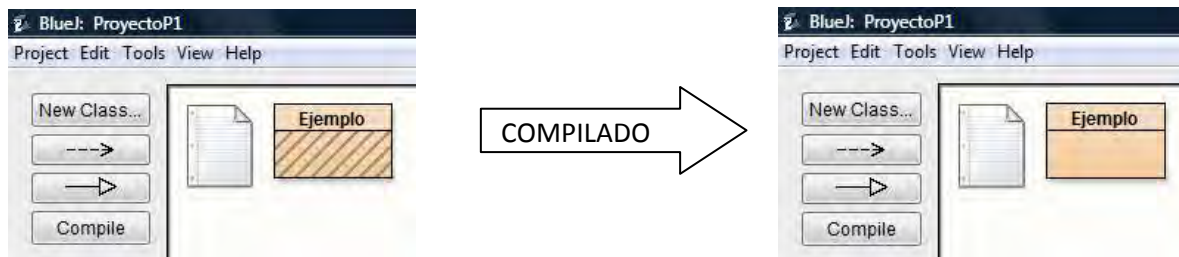
Vamos a crear nuestro primer proyecto. Para ello conviene crear primero una carpeta donde ir almacenando los proyectos que vayamos creando. Hazlo en la ruta que te parezca más adecuada. Nosotros usaremos C:/ProyectosCursoJava. Pulsamos en el menú Project → New Project y buscamos la carpeta donde vamos a guardar el proyecto.



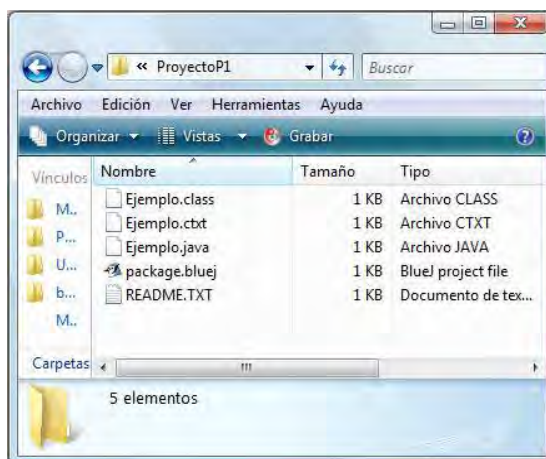
Donde pone “Nombre de archivo” escribiremos ProyectoP1 o cualquier otro nombre que nos parezca adecuado.



A continuación, en el lateral izquierdo pulsamos sobre New Class y donde nos solicita nombre para la clase (Class Name) escribimos “Ejemplo” y pulsamos aceptar. Nos aparecerá un icono con el nombre “Ejemplo” y rayas transversales. Ese icono representa una clase. Discutiremos qué es una clase en Java un poco más adelante, por ahora simplemente pensaremos que una clase es código Java. Ahora vamos a ejecutar una pequeña prueba. Pulsa sobre el botón “Compile” y el rayado que aparecía deberá haber desaparecido.



¿Qué hemos hecho en este proceso? Al crear la clase, hemos creado un archivo denominado Ejemplo.java. Al pulsar sobre “Compile”, hemos transformado el código que contenía ese archivo en bytecode y hemos creado el archivo Ejemplo.class. ¿Pero qué código hemos compilado si no hemos escrito nada? Efectivamente, no podríamos compilar algo vacío. La explicación radica en que cuando se crea una clase vacía BlueJ la crea con un código por defecto, digamos que un ejemplo muy básico de código que ya es compilable. Vamos a comprobar lo que hemos dicho sobre los archivos: para ello vamos al explorador de archivos y nos dirigimos a la ruta donde tenemos guardado el proyecto.



En esta ruta comprobamos los archivos de los que consta nuestro proyecto:

- **Ejemplo.java:** el código fuente en lenguaje Java.
- **Ejemplo.class:** el bytecode o código máquina para la máquina virtual Java.
- **Otros archivos:** archivos creados por BlueJ a los que de momento no vamos a prestar atención.

Pulsa ahora sobre el icono del archivo Ejemplo.java y con el botón derecho del ratón elige “Abrir” para verlo con un editor de texto como el bloc de notas. Al abrirse el archivo podrás ver el código fuente (e incluso podríamos cambiarlo desde aquí si quisiéramos). Cierra el editor y trata de repetir el proceso con el icono del archivo Ejemplo.class. El resultado será que Windows te indica que no sabe cómo abrir ese archivo, o, si logras abrirlo, que te aparezcan una serie de caracteres “raros”. Esto concuerda con lo que habíamos dicho: el código fuente está constituido por texto y podemos verlo y editarlo. El bytecode es un tipo de código máquina, por tanto no podemos editarlo directamente porque no es comprensible para nosotros.

Ya hemos visto que el archivo Ejemplo.java contiene un código. Esa es la razón por la que nos ha sido posible compilar anteriormente usando BlueJ. Vuelve a BlueJ y para acceder al código que se encuentra en la clase hacemos doble click en el icono, con lo cual se nos abrirá la ventana del editor de BlueJ.

**Próxima entrega:** CU00616B

**Acceso al curso completo** en [aprenderaprogramar.com](http://www.aprenderaprogramar.com) -- > Cursos, o en la dirección siguiente:

[http://www.aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=68&Itemid=188](http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=68&Itemid=188)