

ciclo: [DAM]

MÓDULO DE [PROGRAMACIÓN]

[Tarea Nº 02]

Alumno:

[Juan Carlos Filter Martín]

[15456141A]

**Contenido**

[1. Documentos que se adjuntan a este informe. 3](#__RefHeading___Toc199_400326990)

[2. RA2\_b) Se han escrito programas simples. 3](#__RefHeading___Toc201_400326990)

[3. RA2\_c) Se han instanciado objetos a partir de clases predefinidas. 4](#__RefHeading___Toc203_400326990)

[4. RA2\_e) Se han escrito llamadas a métodos estáticos. 8](#__RefHeading___Toc205_400326990)

[5. RA2\_f). Se han utilizado parámetros en la llamada a métodos. 9](#__RefHeading___Toc203_400326990_Copia_1)

[6. RA2\_g) Se han incorporado y utilizado librerías de objetos. 10](#__RefHeading___Toc348_2791400153)

[7. RA2\_h) Se han utilizado constructores. 11](#__RefHeading___Toc350_2791400153)

[8. RA2\_i) Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples. 12](#__RefHeading___Toc352_2791400153)

# Documentos que se adjuntan a este informe.

A continuación se detallan los documentos que componen la presente entrega de la tarea:

1. Informe de elaboración de la tarea.
2. Proyecto Java completo.

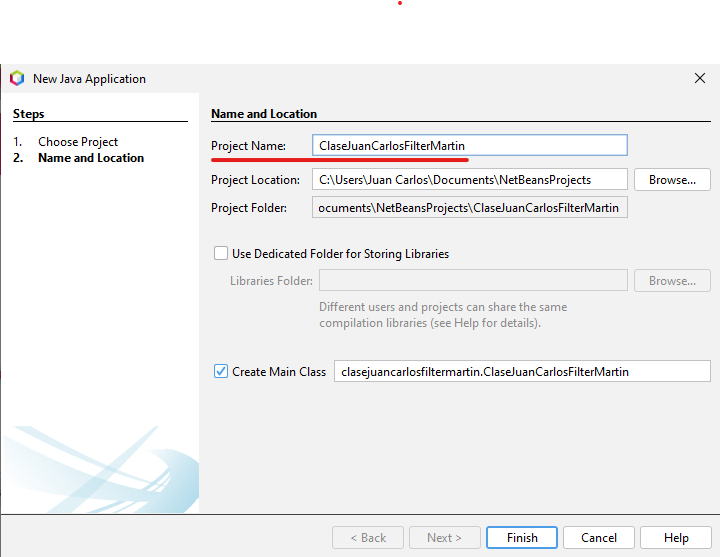
# RA2\_b) Se han escrito programas simples.

Para crear un Proyecto de Java en NetBeans nos dirigimos a **File > New Project...**

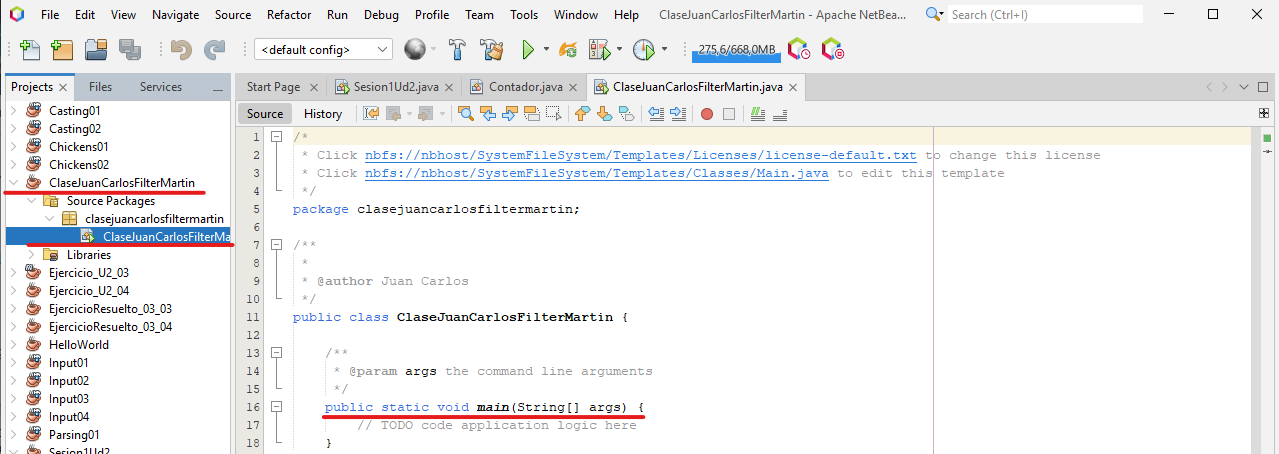
y en la siguiente ventana elegimos nuestro proyecto en nuestro caso como es Java vamos a **Java with Ant > Java Class Library**

## 

Ahora tendremos que asignarle un nombre al proyecto que en mi caso sería **ClaseJuanCarlosFilterMartin** en el apartado **Project Name.**

También podemos cambiar la localización del proyecto, crear automaticamente la clase main, etc.

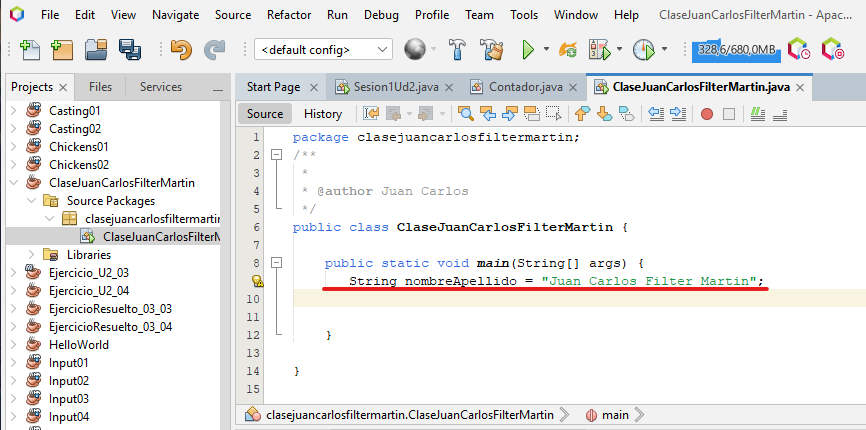
Como podemos ver con esto ya tendríamos nuestro proyecto con la clase main creada.



# RA2\_c) Se han instanciado objetos a partir de clases predefinidas.

**Crear variable con nombre y apellidos.**

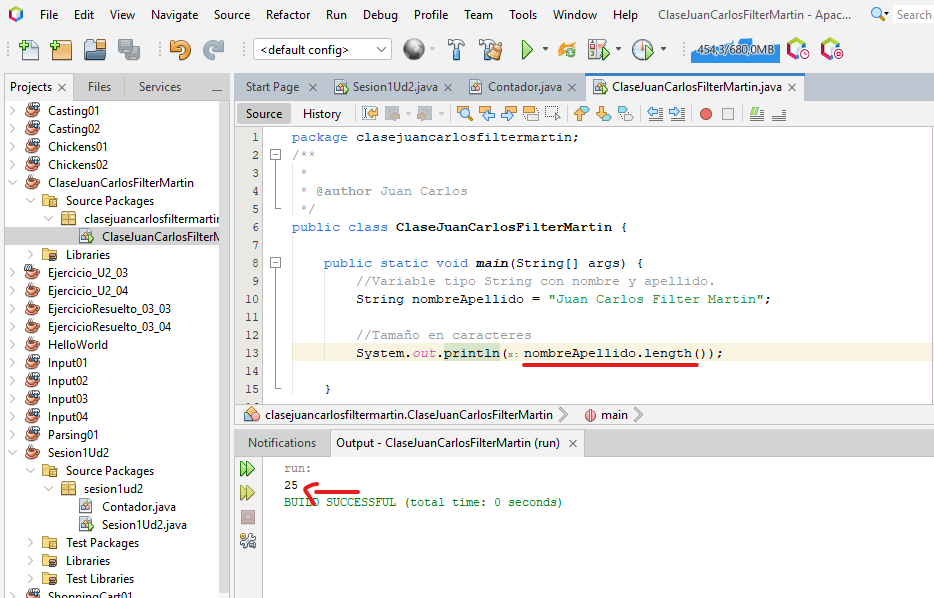
Tiene que ser una variable de tipo String ya que es una cadena de caracteres.



**Extraer y emprimir la siguiente información:**

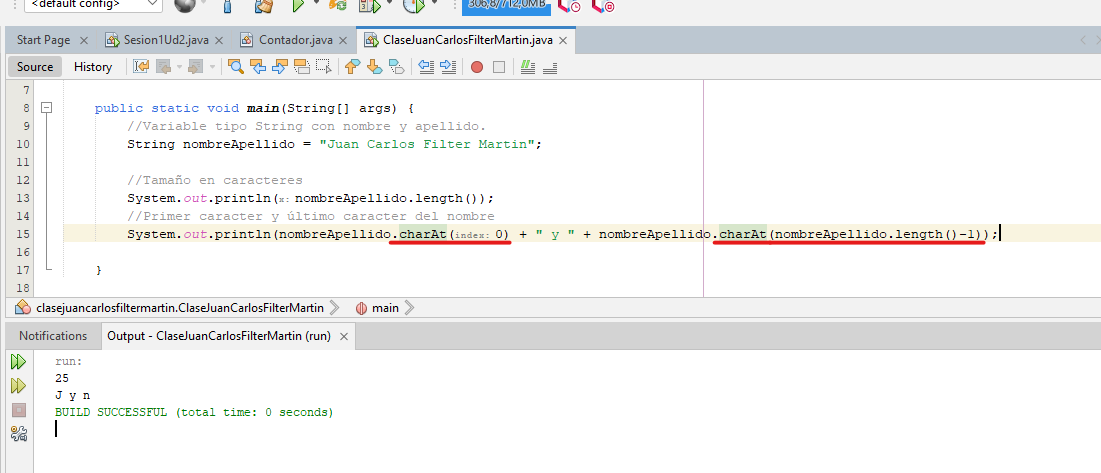
Tamaño en caracteres.

Para ello usamos el método **length** en nuestra variable que es de tipo String.



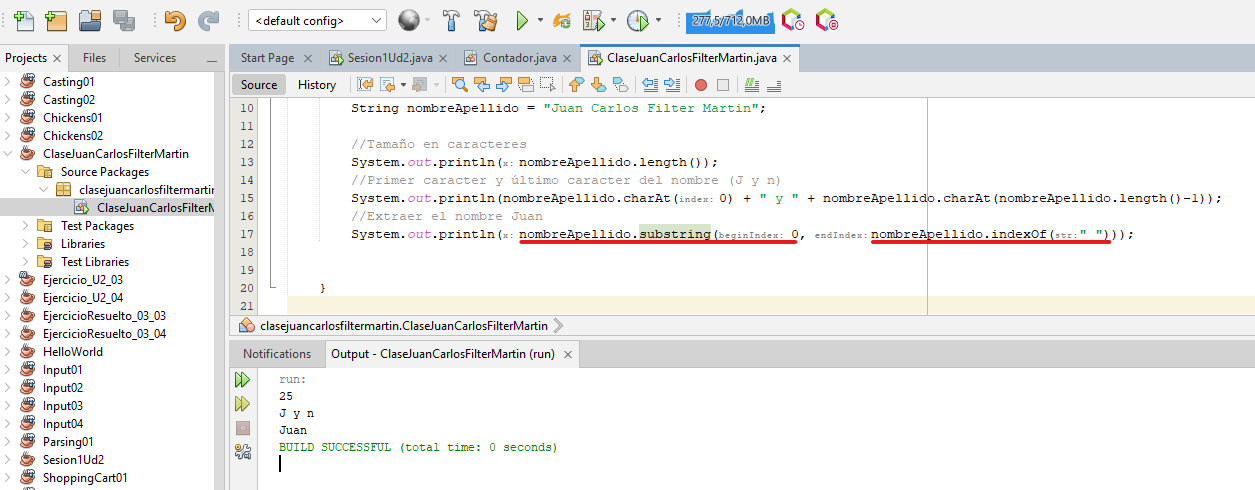
Primer caracter y último caracter del nombre.

Usaremos el método: **charAt (0)** indicándole que empiece en el primer carácter y concatenándolo con **charAt(nombreApellido.length()-1)** para indicarle que se posicione en el último carácter restándole -1 ya que el método charAt comienza en 0 y todo esto **pasárselo al método charAt** para imprimir el ultimo carácter de la variable.



El nombre. Si tiene varias partes solo se devuelve la primera.

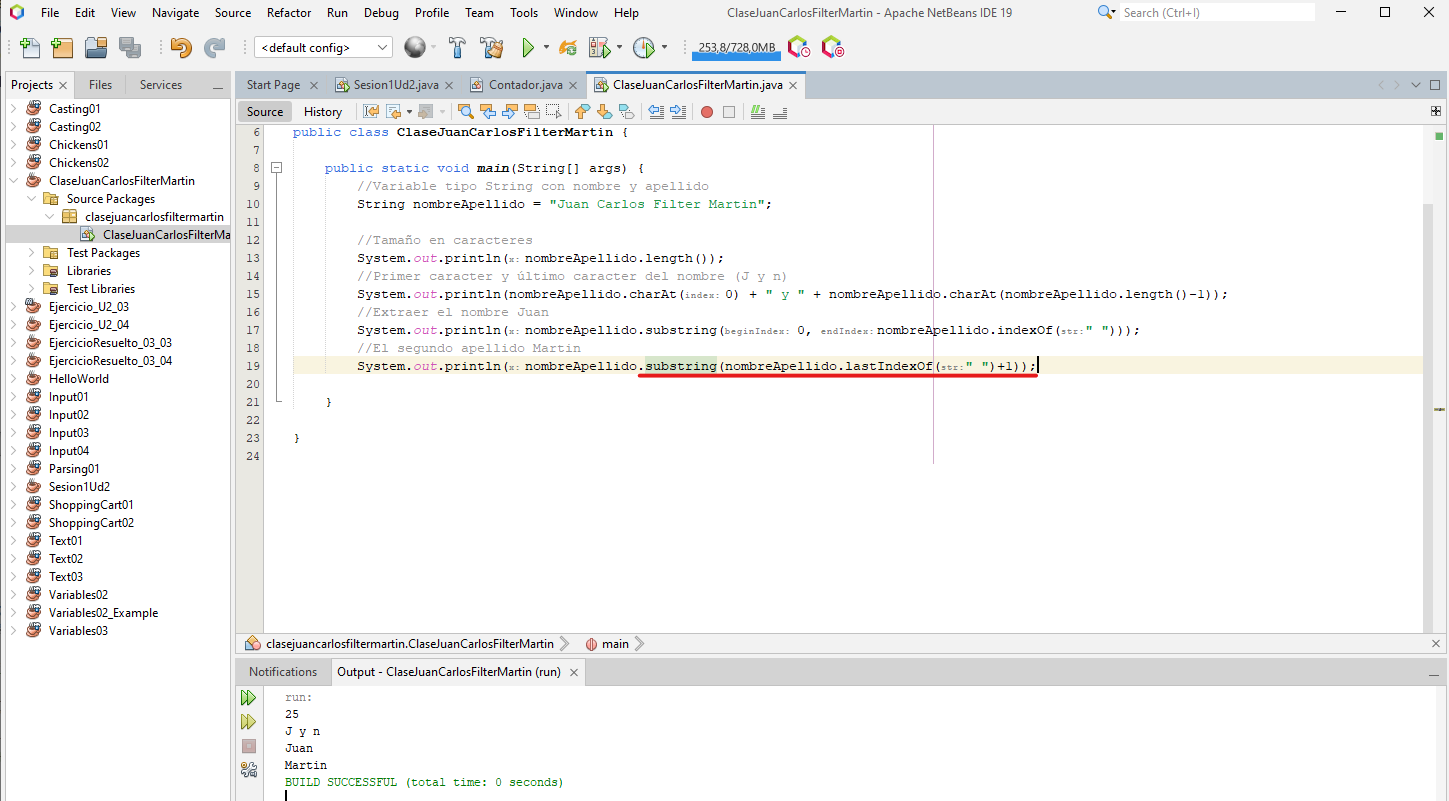
Con el método **substring(0)** imprimiremos la cadena de caracteres desde la posición 0 hasta el método **indexOf(" ")** Indicándole que se detenga antes del primer espacio.



El segundo apellido. Si tiene varias partes solo mostrar la última.

En este caso **utilizamos el método substring** y le indicamos que **comience a partir del ultimo espacio** para ello voy a **utilizar lastIndexOf**. *Este método comienza desde la ultima posición hasta el carácter indicado (“ “).*

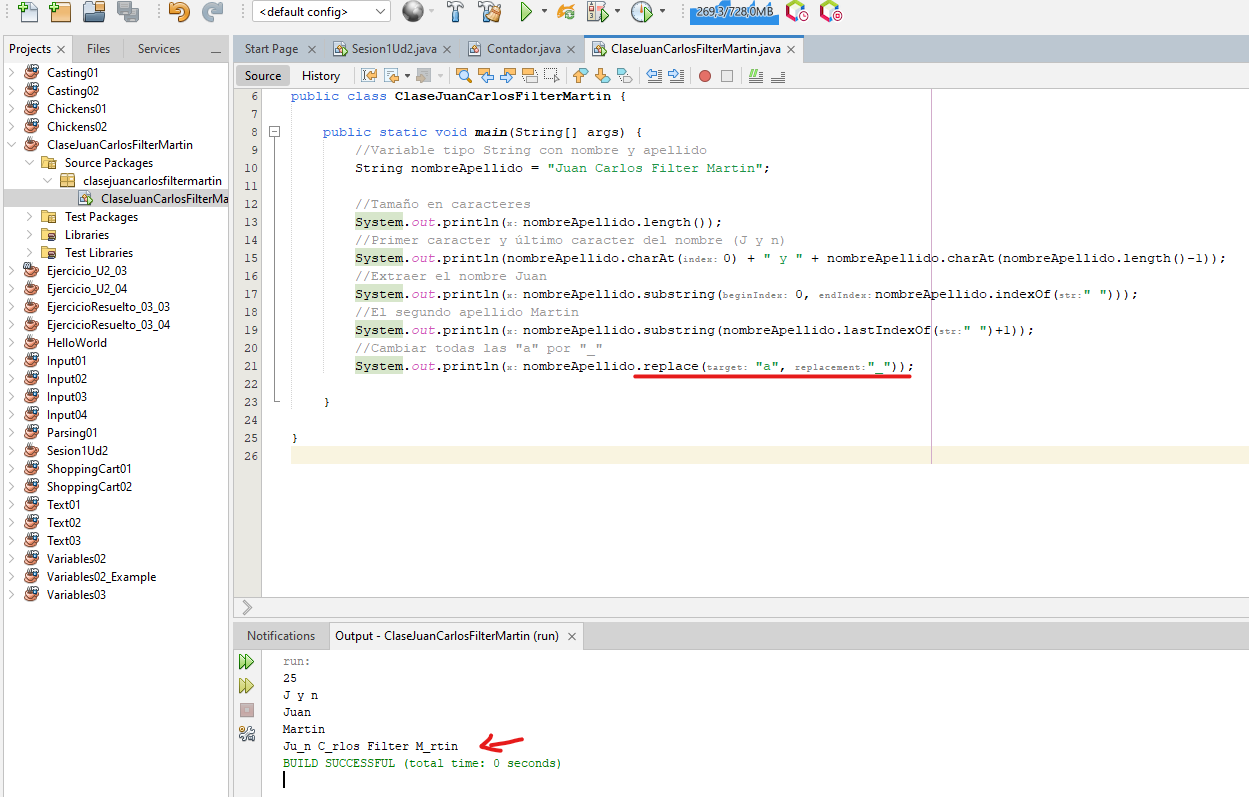
Entonces **ponemos el lastIndexOf dentro del substring enviándole la posición del ultimo espacio sumándole +1** para que nos imprima a partir de la primera letra del ultimo apellido



Cambiar todas las "a" por "\_" .

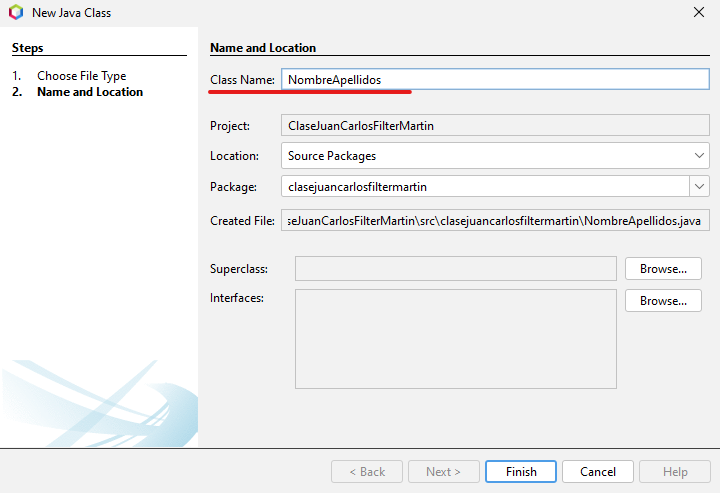
Para esto utilizamos .replace y le indicamos que “a” se remplace por “\_”

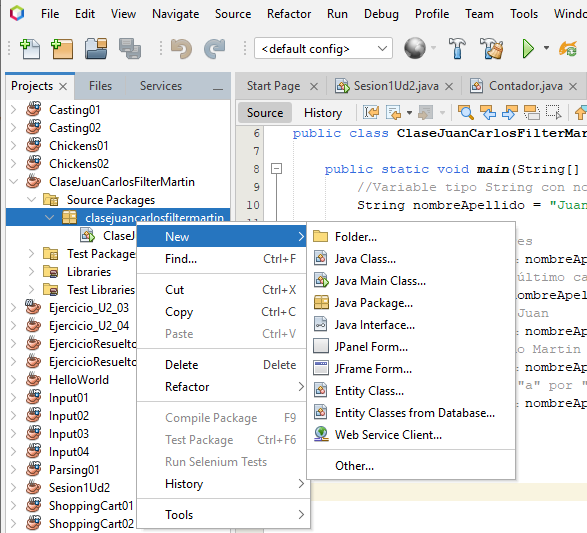
**nombreApellido.replace("a", "\_")**



**Dentro del proyecto crear una clase que se denomine, "NombreApellidos.java".**

Hay varias formas de crear una clase pero por ejemplo podemos pulsar boton derecho sobre el paquete del proyecto> new > Java Class… e introducimos el nombre de la clase en Class Name.

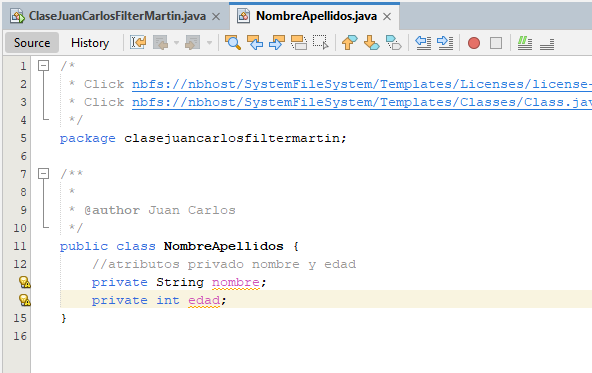




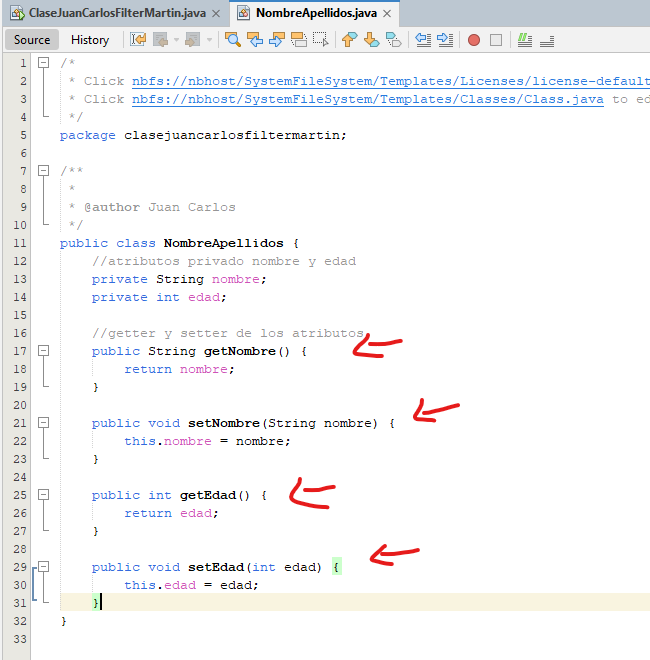
Con esto ya tendríamos nuestro proyecto con la:

clase main **ClaseJuanCarlosFilterMartin** y la clase **NombreApellidos**

Clase con 2 atributos privados (nombre y edad)



Construir getter y setter para los atributos de la clase

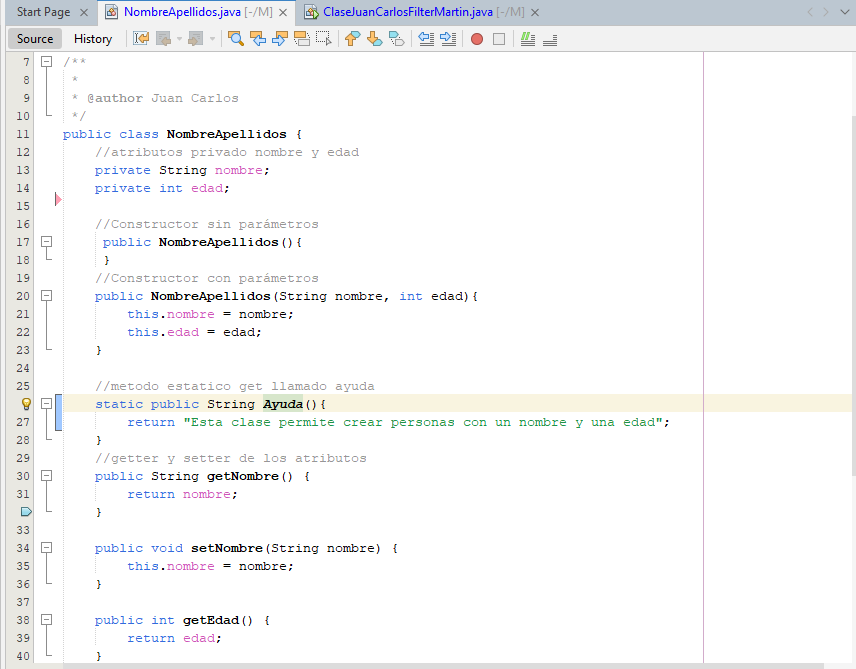


# RA2\_e) Se han escrito llamadas a métodos estáticos.

Crear un método estático llamado ayuda que único que hará será mostrar un mensaje diciendo: "Esta clase permite crear personas con un nombre y una edad"

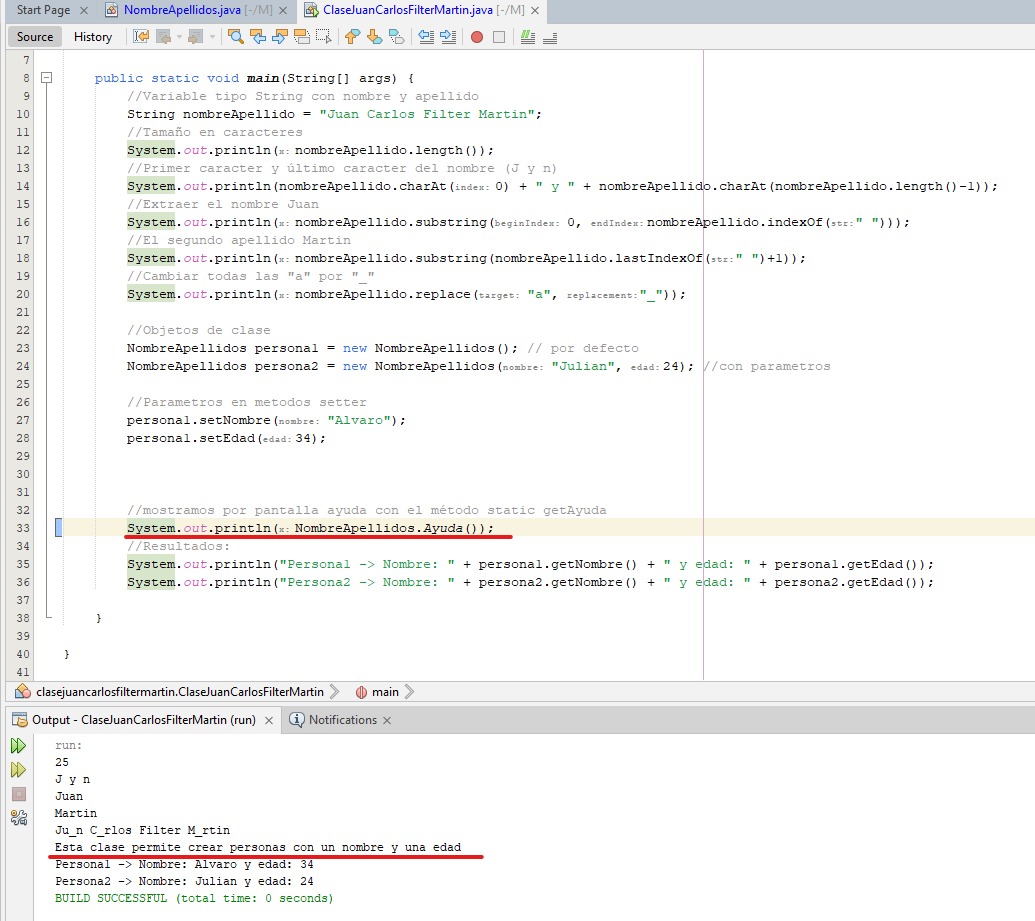
Creamos un método **static public String Ayuda()** diciéndole que devuelva:

*"Esta clase permite crear personas con un nombre y una edad”*



Nos vamos a la **clase main** y con **un System.out.println** llamando a nuestra clase **NombreApellidos** y el método donde está la variable ayuda podemos ver el resultado por pantalla:

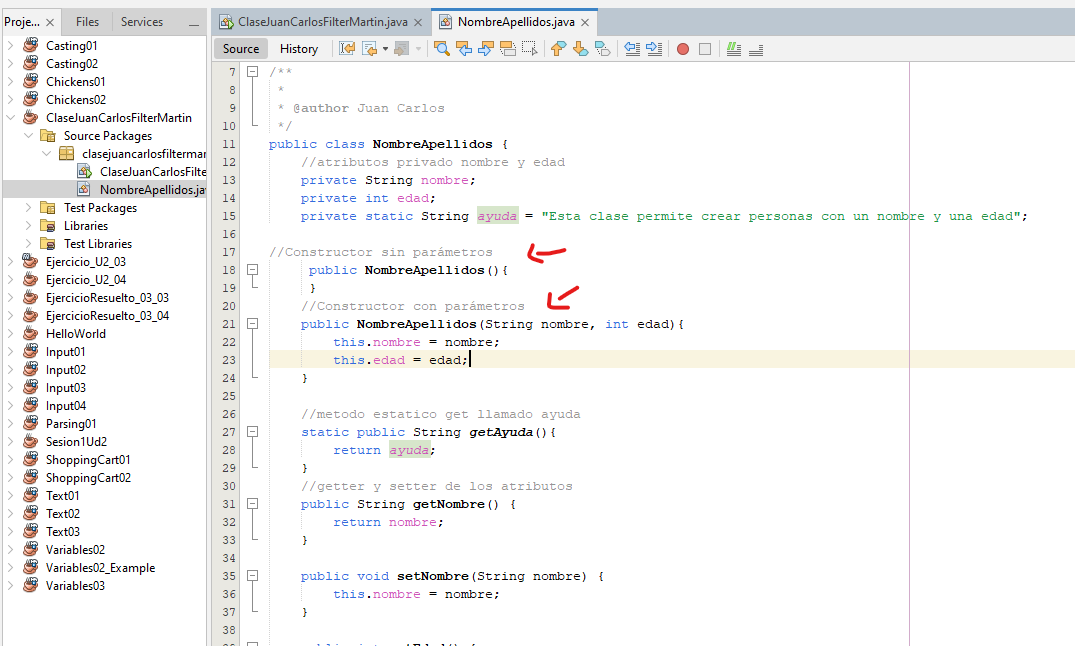
**System.out.println(NombreApellido.getAyuda());**



# RA2\_f). Se han utilizado parámetros en la llamada a métodos.

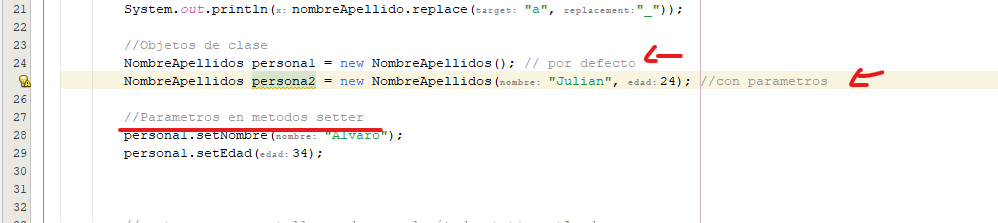
Tanto los métodos setter como uno de los constructores, deberán llevar parámetros donde recibirán los datos de la persona que vamos a crear (nombre y edad).

**Constructores por defecto y con parámetros** en la clase Nombreapellidos

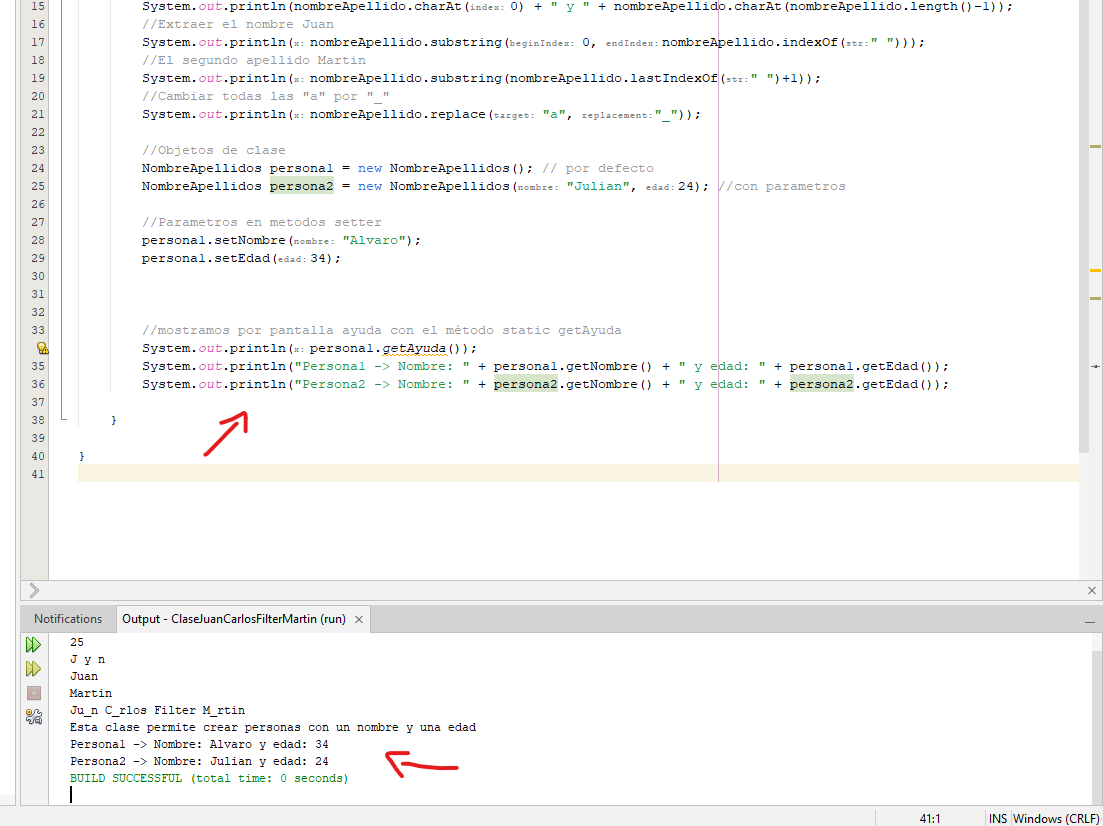


Ahora en la clase main creamos un objeto **persona1** con el constructor por defecto y otro objeto **persona2** con el constructor indicándoles parámetros.

Seguido de los **parámetros** asignados a los **metodos setter** al objeto **persona1**

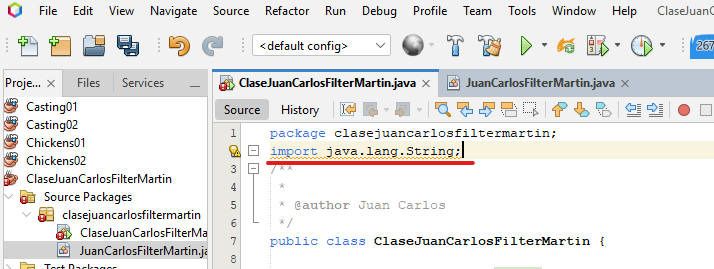


Resultado al imprimir por pantalla:



# RA2\_g) Se han incorporado y utilizado librerías de objetos.

Al estar usando la clase String no hace falta importarlas ya que se importa de forma predeterminada en todos los programas java. De todas forma se importaría de esta forma:

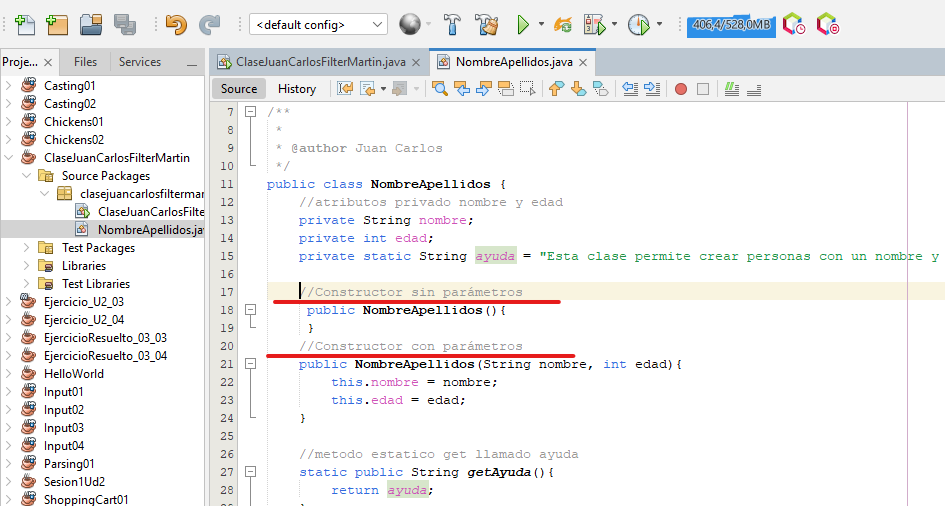


# RA2\_h) Se han utilizado constructores.

**Dos constructores:**

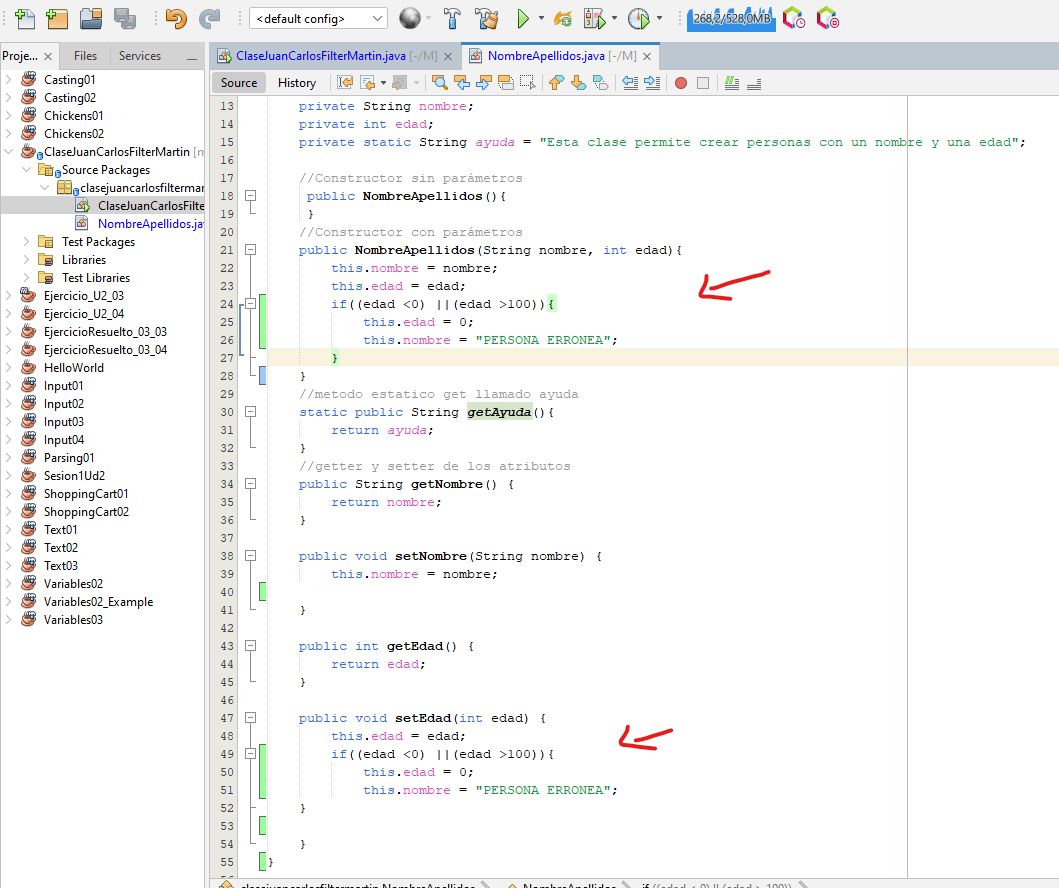
**1º sin parámetros.**

**2º constructor recibirá como parámetros nombre y edad que usará para inicializar los atributos.**

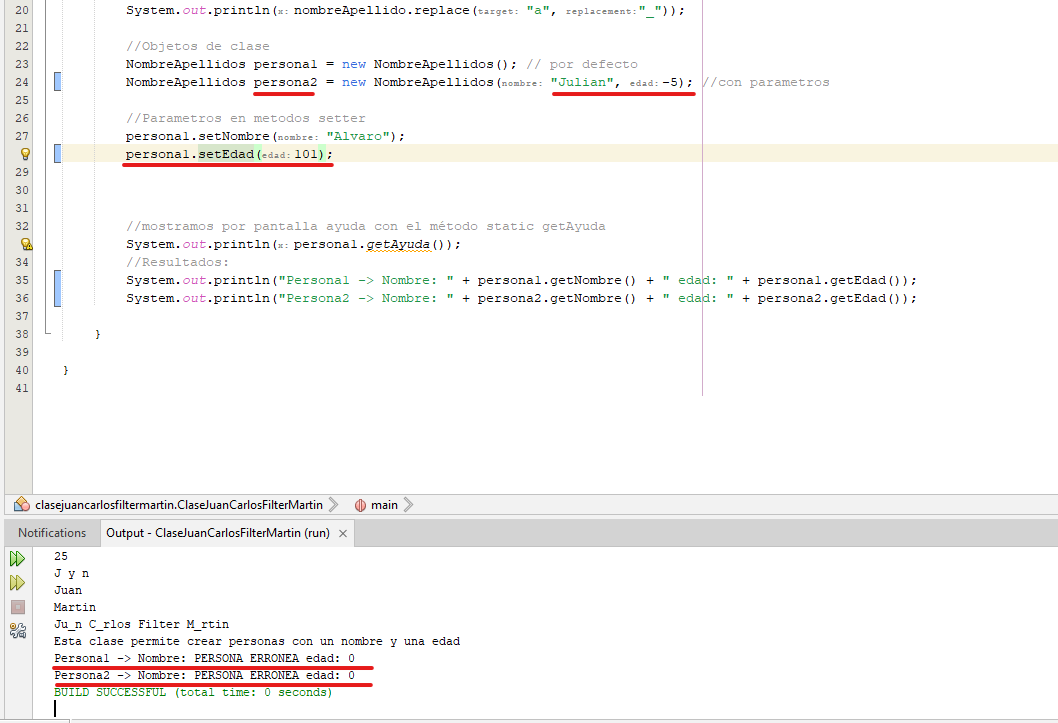
****

**Tanto en las funciones set como en los constructores se comprobará que la edad sea correcta (0-100 años). *Si intentamos introducir una edad incorrecta, se pondrá a 0 y el nombre se pondrá a "PERSONA ERRONEA".***

Para ello vamos a indicar tanto en el constructor como en el set edad con un if que si edad es menor a 0 o mayor a 100 cambie la variable edad = 0 y la variable nombre = “PERSONA ERRONEA”

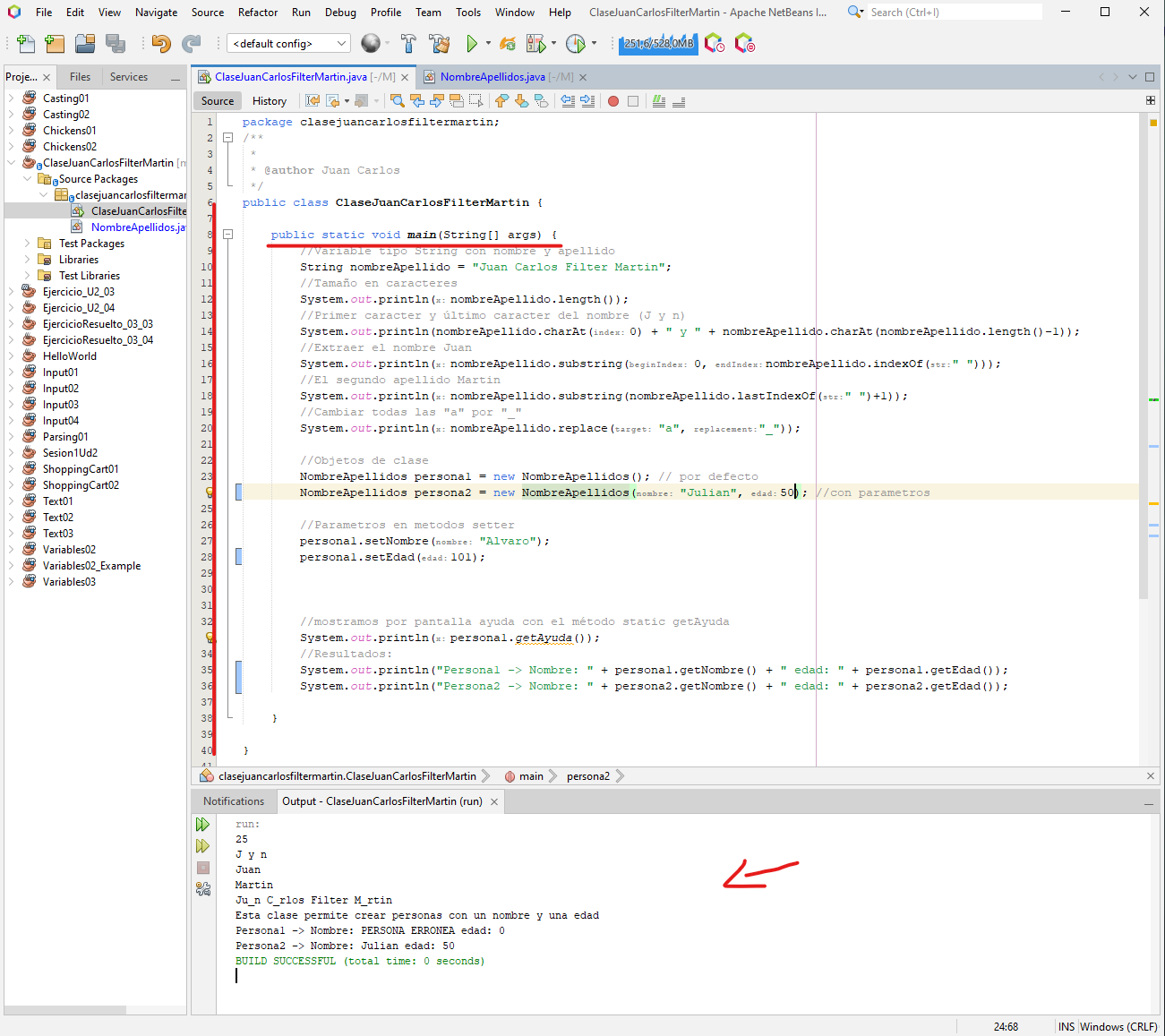


Podemos ver los resultados si vamos a nuestra clase main e introducimos un campo que salga del rango de edad entre 0 y 100:

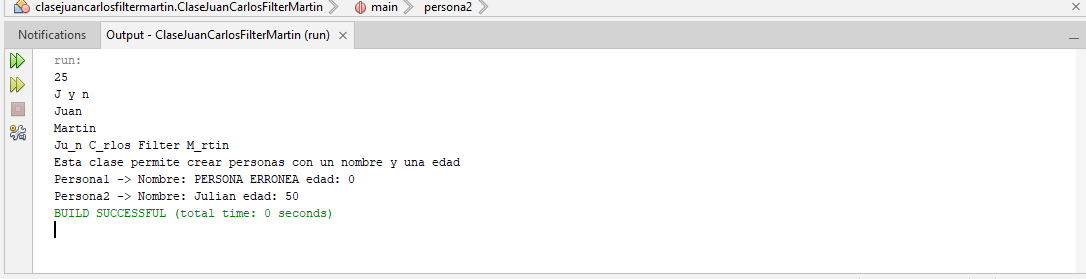


# RA2\_i) Se ha utilizado el entorno integrado de desarrollo en la creación y compilación de programas simples.

**Todas las pruebas, tanto las referentes a la clase String como a la clase que vamos a crear nosotros se harán desde la función main.**

****

**Al ejecutar deberemos de obtener unos mensajes que debemos de capturar y mostrar en el informe como prueba de que la aplicación funciona.**

****

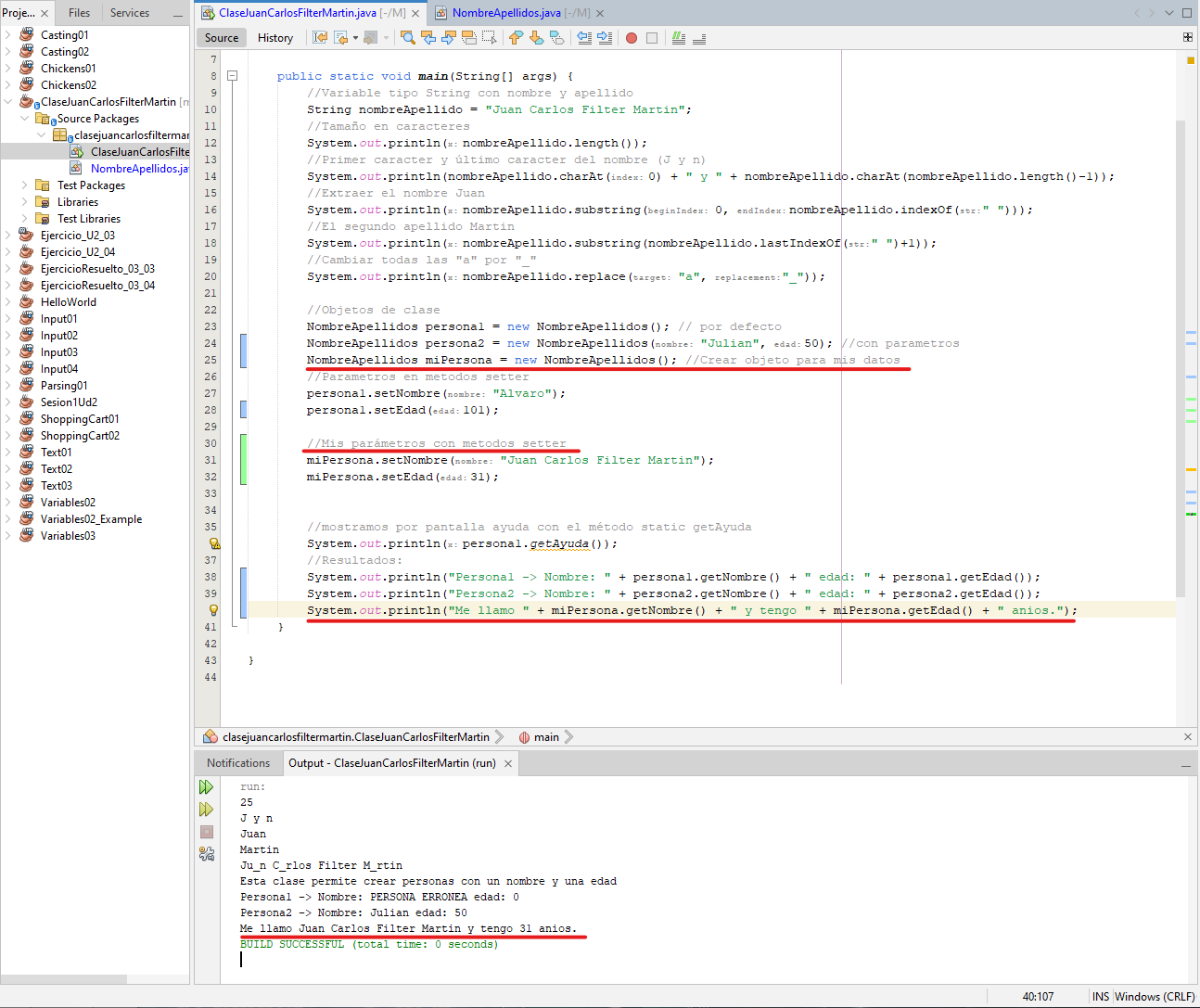
**Para probar nuestra clase, vamos a crear un objeto con los valores: vuestro nombre y vuestra edad**

Para ello:

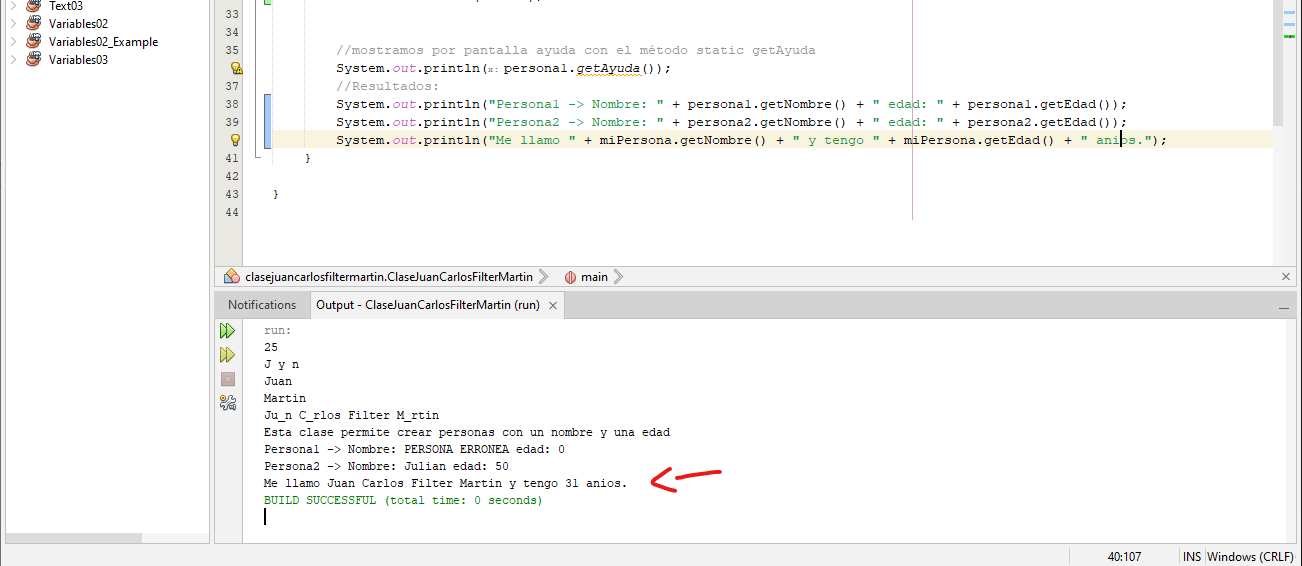
1º **Instanciamos un objeto** con nombre miPersona de la clase NombreApellidos

2º **Le asignamos** los parámetros del **nombre** y la **edad mediante** los **metodos set** “Juan Carlos Filter Martin” y edad 31.

3º **Escribimos un System.out.print** con los el objeto **miPersona.getNombre** y **getEdad**

****

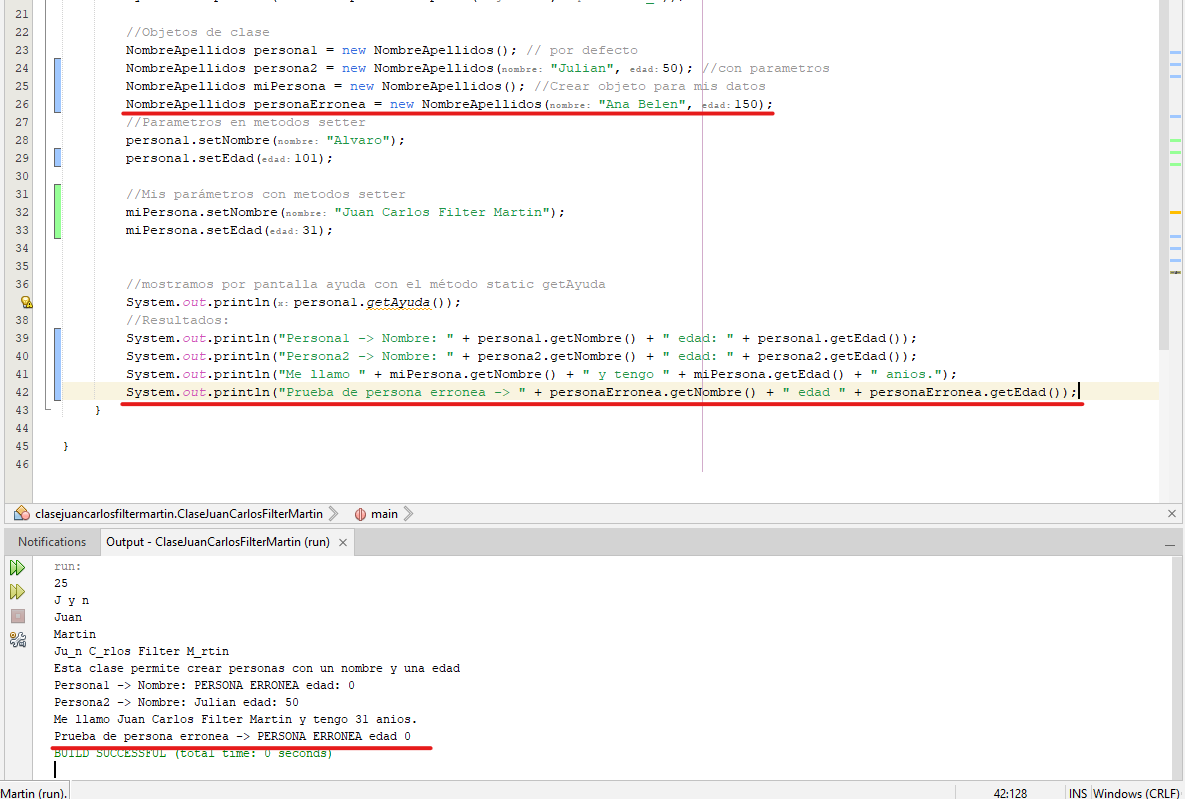
**Mostrar por pantalla el nombre y edad del objeto creado.**

****

**Crear un objeto con datos erróneos (edad fuera del rango permitido), para ver qué mensaje aparece.**

Creamos un objeto en el que vamos a indicarle una edad mayor a 100 en el propio constructor

Usamos el System.out.println y podemos en el resultado que nos ha cambiado el nombre a PERSONA ERRONEA y con edad de 0.

****