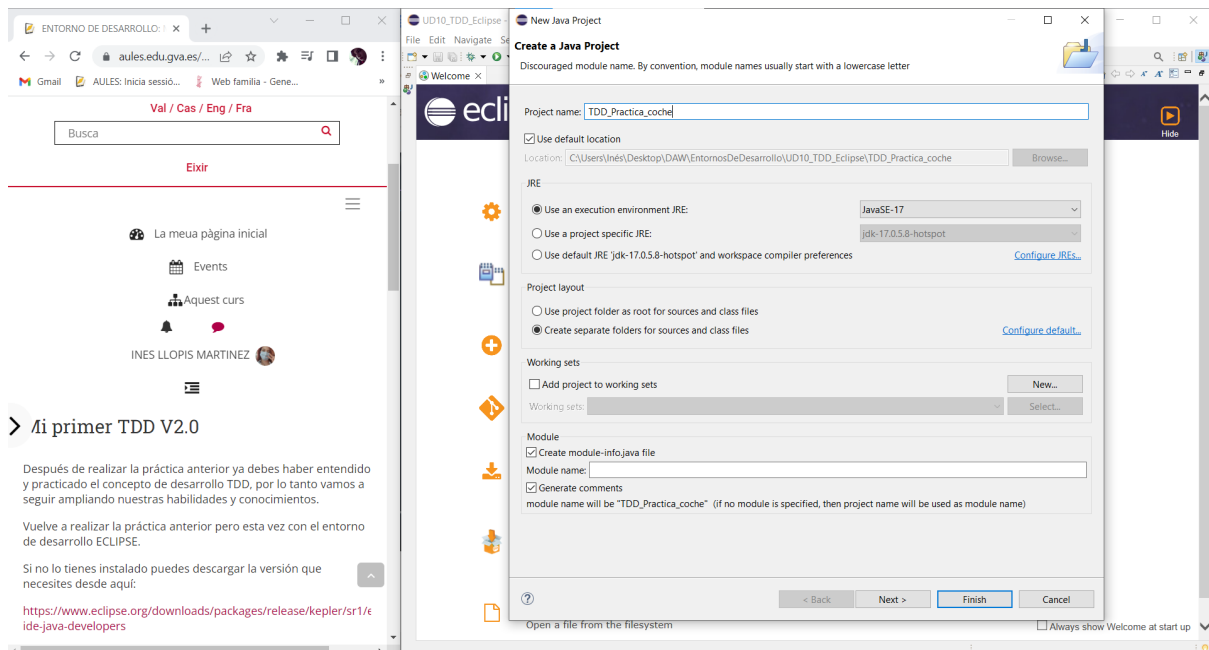


# Entornos de Desarrollo

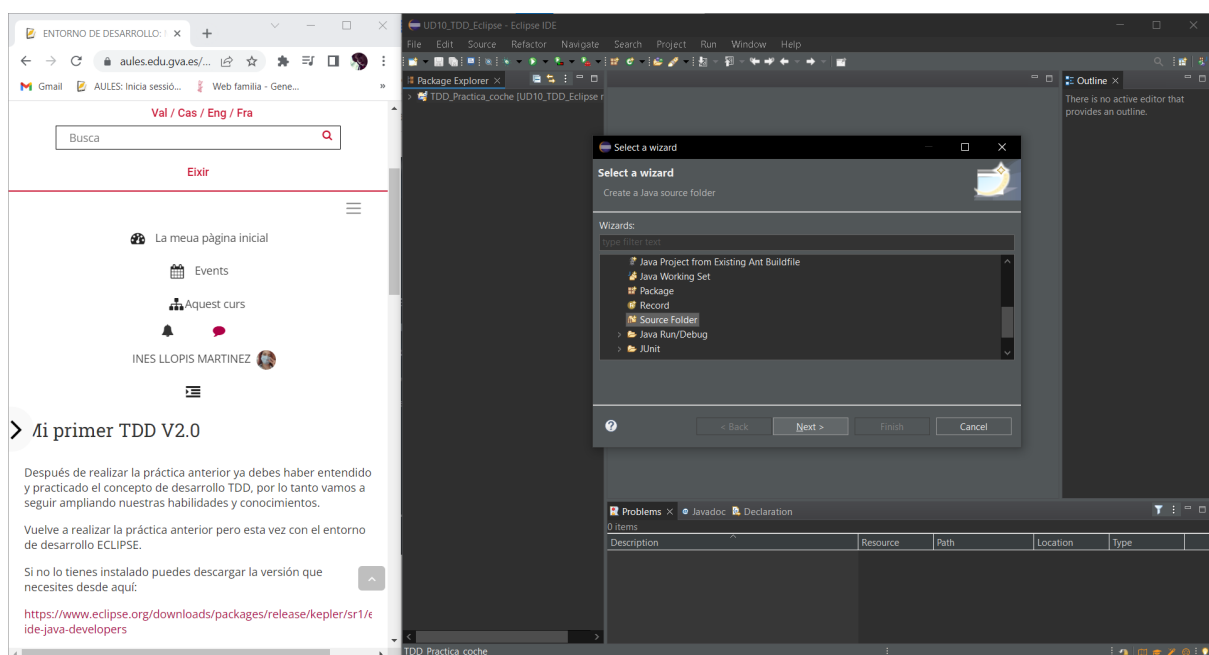
Ejercicio para entregar UD10

Eclipse

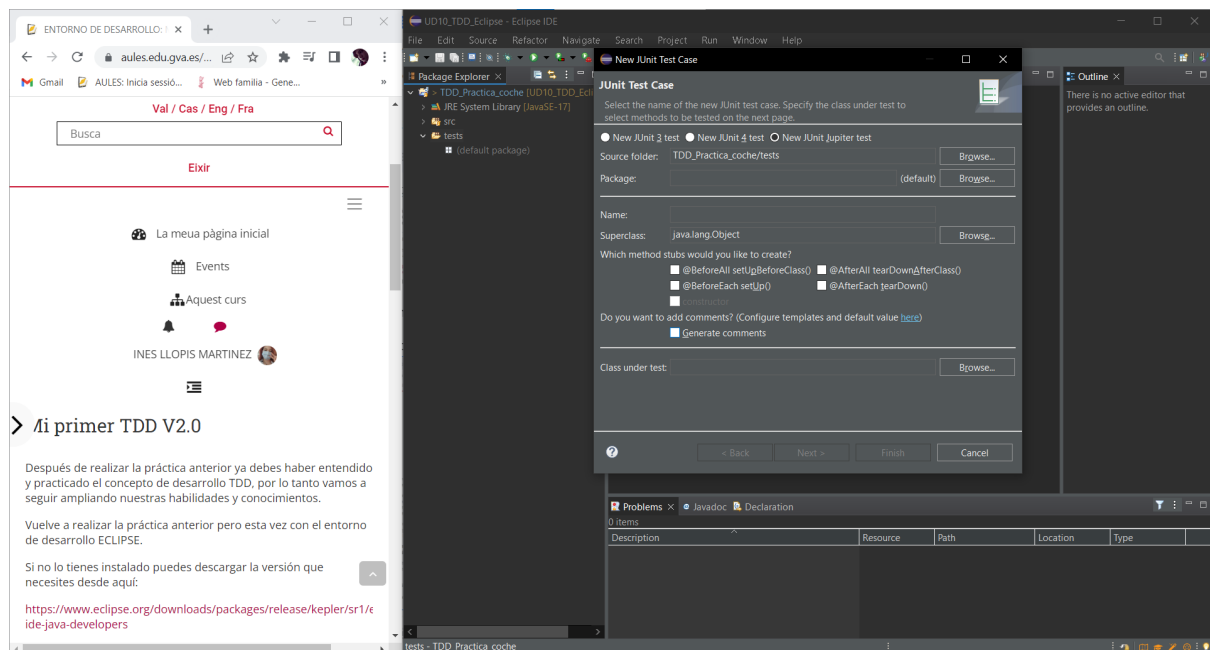
Iniciamos la práctica creando un proyecto llamado “TDD\_Practica\_Coche” en el entorno Eclipse. A diferencia de IntelliJ, Eclipse te obliga a usar barras bajas para separar las palabras de los nombres de los proyectos. Para ello, hacemos clic en “Create new project” en la pantalla de inicio del entorno, escribimos el nombre que nos interesa y hacemos clic en “Finish”. Realizamos un primer commit.



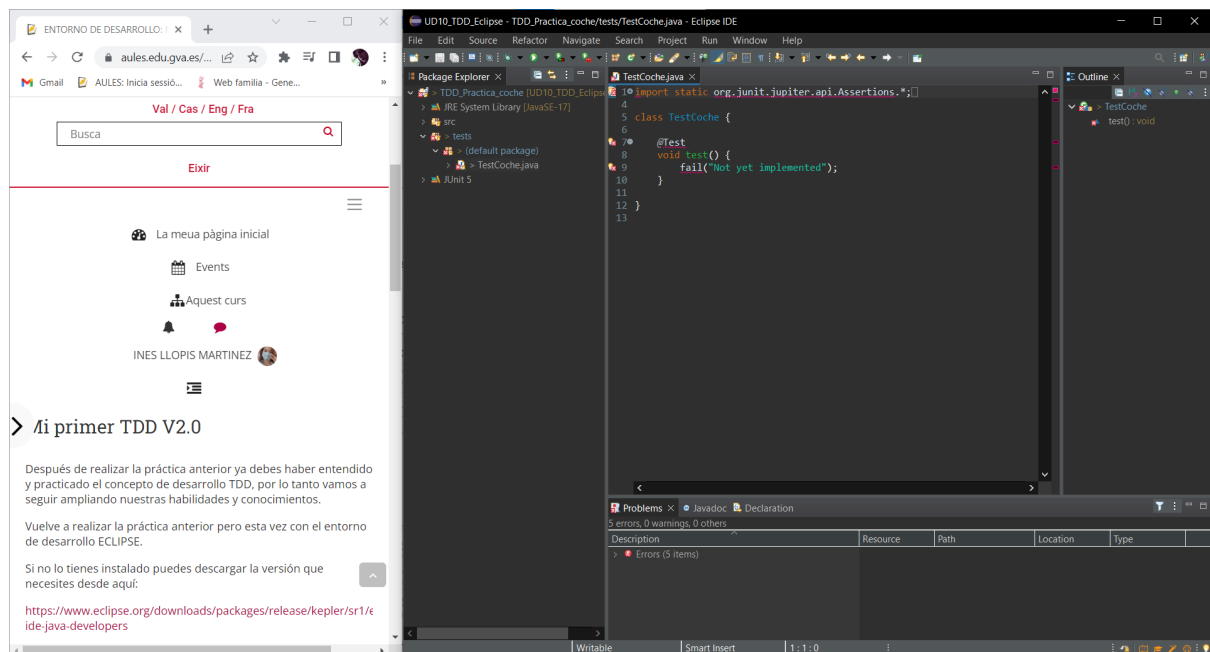
A continuación, creamos un directorio llamado “tests” dentro del proyecto. Para ello hacemos clic derecho sobre el proyecto y seleccionamos “New... > Source folder”. Hacemos commit.



Creamos una JUnit Test Case dentro del directorio llamada “TestCoche” haciendo clic derecho sobre la carpeta “tests” y seleccionando “New... > JUnit Test Case”. Hacemos commit.

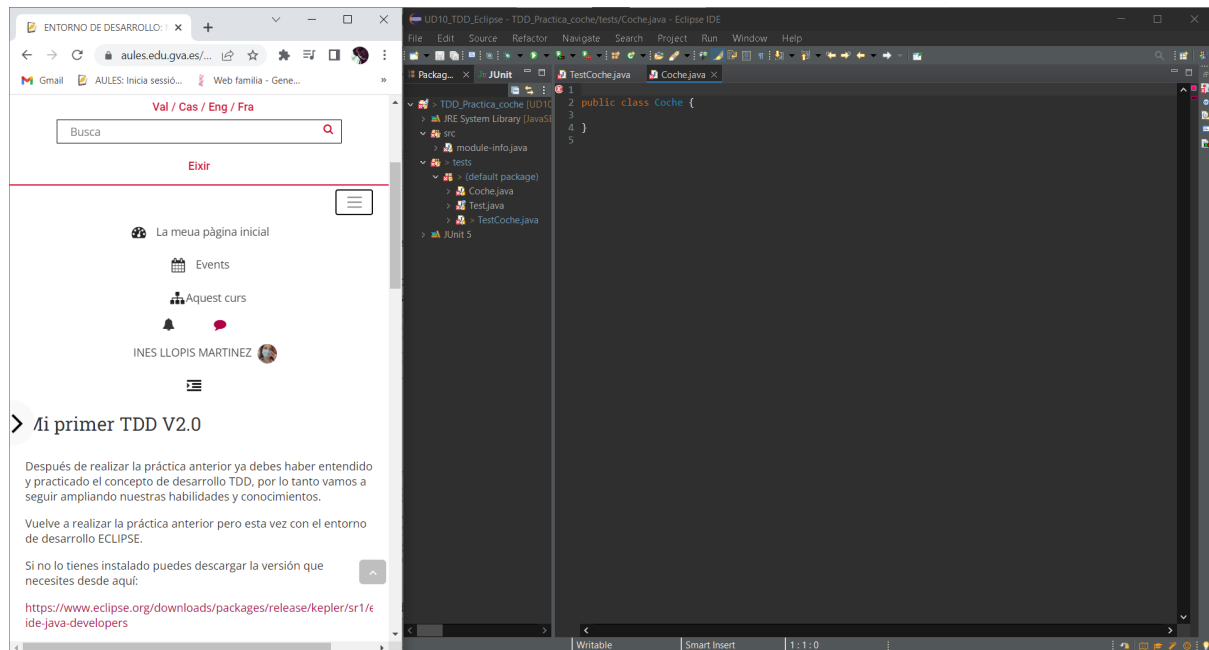


El entorno Eclipse te importa directamente la extensión de JUnit necesaria para realizar los tests.

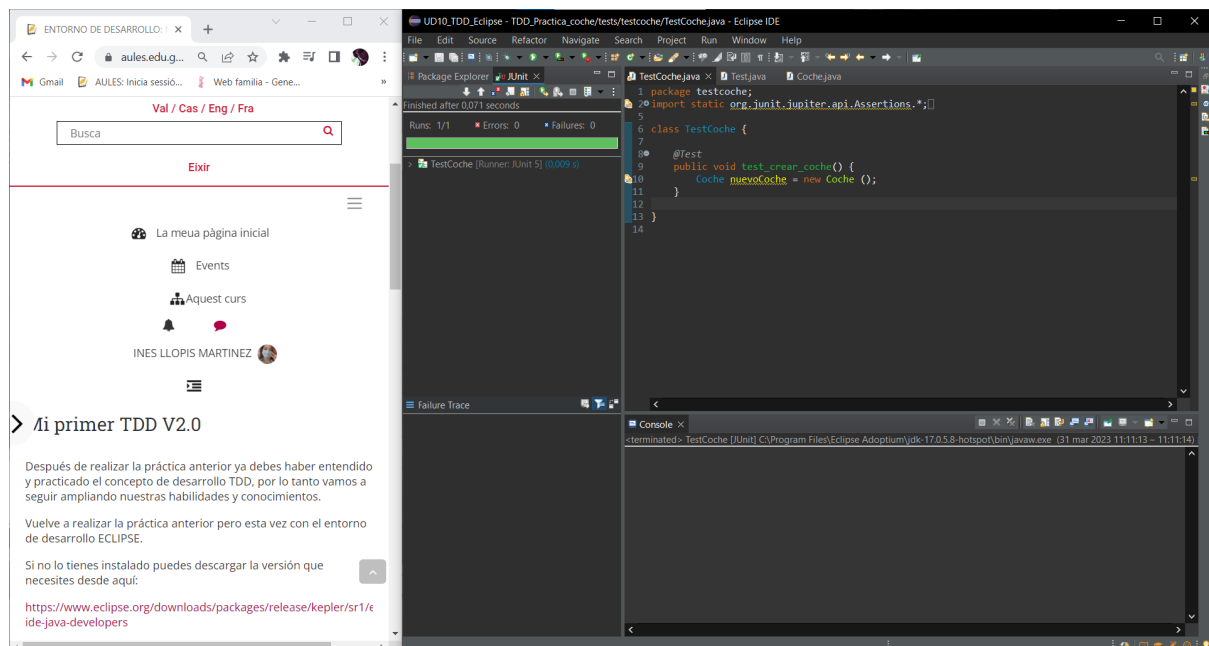


Creamos el método “test\_crear\_coche()”. Hacemos commit.

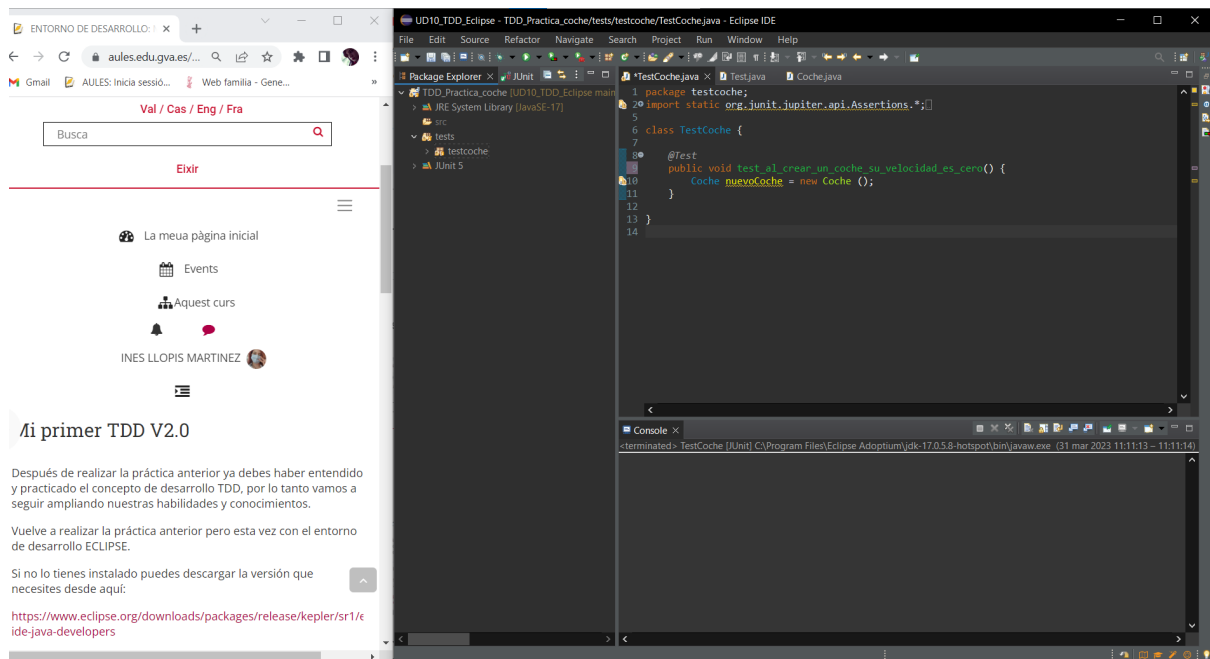
Creamos una clase “Coche” manteniendo el cursor encima de la palabra “Coche” en subrayada en rojo y haciendo clic en “Crear clase Coche”. Hacemos commit.



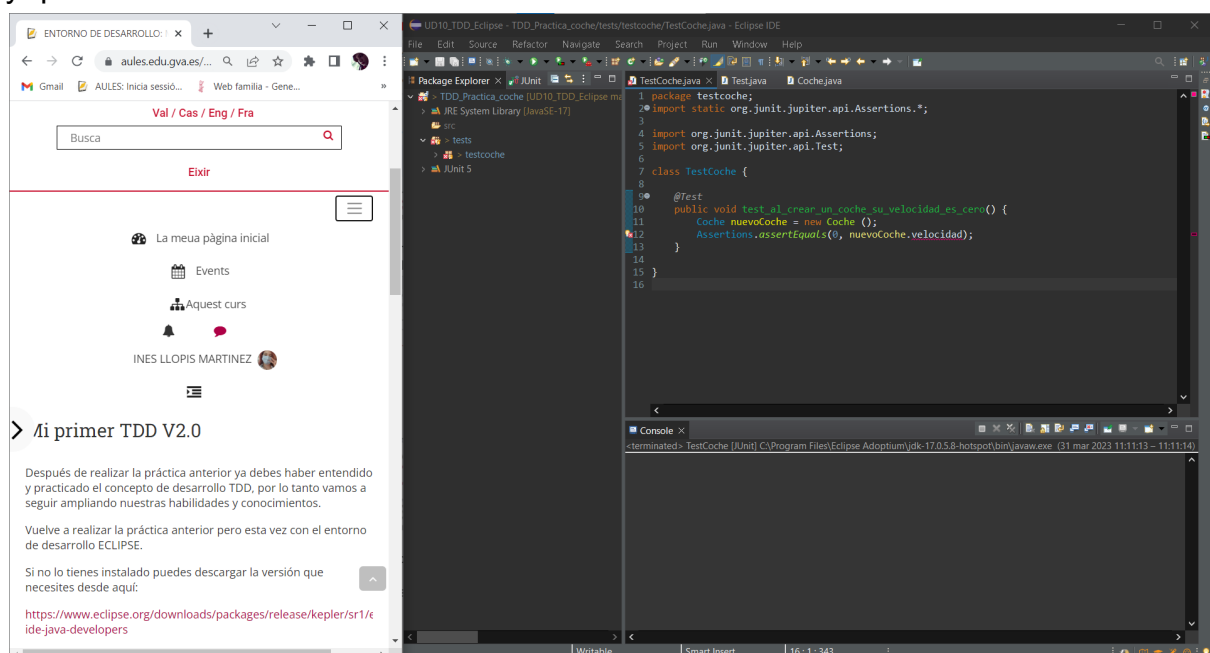
Realizamos una primera prueba para ver si el programa funciona correctamente haciendo clic en “Run”.



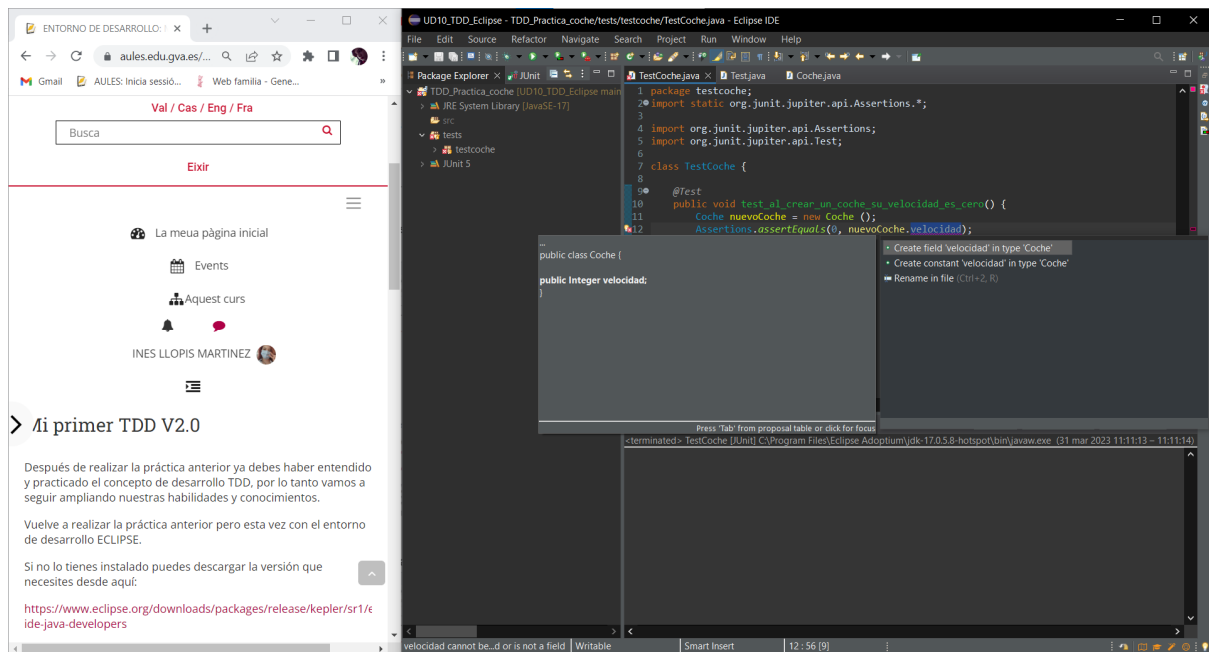
Refactorizamos el nombre del método “test\_crear\_coche” y lo cambiamos por “test\_al\_crear\_un\_coche\_su\_velocidad\_es\_cero” haciendo clic derecho sobre el nombre del método y seleccionando “Refactor > Rename”. Hacemos commit.



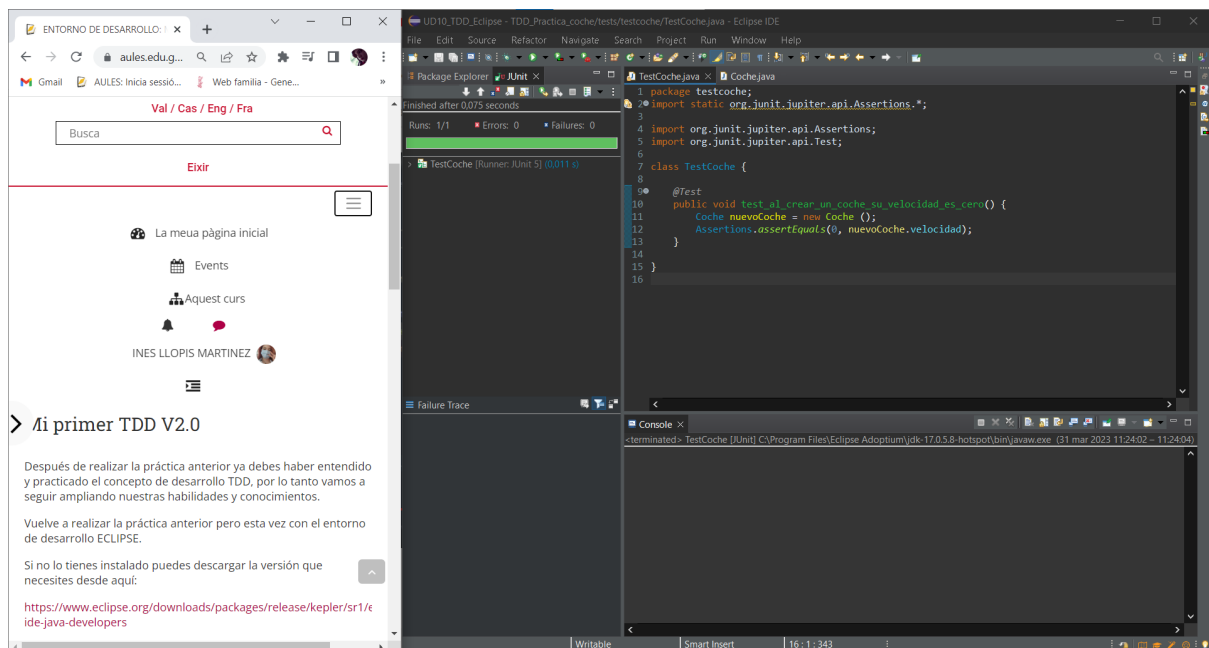
Utilizamos un assertions para establecer que la velocidad esperada del coche será 0 y que el valor real será “nuevoCoche.velocidad”. Hacemos commit.



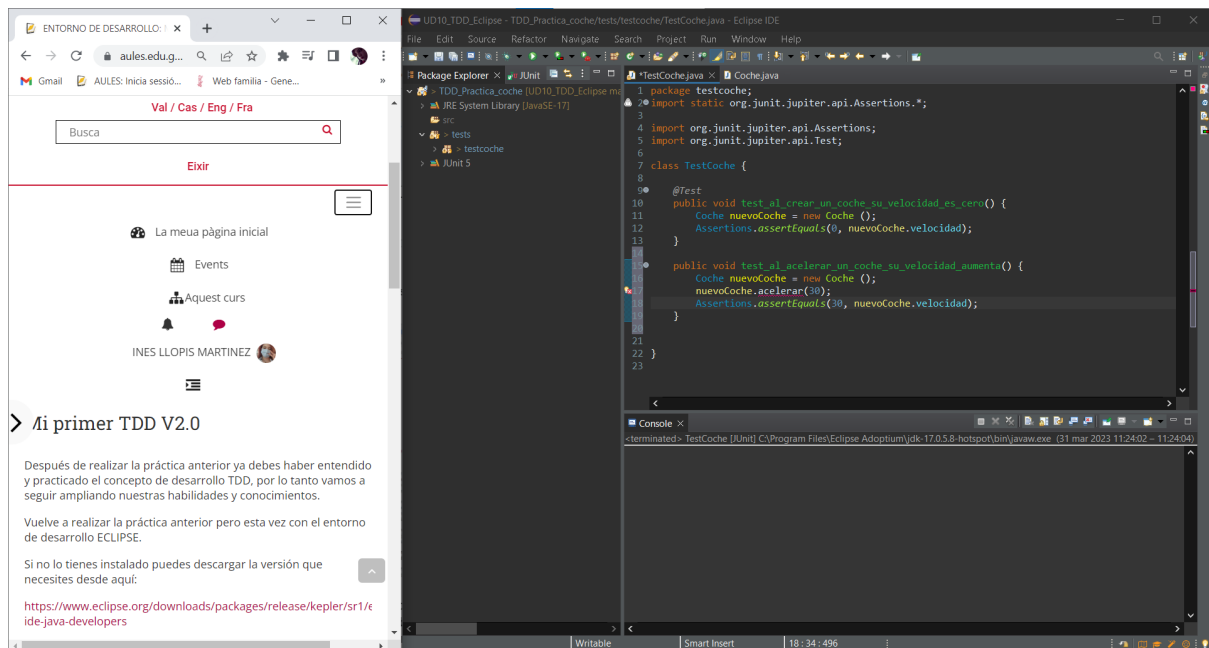
Creamos el elemento “velocidad” dentro de la clase “Coche” manteniendo el cursor encima de la palabra “velocidad” subrayada en rojo y haciendo clic en “Crear nuevo símbolo velocidad”. Hacemos commit.



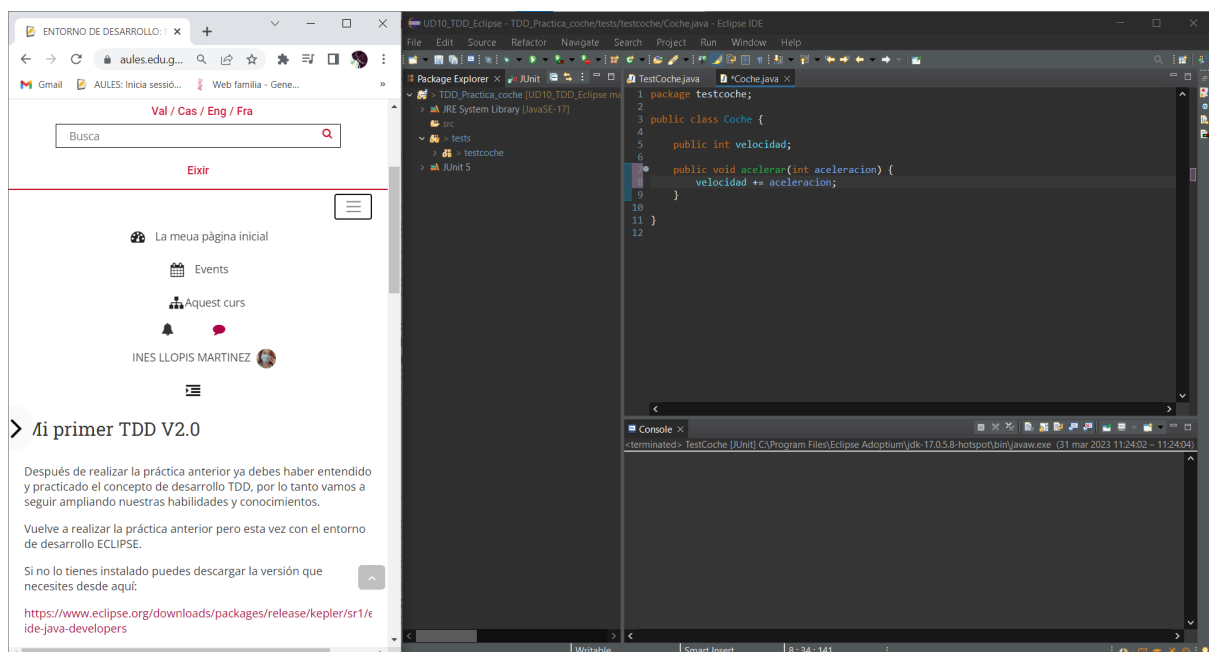
Realizamos una segunda prueba al programa para comprobar que sigue funcionando como debería.



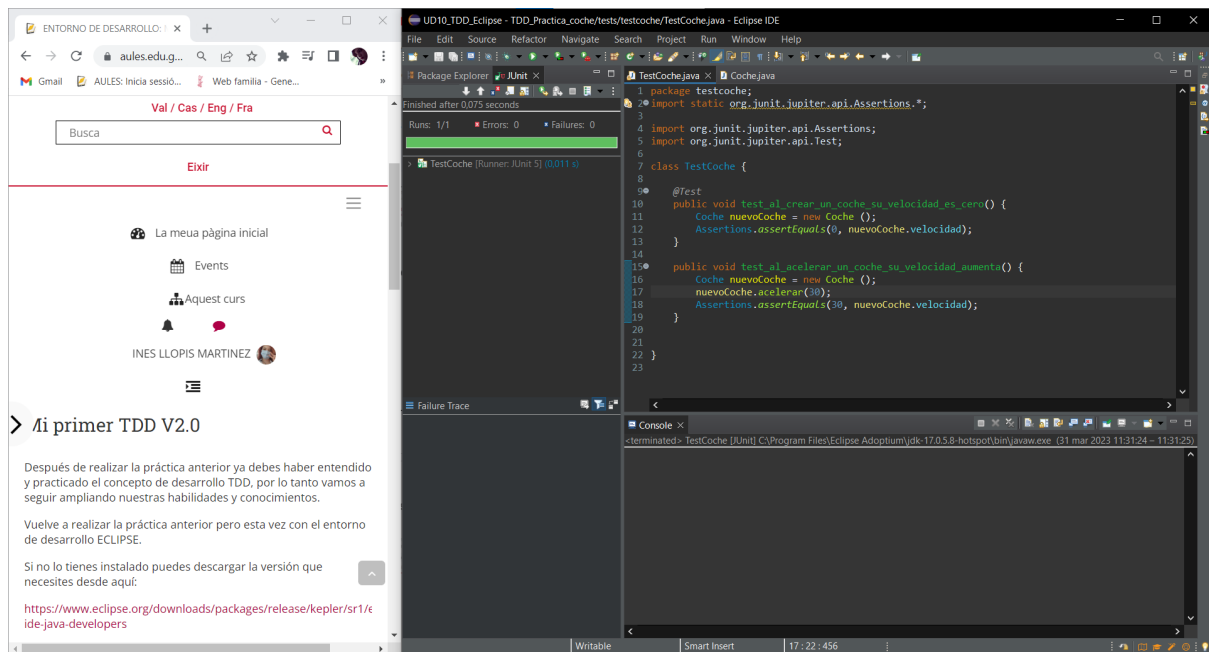
Creamos un nuevo método llamado "test\_al\_acelerar\_un\_coche\_su\_velocidad\_aumenta()". Hacemos commit.



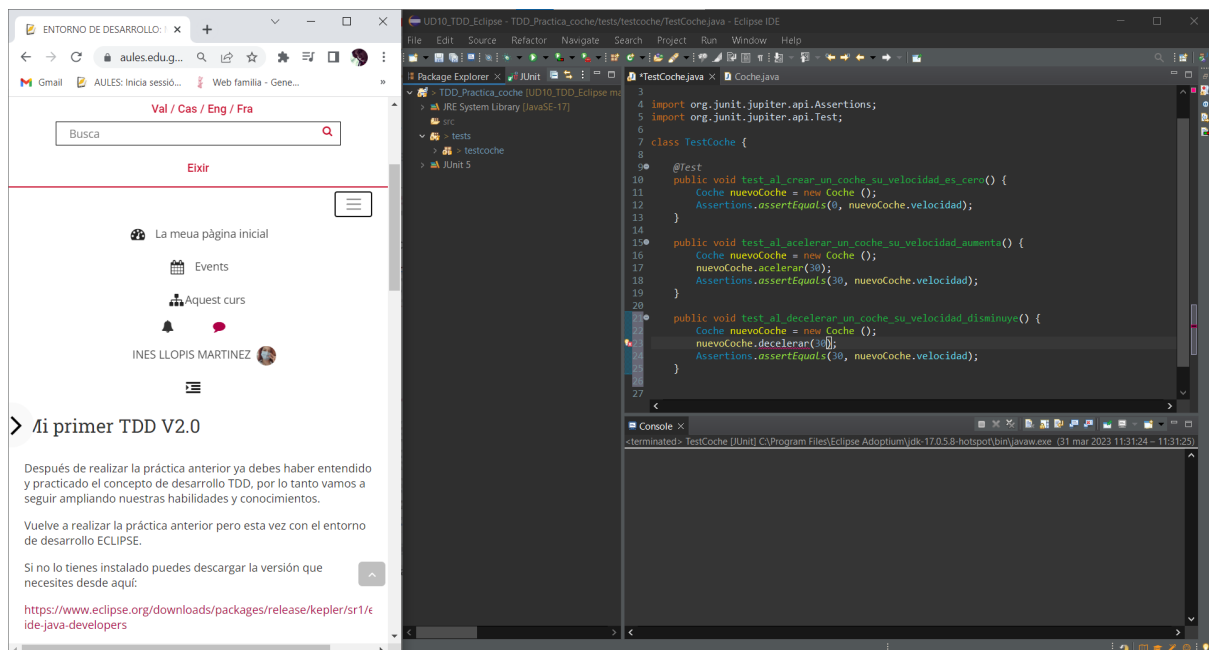
Creamos el método “acelerar” dentro de la clase Coche manteniendo el cursor sobre la palabra “acelerar” subrayada en rojo y haciendo clic en “Crear nuevo método acelerar”. Hacemos commit.



Realizamos una tercera prueba al programa para comprobar que sigue funcionando como debería.

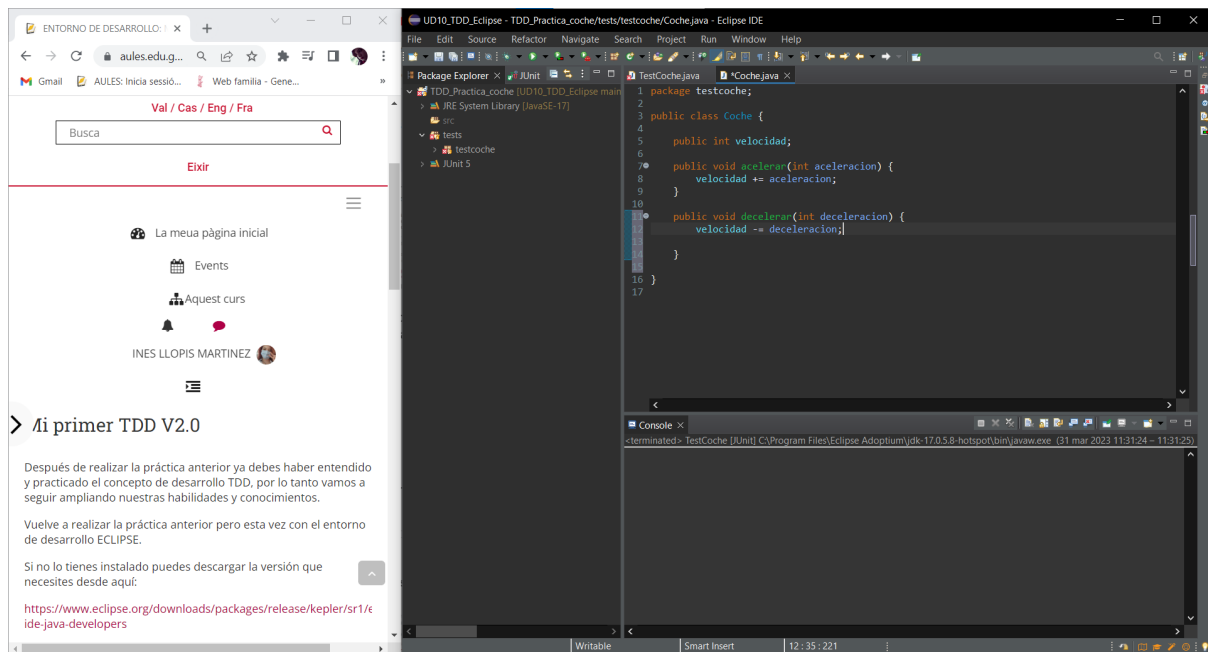


Creamos un nuevo método “test\_al\_decelerar\_un\_coche\_su\_velocidad\_disminuye()”. Hacemos commit.

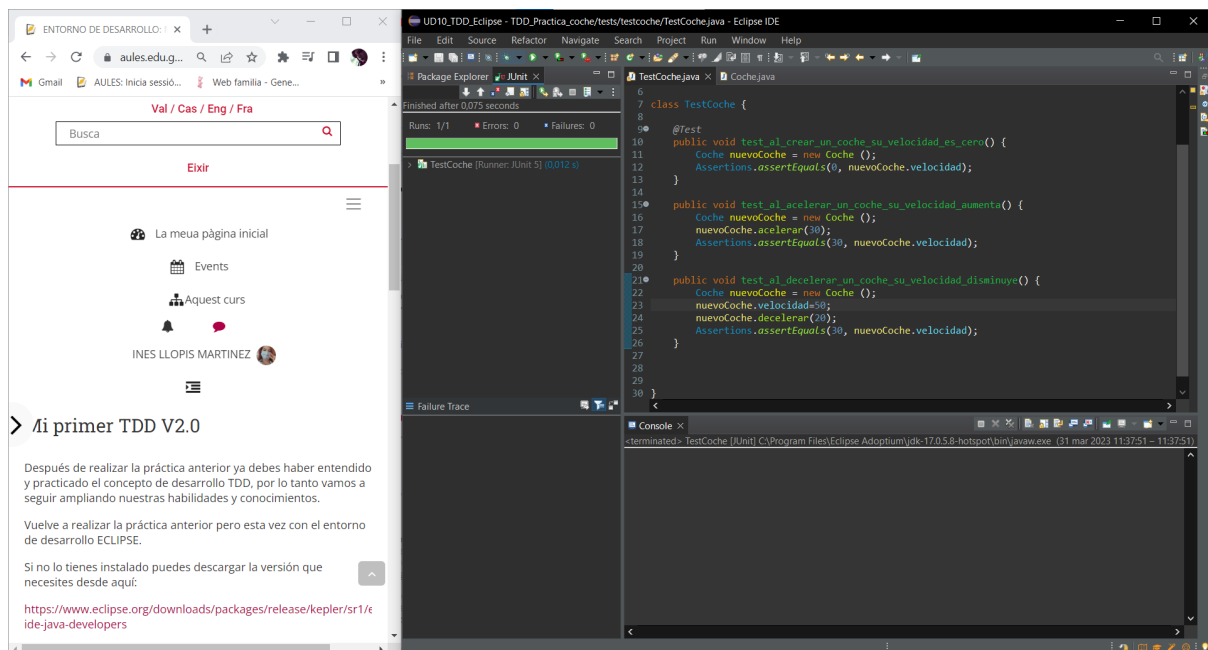


Creamos el método “decelerar” dentro de la clase Coche manteniendo el cursor sobre la palabra “decelerar” subrayada en rojo y haciendo clic en “Crear nuevo método decelerar”. Hacemos commit.

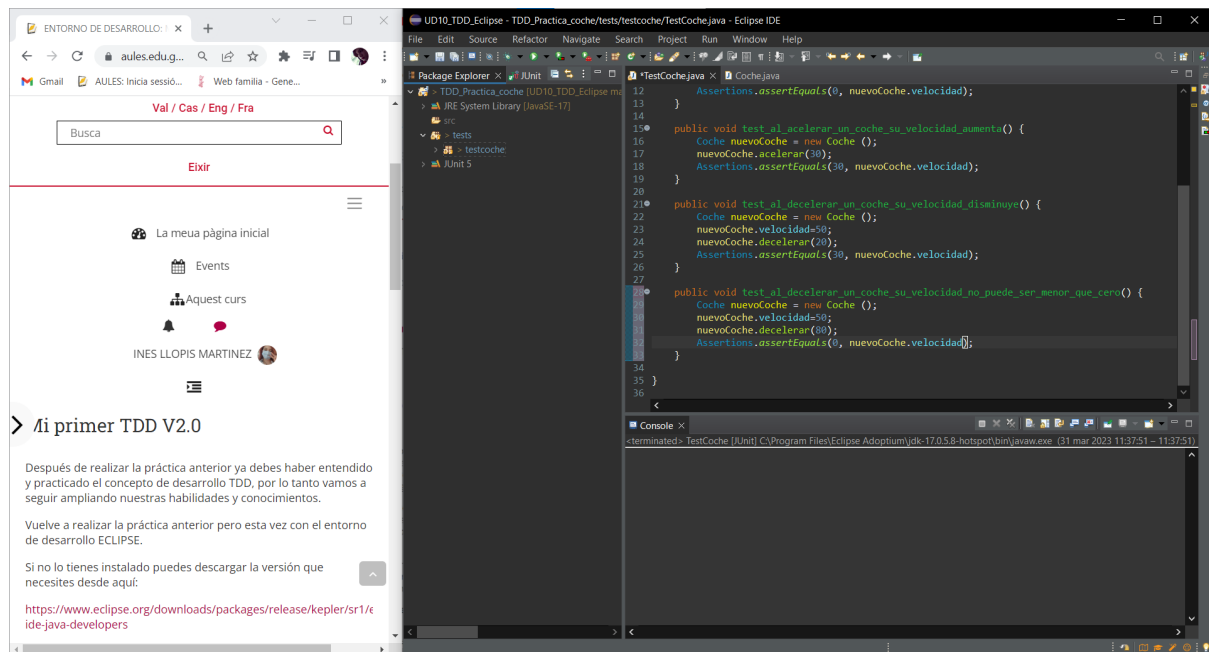




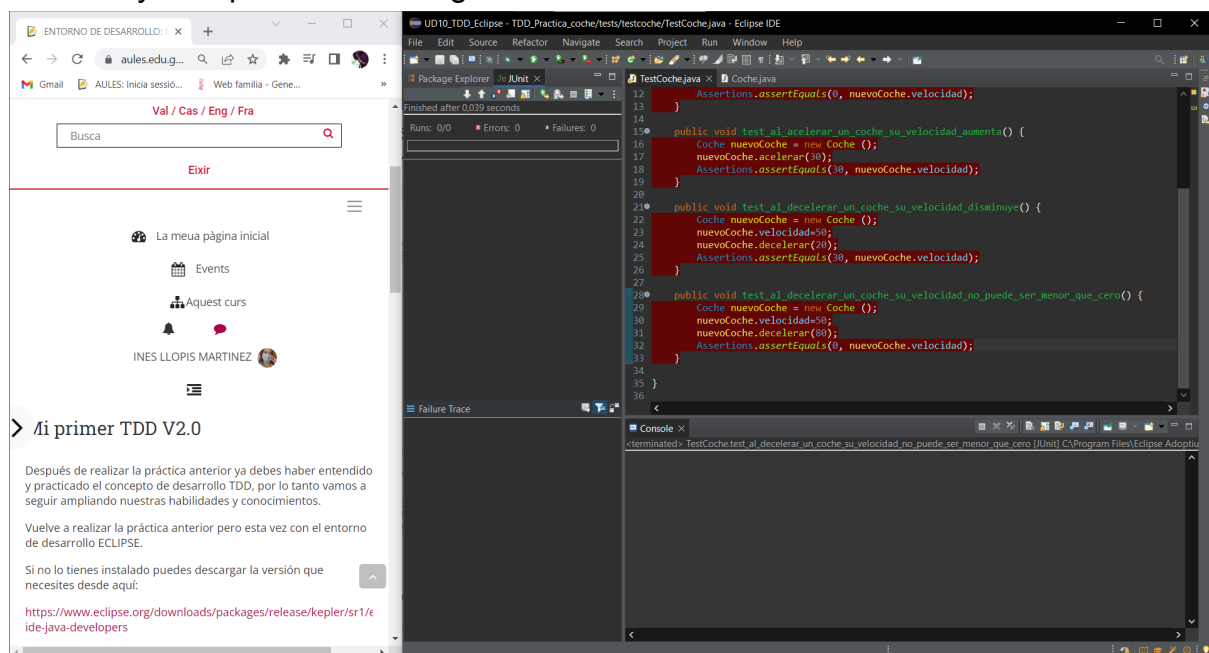
Realizamos una cuarta prueba al programa para comprobar que sigue funcionando como debería.



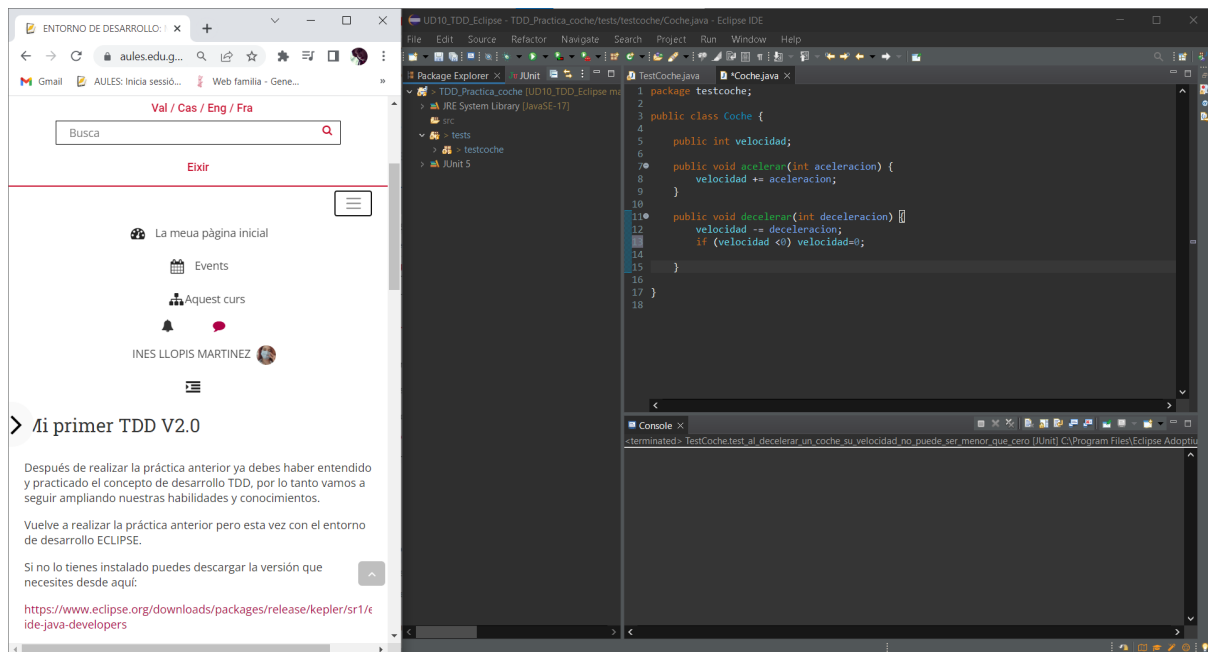
Creamos el método "test\_al\_decelerar\_un\_coche\_su\_velocidad\_no\_puede\_ser\_menor\_que\_cero()". Hacemos commit.



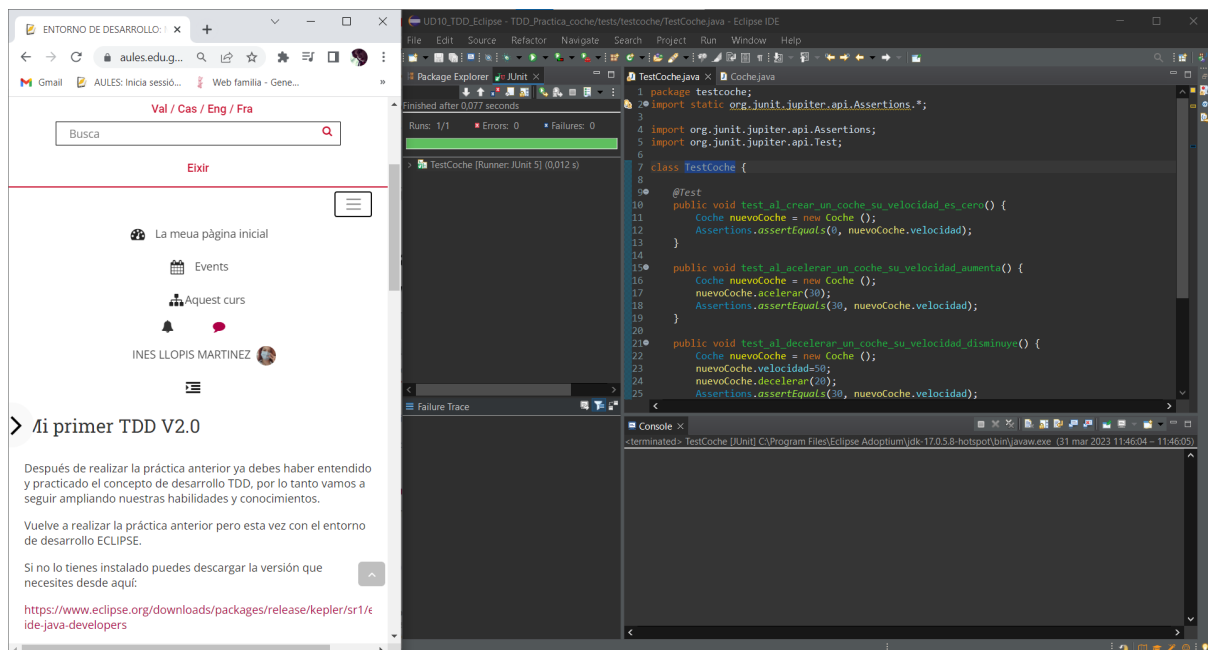
Realizamos una quinta prueba al programa para comprobar que sigue funcionando como debería y vemos que no se ha superado la prueba porque el resultado obtenido y el esperado no son iguales.



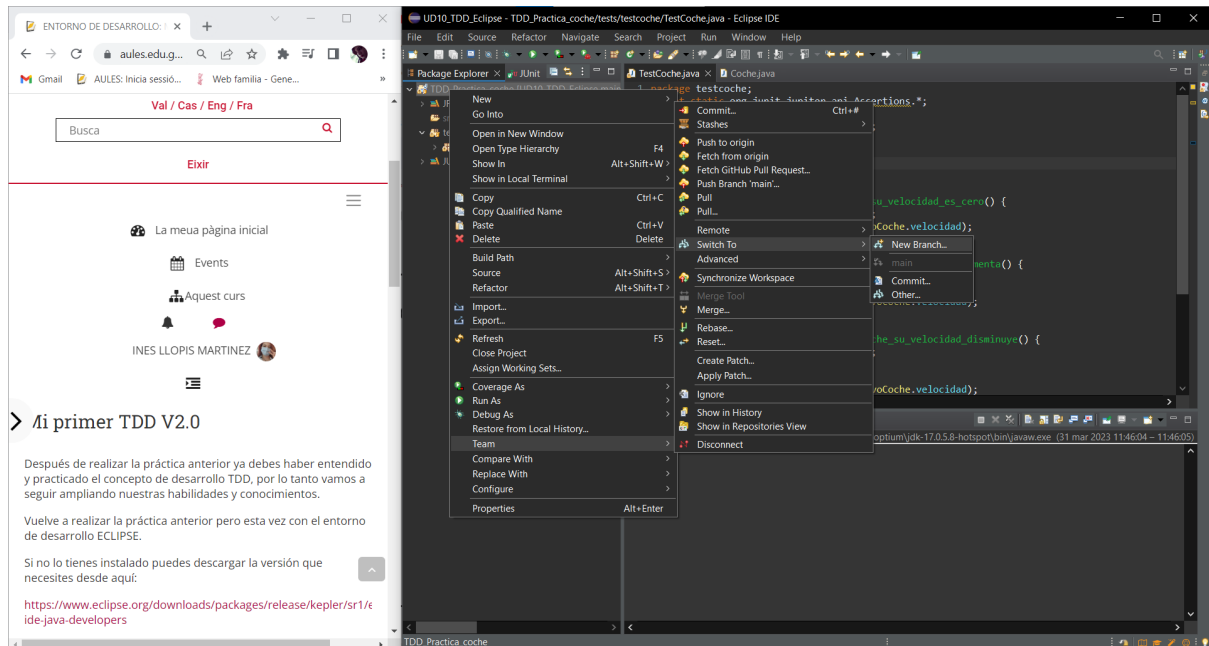
Establecemos el límite de deceleración en 0 mediante un *if* en el método “decelerar” de la clase Coche. Hacemos commit.



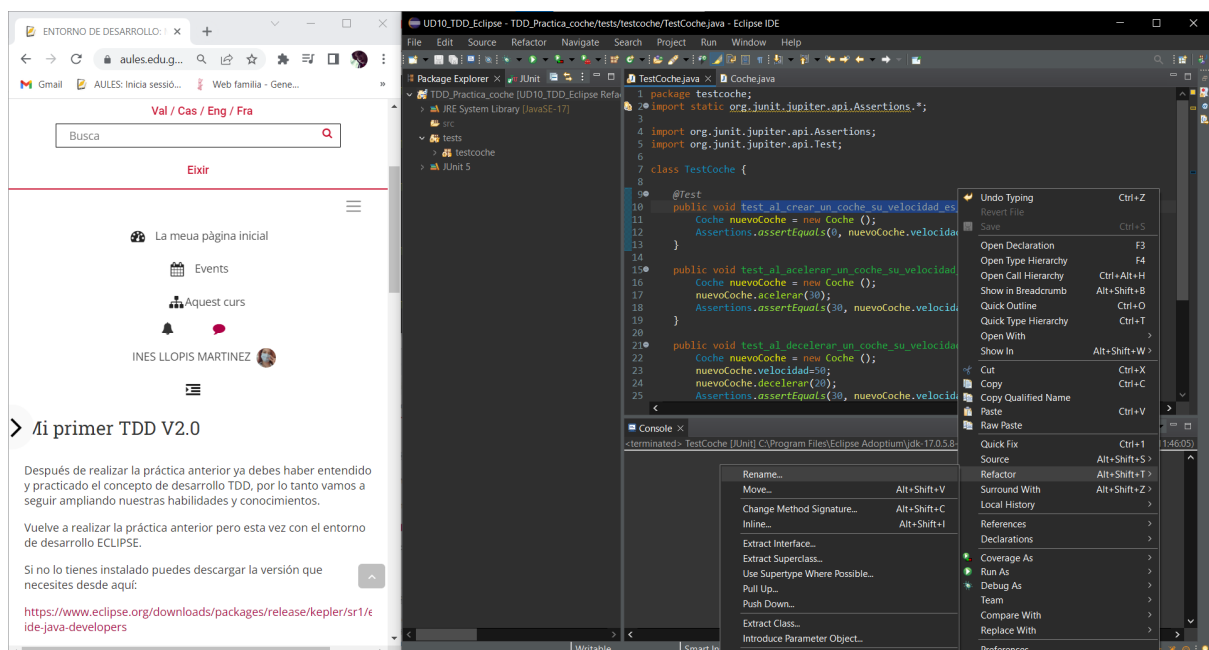
Realizamos una sexta prueba al programa para comprobar que sigue funcionando como debería y vemos que se pasan los 4 tests correctamente.



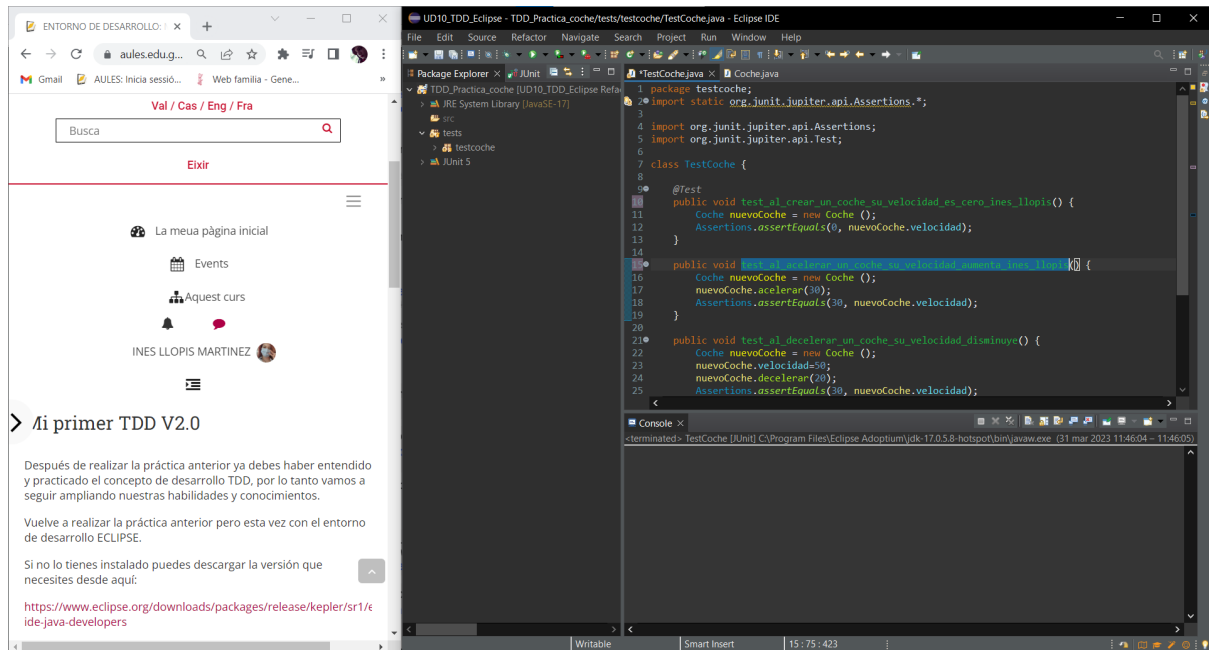
Creamos una nueva rama llamada "Refactorizado" haciendo click derecho sobre el proyecto y seleccionando "Team > Switch to... > New Branch...".



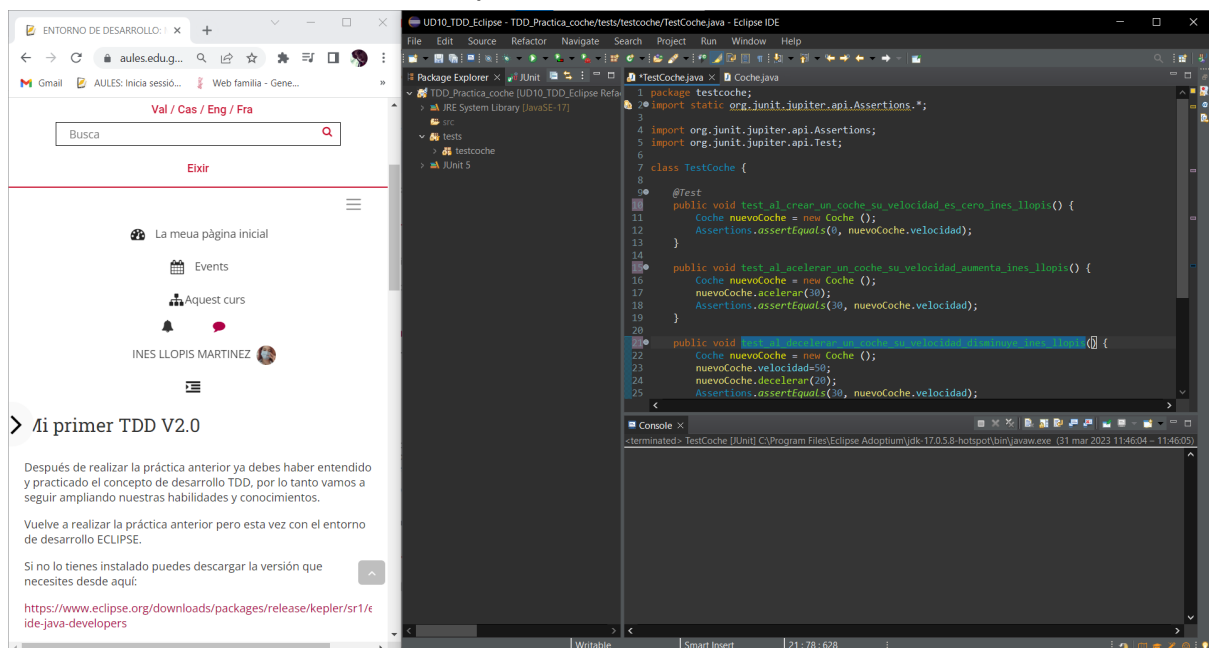
Refactorizamos el primer método para añadir nuestro nombre al final del mismo haciendo clic derecho sobre el nombre del método y seleccionando “Refactor > Rename...”.



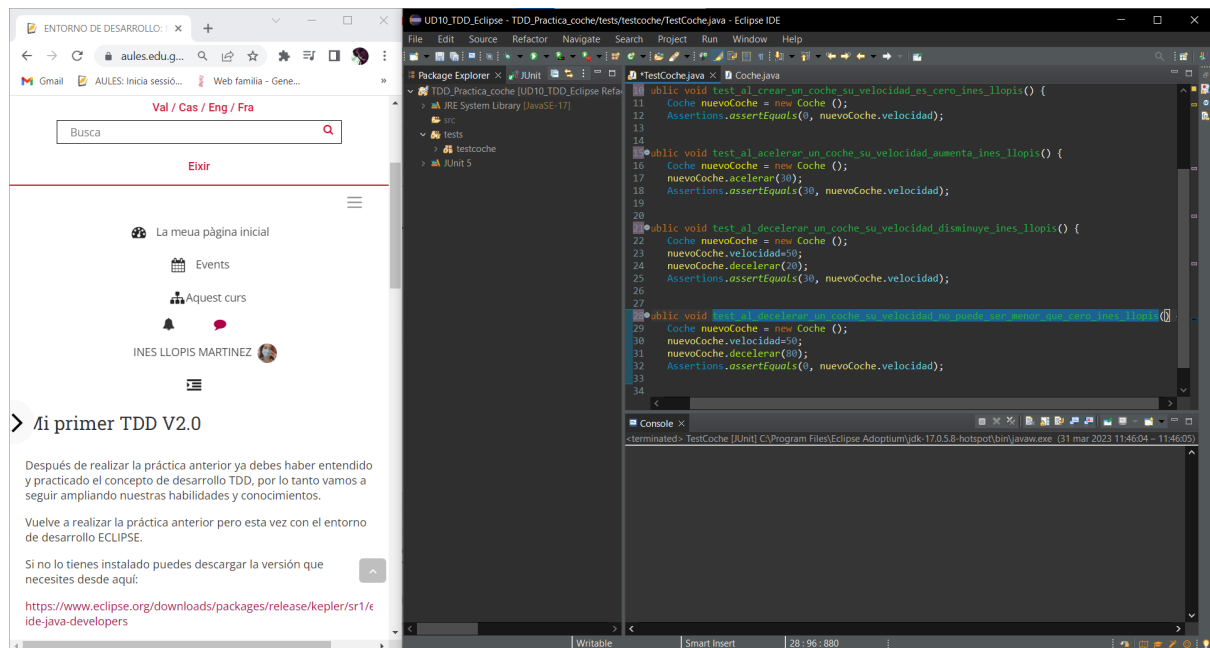
Refactorizamos el segundo método pulsando Alt+Shift+T.



## Refactorizamos el tercer método pulsando Alt+Shift+T.



## Refactorizamos el cuarto método pulsando Alt+Shift+T.



Refactorizamos los dos métodos de la clase Coche pulsando Alt+Shift+T.

