

Creación de sus primeros proyectos Java en NetBeans

Introducción

Acerca de los paquetes Java

- Los paquetes Java se utilizan para identificar fácilmente clases donde no haya conflictos de nombres, así como para mantener juntos los archivos .java relacionados
- La API de Java contiene muchos paquetes
- La API de Java es una biblioteca de clases ya escritas que se pueden utilizar gratuitamente y que están disponibles en Java Runtime Environment
- También podrá crear sus propias clases que se conservarán mejor en el paquete que cree
- Si no crea un paquete para estas, NetBeans, Eclipse u otros IDE utilizarán el paquete por defecto
- **Para los ejemplos de este curso, a menudo utilizará el paquete por defecto cuando desee probar los ejemplos de codificación con las instrucciones de la parte 1**
- Para la **parte 2**, creará un archivo .java dentro de un paquete denominado **mysecondproject**
- Cuando crea un archivo .java en un paquete llamado **mysecondproject**, necesita que la declaración del paquete sea su primera línea de código:
 - `package mysecondproject;`
- Si prefiere utilizar el paquete por defecto, no se utiliza una declaración de paquete

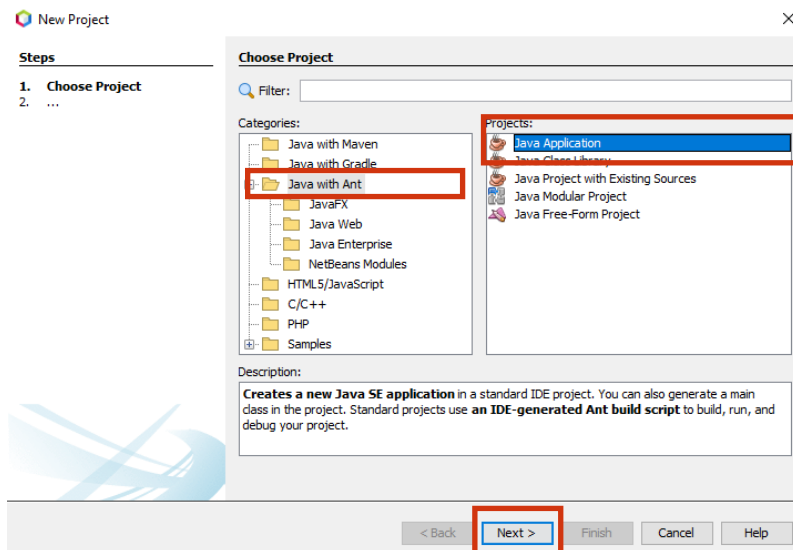
Parte 1: Crear un proyecto Java y agregar un archivo .java existente

Visión general

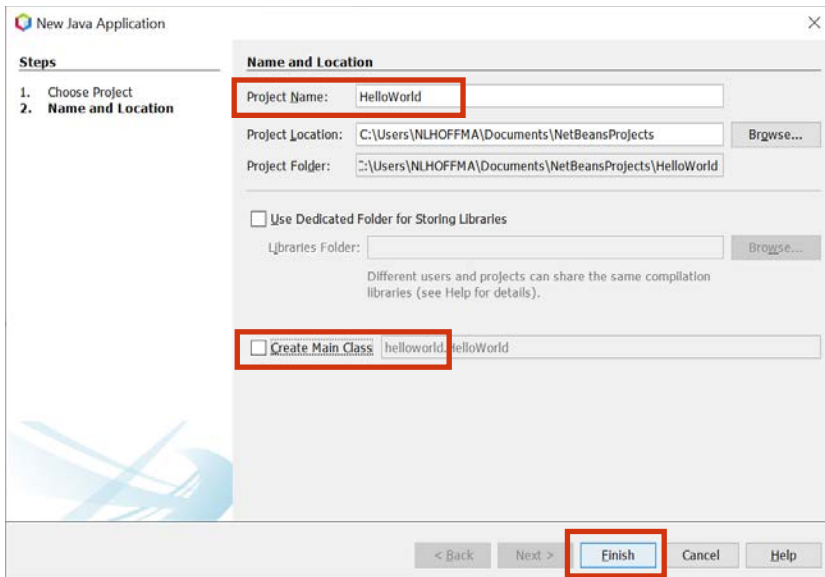
En esta sección, descargará el material de referencia HelloWorld.zip, creará un proyecto nuevo en NetBeans y agregará un archivo java existente al proyecto

Tareas

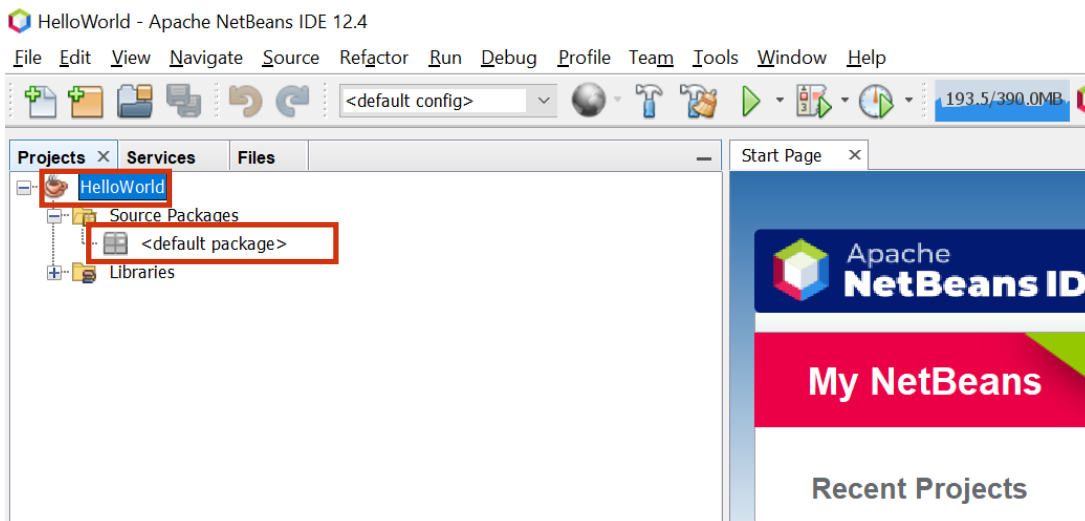
1. Descargue el archivo HelloWorld.zip de los materiales de referencia para esta práctica
2. Extraiga el archivo zip HelloWorld.zip y anote la ubicación
3. Inicie NetBeans
4. Vaya a **File > New Project** y seleccione lo siguiente:
 - a. Categories: **Java with Ant**
 - b. Projects: **Java Application**
5. Haga clic en **Next**



6. **Asigne al proyecto el nombre HelloWorld, elimine la marca** de la casilla para crear una clase main y, a continuación, haga clic en **Finish**
- a. Para este curso, se recomienda que asigne a los proyectos el mismo nombre que al archivo .java

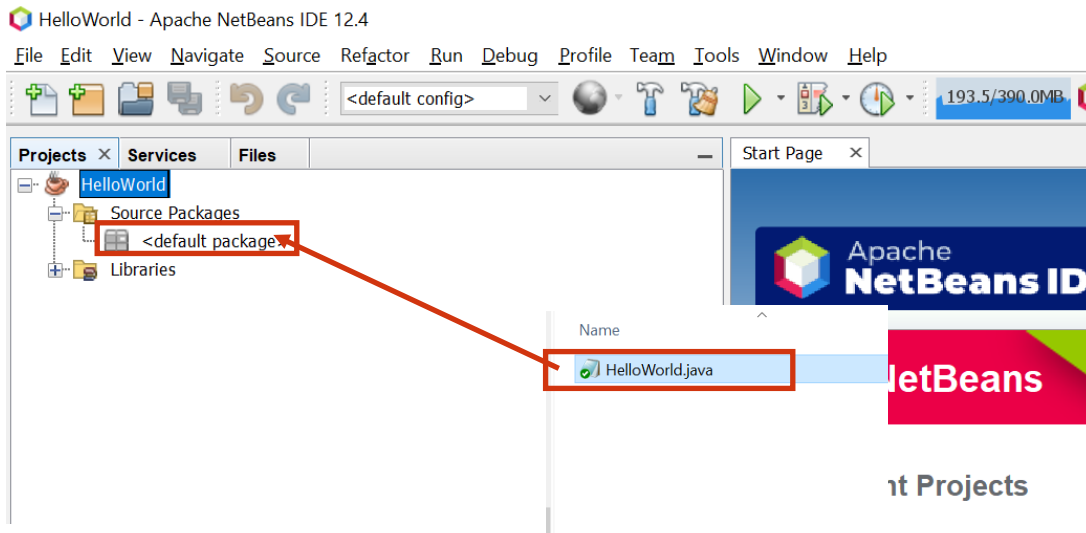


7. En la ventana Projects, **haga clic en el signo más** para ampliar el proyecto y **haga clic en el signo más** para ampliar Source Packages y que aparezca el paquete por defecto, <default package>



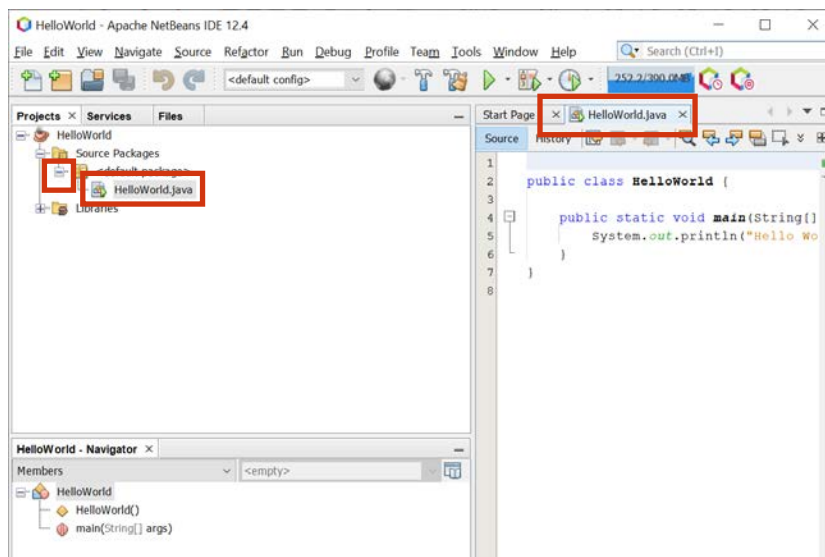
8. **Desplácese** a la carpeta en la que haya extraído el archivo HelloWorld.java en el paso 1

9. **Arrastre** el archivo HelloWorld.java desde la carpeta del dispositivo y **suéltelo** en <default package>

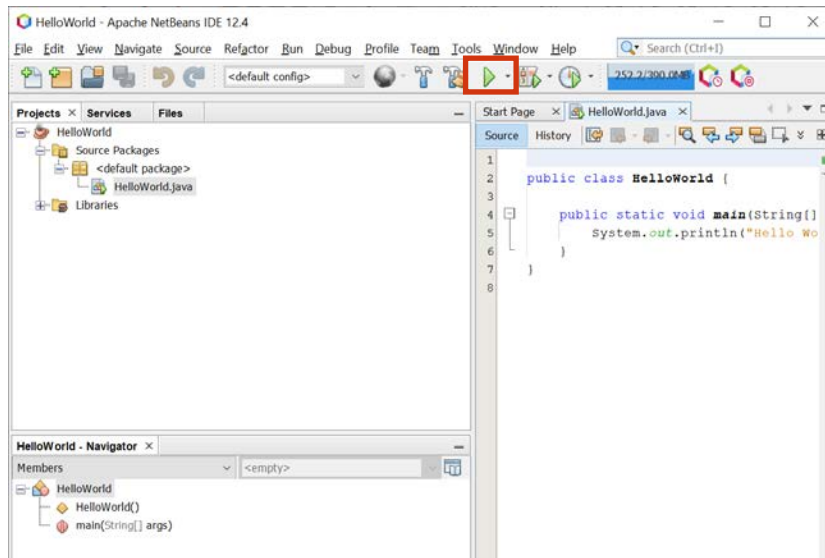


10. **Haga clic en el signo más** situado junto a <default package> y verá el archivo HelloWorld.java en el paquete

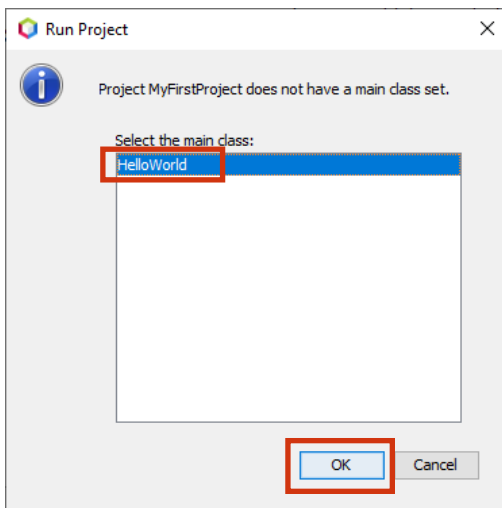
11. **Haga doble clic** en el archivo .java para que se abra en un nuevo separador en la ventana del editor de código



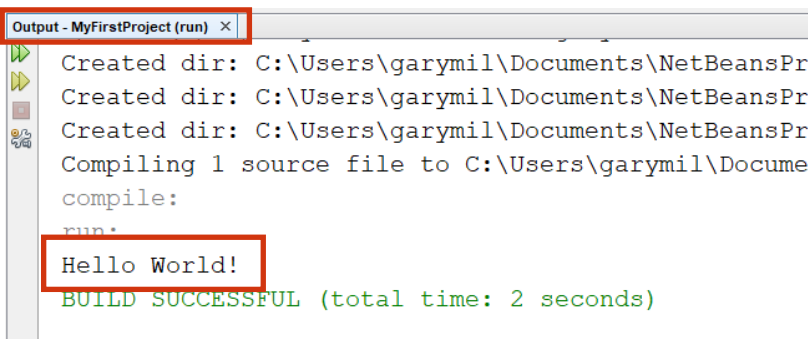
12. Haga clic en el botón Run para realizar la prueba



13. Si se le solicita que seleccione una clase main, elija la clase por defecto, HelloWorld, y haga clic en OK



14. En la ventana Output que aparece debajo del editor de código, verá el mensaje "Hello World"



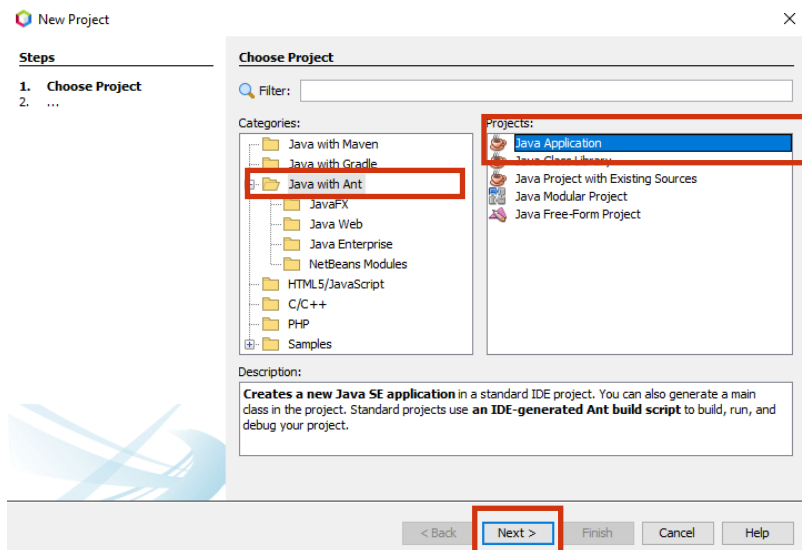
Parte 2: Crear un paquete y un proyecto en NetBeans con una clase main

Visión general

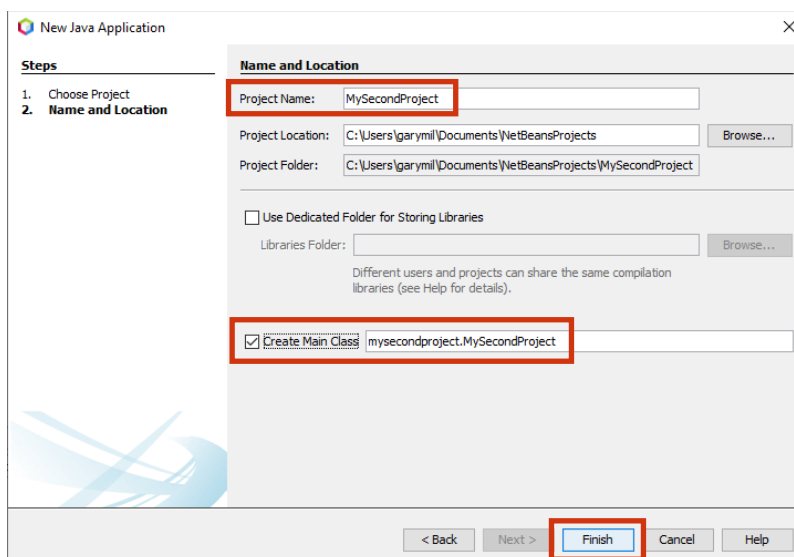
En esta sección, creará y ejecutará un proyecto Java de NetBeans con una clase Java main

Tareas

1. Inicie NetBeans si no está abierto
2. Vaya a **File > New Project** y seleccione lo siguiente:
 - a. Categories: **Java with Ant**
 - b. Projects: **Java Application**
3. Haga clic en **Next**

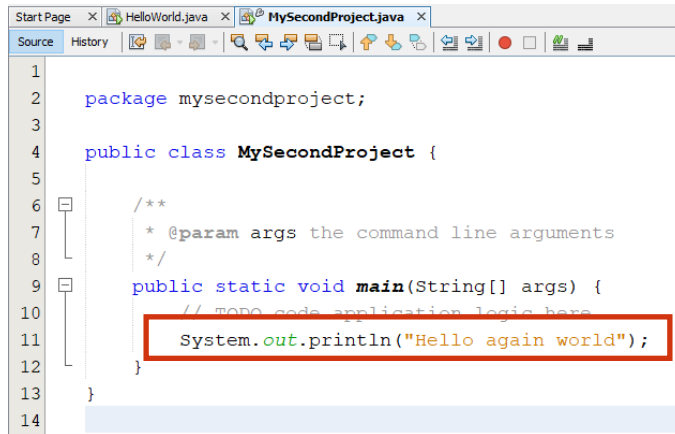


4. Asigne al proyecto el nombre **MySecondProject**, esta vez active la casilla para crear una clase main y, a continuación, haga clic en **Finish**



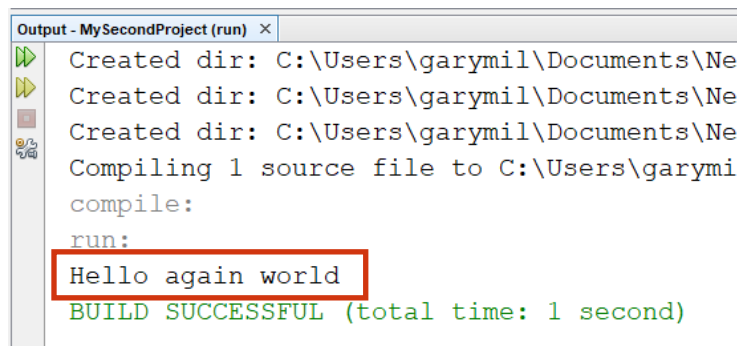
5. En el editor de código, busque el método main de la clase MySecondProject e **introduzca la línea de código** como se muestra a continuación:

System.out.println("Hello again world");



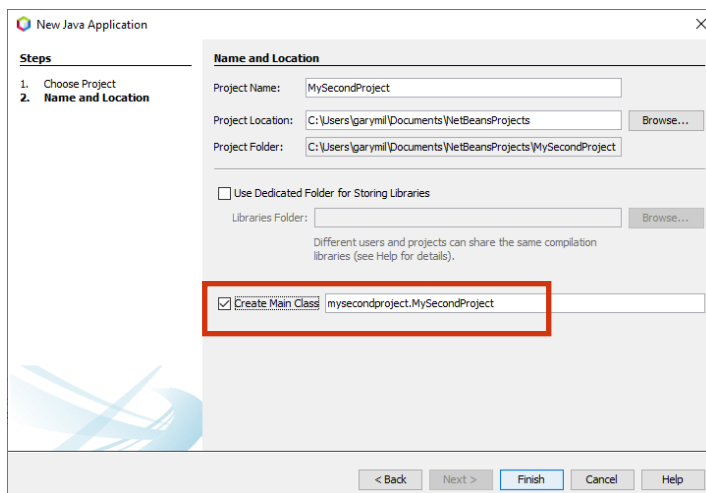
```
1
2 package mysecondproject;
3
4 public class MySecondProject {
5
6     /**
7      * @param args the command line arguments
8      */
9     public static void main(String[] args) {
10         // TODO code application logic here
11         System.out.println("Hello again world");
12     }
13 }
14
```

6. Haga clic en el botón Run para realizar la prueba
- a. Debería ver el mensaje Hello World en la consola de Output en la parte inferior del IDE

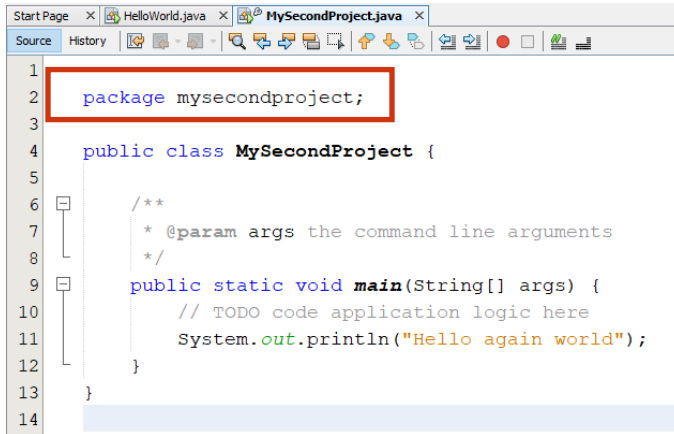


```
Output - MySecondProject (run) x
Created dir: C:\Users\garymil\Documents\Ne
Created dir: C:\Users\garymil\Documents\Ne
Created dir: C:\Users\garymil\Documents\Ne
Compiling 1 source file to C:\Users\garymi
compile:
run:
Hello again world
BUILD SUCCESSFUL (total time: 1 second)
```

- Observe una diferencia importante entre el primer proyecto y el segundo proyecto
- En MySecondProject, **cuando se activa la casilla Create Main Class en la tarea 4, también se crea un paquete con nombre**
- NetBeans agrega un paquete sugerido y un nombre de clase basados en el nombre del proyecto



- El formato del nombre sugerido es `nombrepaquete.NombreClaseMain`. Puede editar los nombres en este punto si desea otro formato distinto al formato de nomenclatura sugerido
- En la parte 1, como no ha creado una clase main, el paquete se llama `<default package>`
- Si un paquete tiene un nombre que no sea `<default package>`, es importante tener en cuenta que se debe incluir una declaración de paquete como primera línea de código en el archivo Java
- Si crea una clase en un paquete existente, NetBeans agregará automáticamente la declaración del paquete
- En el editor de código del segundo proyecto, desplácese hasta el principio del código y verá la declaración del paquete



```

1
2 package mysecondproject;
3
4 public class MySecondProject {
5
6     /**
7      * @param args the command line arguments
8      */
9     public static void main(String[] args) {
10         // TODO code application logic here
11         System.out.println("Hello again world");
12     }
13 }
14

```

- Observe que, en su primer proyecto, como el paquete era `<default package>`, si se desplaza al principio del archivo java en el editor de código, verá que no hay ninguna declaración de paquete
- Si agrega un archivo Java existente a un paquete al que se le ha asignado un nombre, deberá agregar la declaración del paquete manualmente como la primera línea de código del archivo java con el formato:
 - **package nombrepaquete;**