**Caso 1:**

El sistema pedirá al usuario un número entre 100 y 1000.

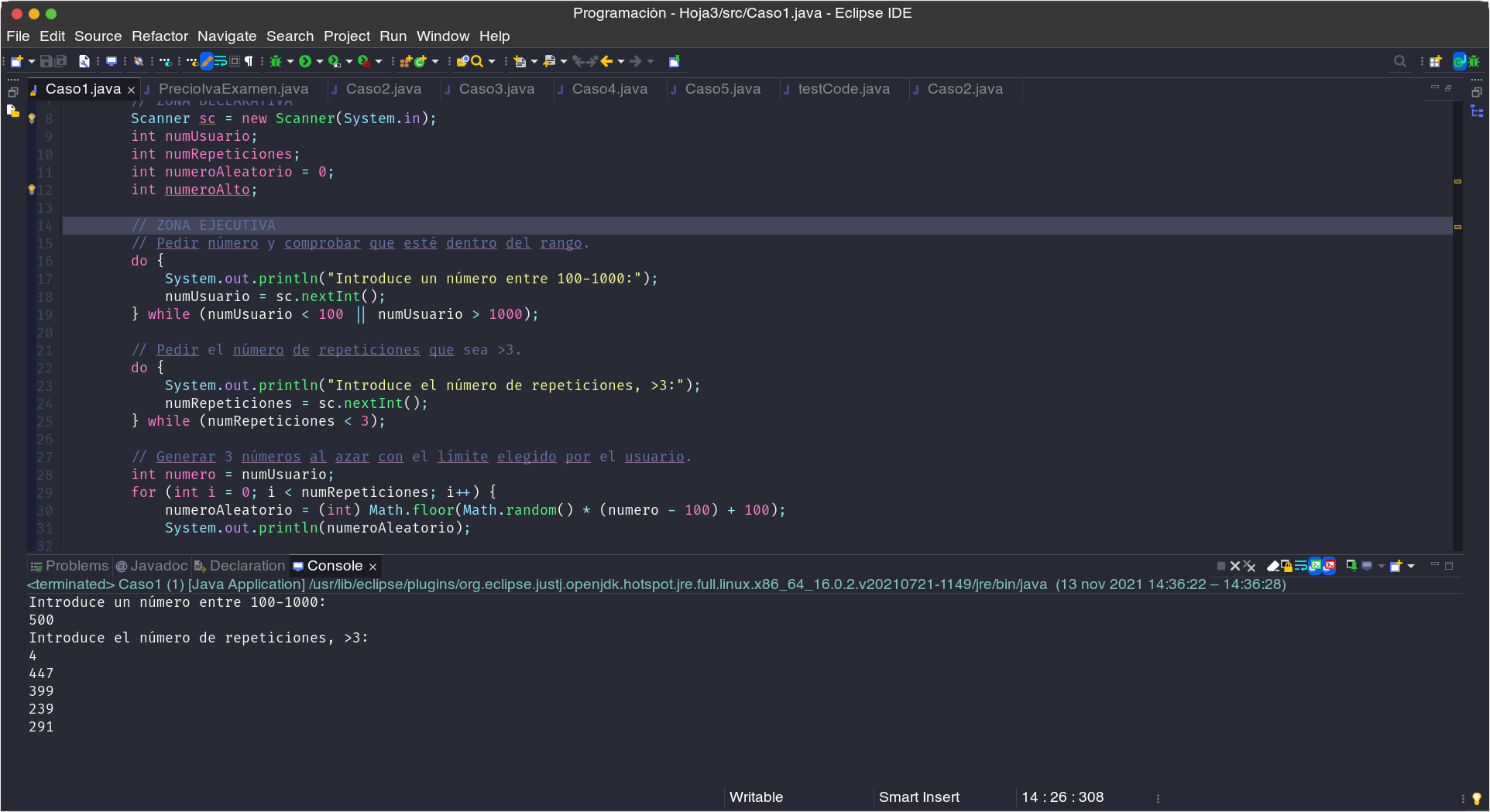
e El sistema pedirá un número de repeticiones superior a 3.

No se admitirán números fuera de esos rangos.

El sistema generará números al azar entre 100 y el número seleccionado por el usuario. Generará tantos números como repeticiones se pidan.

Mostrará los tres números más altos que aparezcan al azar tras las repeticiones.

PRUEBAS:

No consigo resolver el tercer problema del ejercicio. Al tener numeroAleatorio, por más que genere más de 3 números aleatorios, al final numeroAleatorio siempre tendrá un solo valor¿? Tendría que crear más variables para luego poder compararlas con ifs¿? Pero generar el número aleatorio no sé con qué lo haría, con un for¿? y cada número aleatorio almacenarlo en variables¿? Pero, cuántas variables declaro si no sé cuántos números puede introducir el usuario¿?

**Caso 2:**

Determina el precio de la compra según el producto y el peso.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **250g** | **500g** | **1000g** |
| **Manzanas** | 1,5 | 3 | 5,8 |
| **Mandarinas** | 2 | 4 | 7,2 |
| **Naranjas** | 2,7 | 5 | 9,1 |

El sistema permitirá comprar varios productos hasta que el usuario decida parar.

Finalmente se mostrará el precio total de la compra.

PRUEBAS:

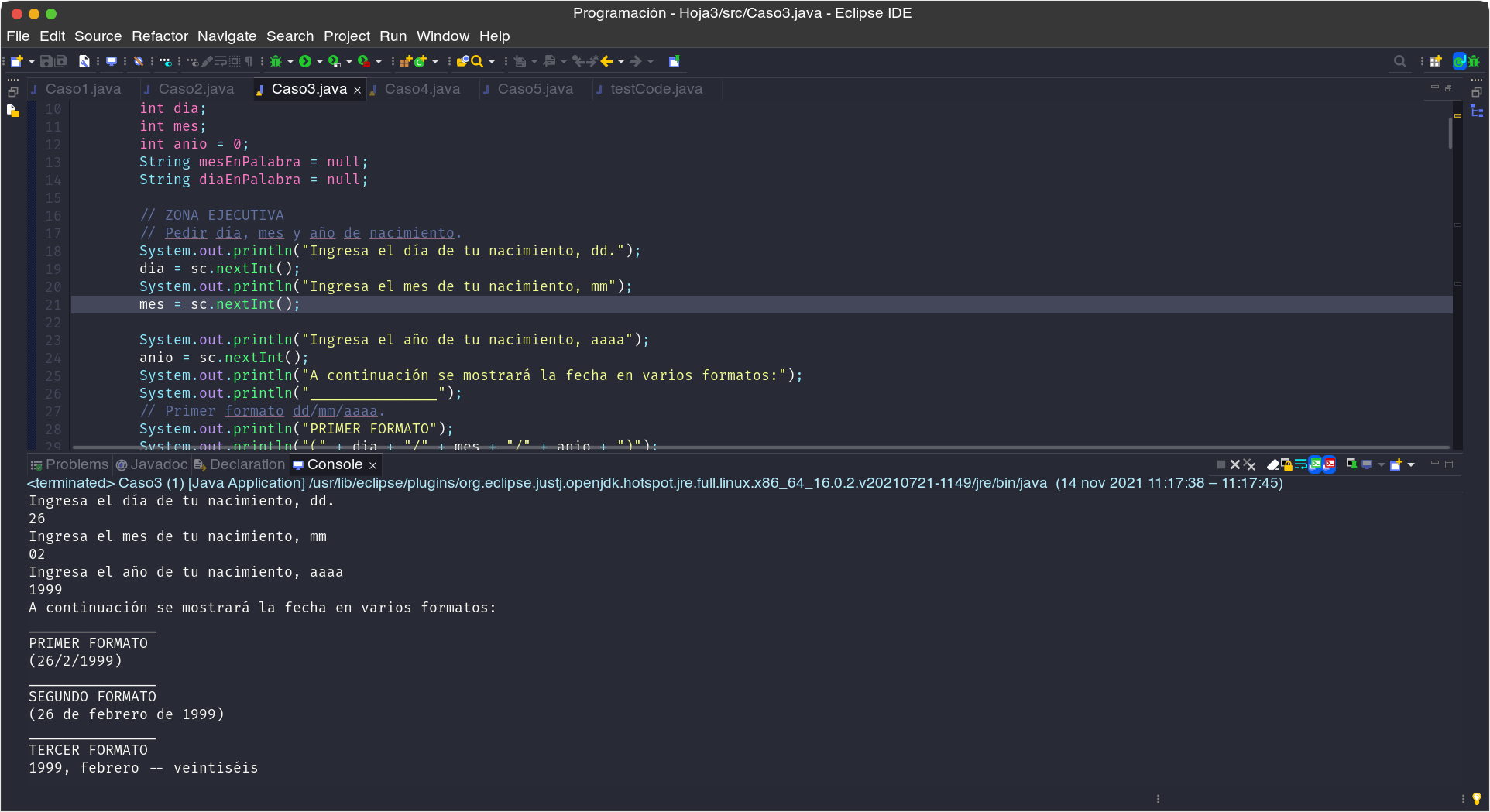
**Caso 3:**

Preguntamos el día, mes y año de nacimiento del usuario.

Mostramos varios formatos:

* ##/##/#### (12/03/2021)
* Día de mes de año (12 de marzo de 2021)
* Año, mes – dia (2021, marzo – doce)

PRUEBAS:



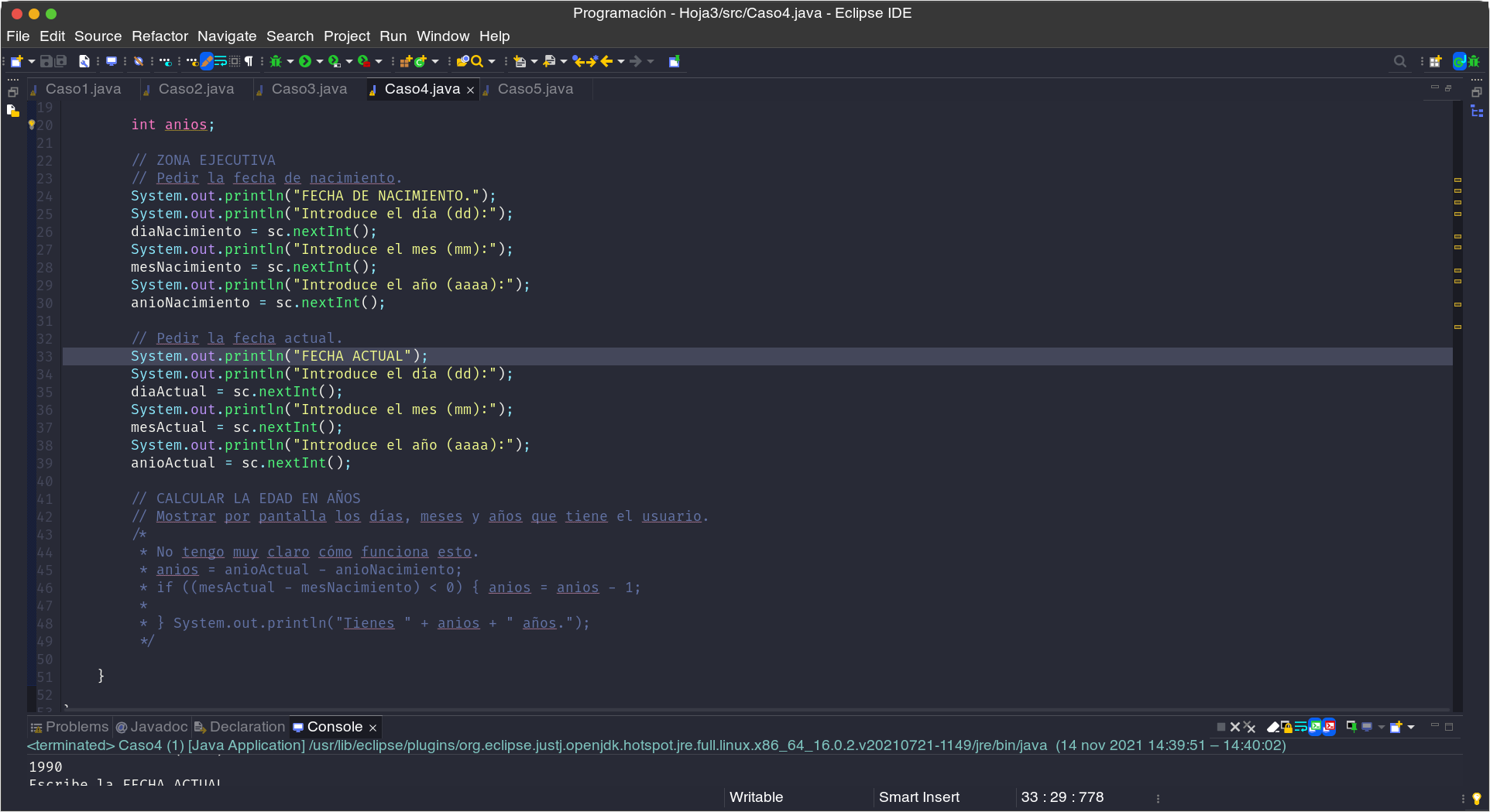
**Caso 4:**

Sobre el caso anterior.

Pedimos la fecha actual con día, mes y año.

Mostramos la edad del usuario en años. Cuidado que no se trata simplemente de restar los años.

PRUEBAS:



**Caso 5:**

Resuelve las reglas de tres pidiendo al usuario A y B:

A – 100%

B - ?

--------------------------

A – 100%

? - B

PRUEBAS:



**Se valora/penaliza:**

* Cada caso vale 2 puntos.
* Uso de las variables adecuadas. Justifícalas.
* Uso de las estructuras adecuadas.
* Uso de comentarios útiles.
* Solidez y optimización de la solución.
* Pruebas realizadas. Se deben adjuntar pantallazos de las pruebas de cada caso en este mismo documento, después de cada caso. Si el programa no funciona se debe indicar dónde y la razón.
* El código mal tabulado, el uso de sentencias de interrupción o el uso de materia no vista en clase aún **suponen una nota de cero en el caso práctico**.
* Se entregan los .java y este documento en pdf comprimidos en un archivo zip con “tu nombre – Hoja 3.zip”. El zip debe contener Hoja 3 2021.pdf, caso1.java, caso2. Java, caso3.java, caso4.java y caso5.java. **No se corregirán entregas en otro formato.**