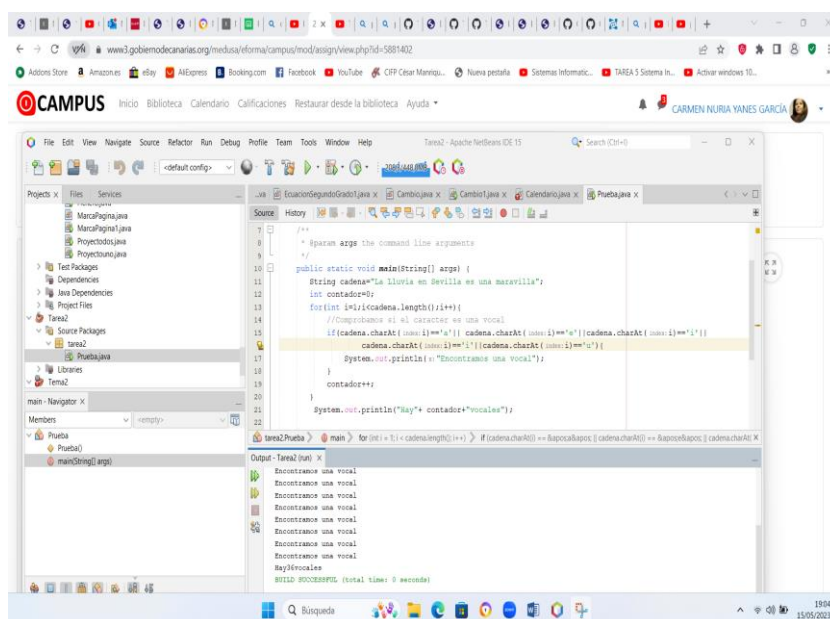


1. Ejercicio. Depurar código.

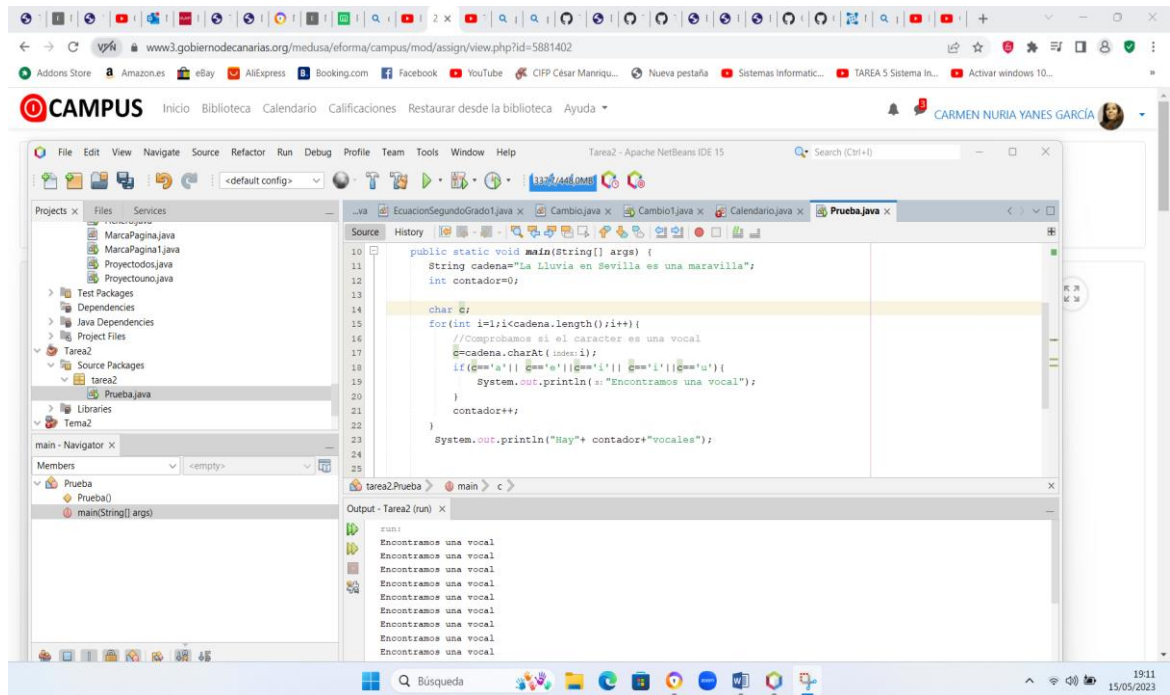
Lo primero que hacemos es picar el código e intentar ejecutarlo

Los errores que he detectado son los siguientes:

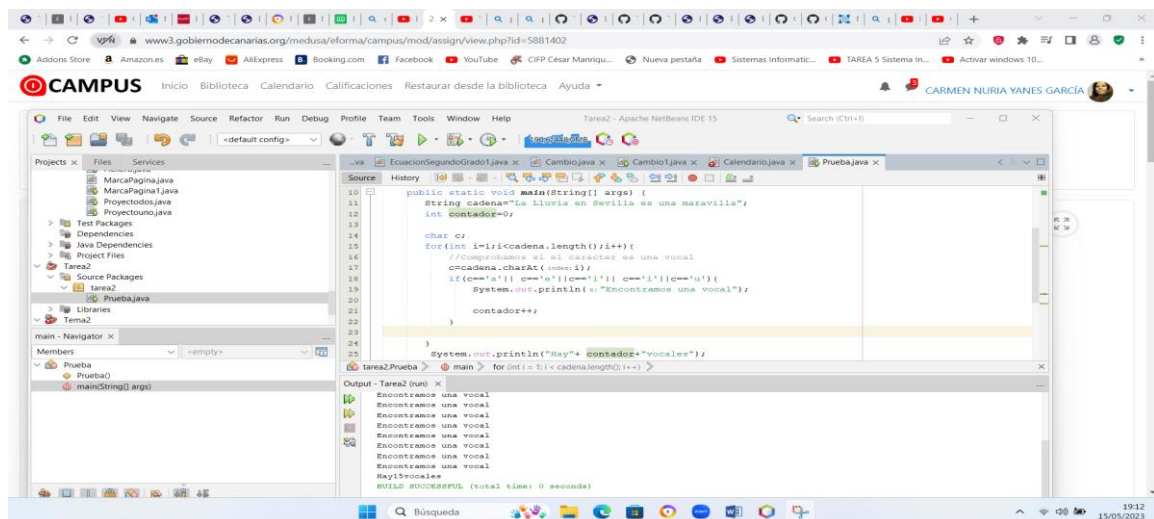
1. En primer lugar observo que no se hace un buen uso del operador de comparación utilizado un "=" cuando en este caso lo que hay que utilizar es "=="

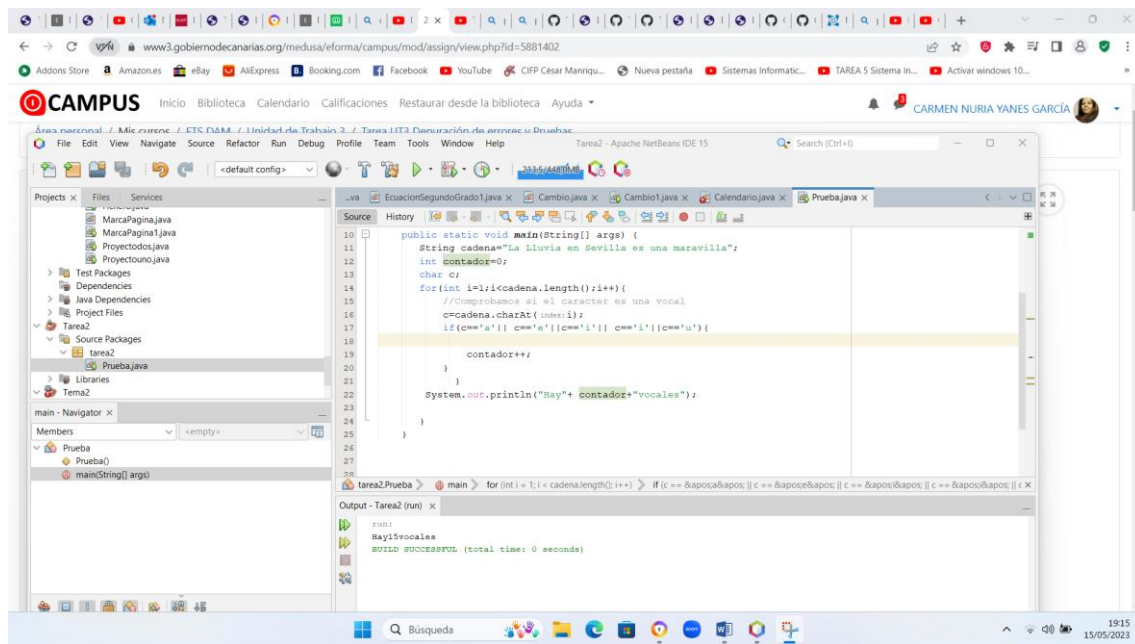


- En este caso es conveniente declarar una variable de tipo char primero y luego inicializándola dentro for. Le hacemos pasar la cadena para que cada vez se ejecute el for vaya detectando si existe alguna vocal y además cuente el número de vocal que existen



- Por último, quitamos el `System.out.println("Encontramos una vocal")` porque lo que nos está haciendo es que cada vez que encontremos una vocal nos salga por teclado la frase y no es eso lo que buscamos. Nos basta con el último System que nos saca por consola el computo total

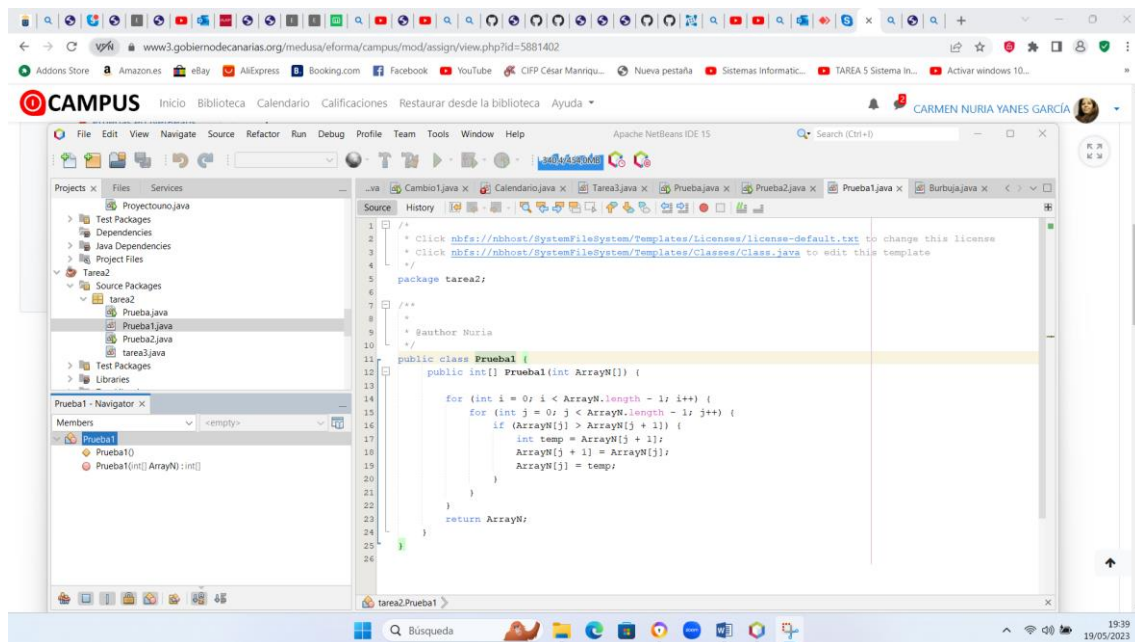




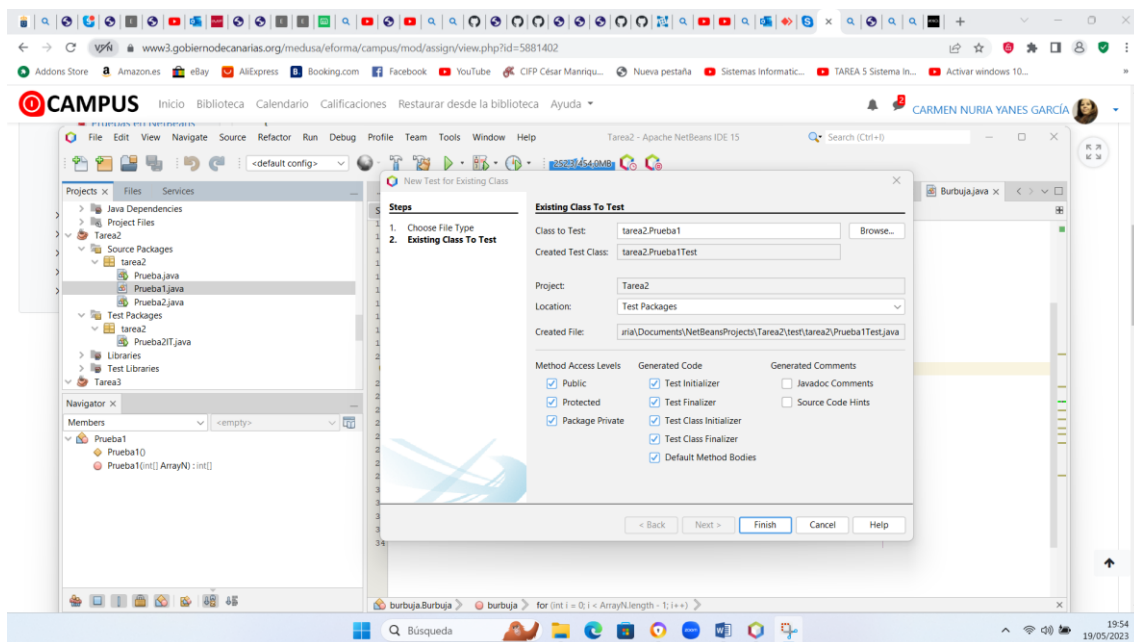
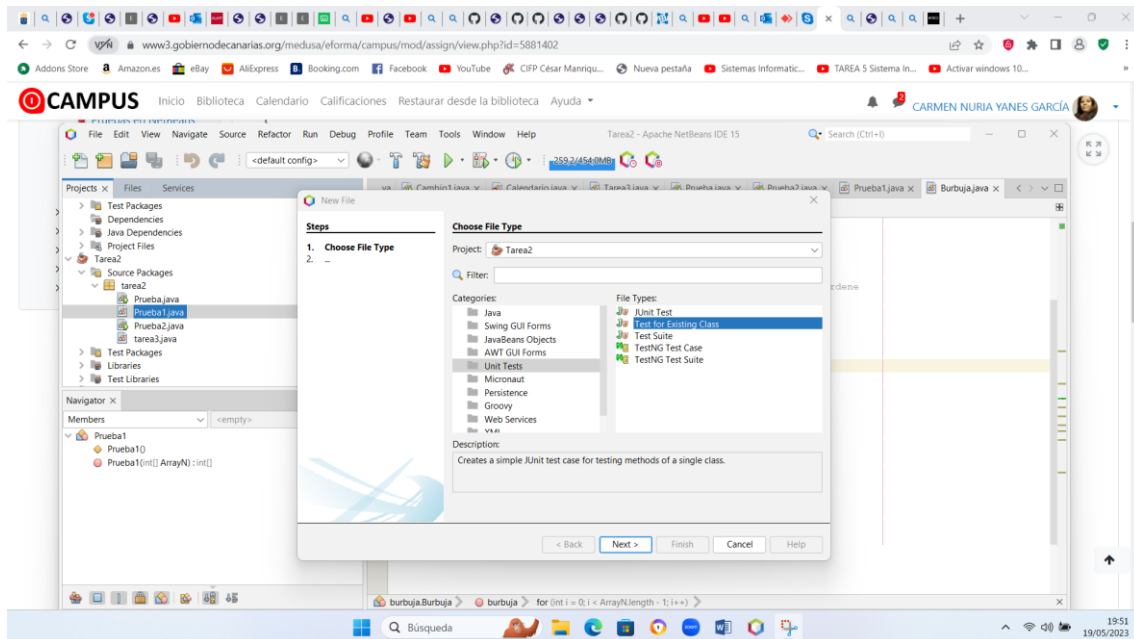
2. Ejercicio. Elaboración de serie de pruebas unitarias

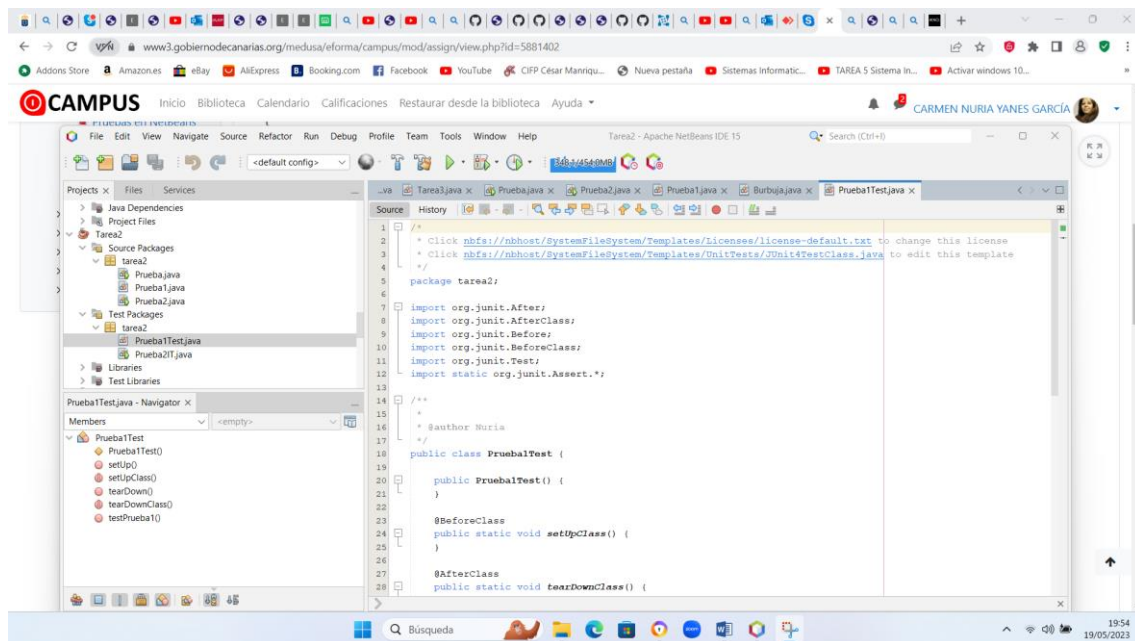
En este caso las pruebas unitarias se basan en el método de la burbuja. Este algoritmo utiliza lo que yo llamo una “pasada”, es decir, un recorrido completo a través del arreglo a ordenar. Se comparan los datos que se encuentran en las casillas adyacentes (por ejemplo, el de la casilla 0 y el de la casilla 1) y se cambian de orden si es necesario. Así se continúa hasta recorrer todo el arreglo.

1. En primer lugar, creamos la clase Prueba1 y escribimos el código

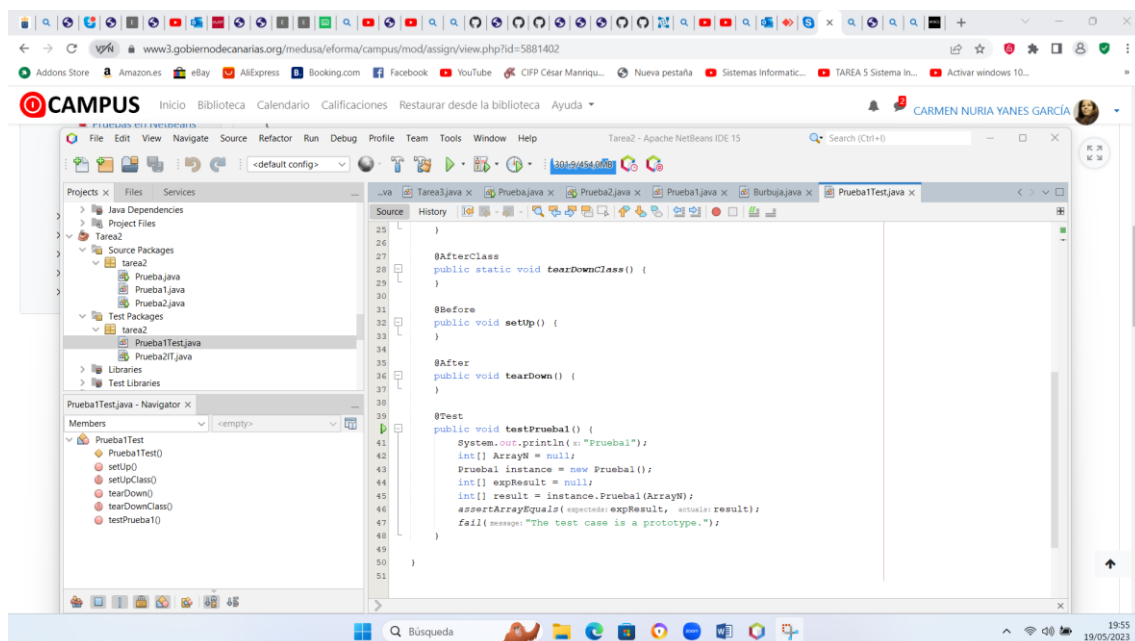


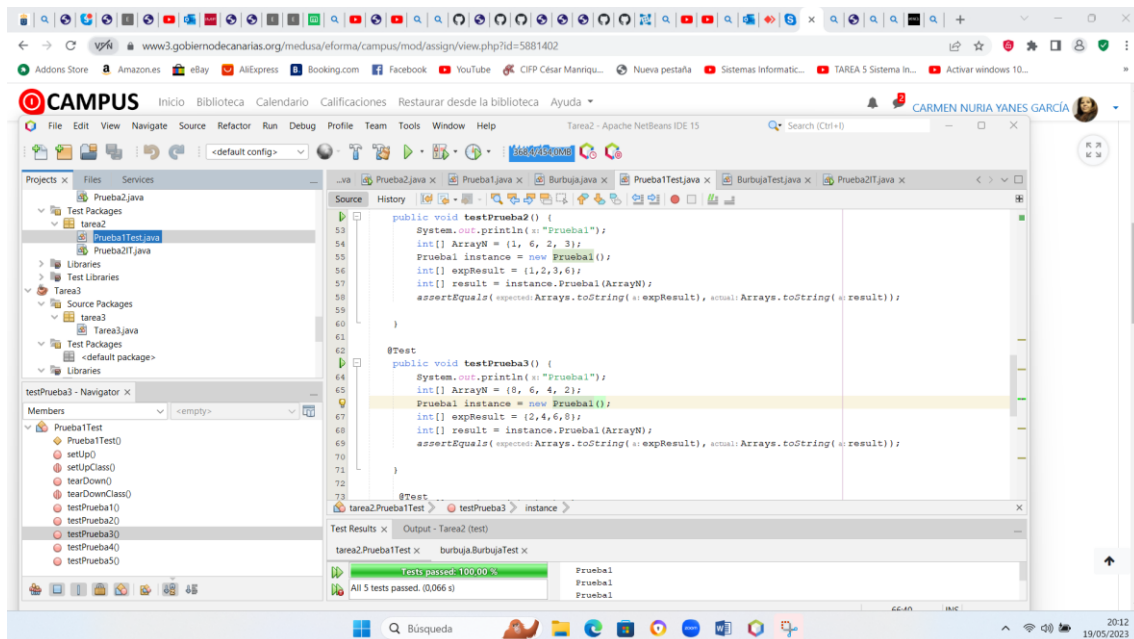
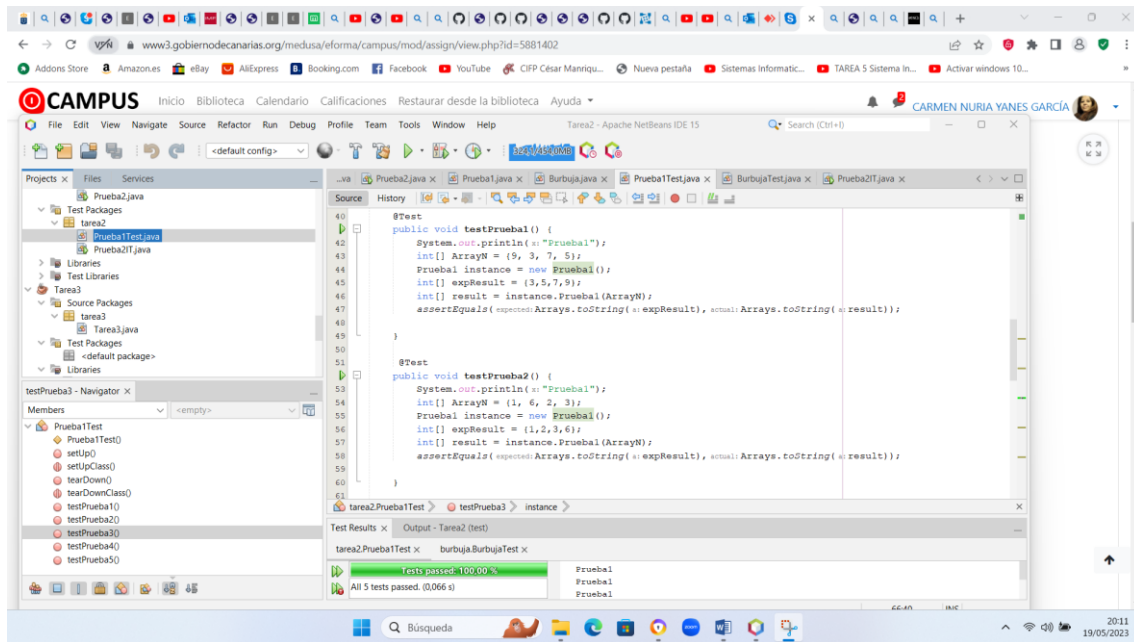
2. Para llevar a cabo las pruebas en el entorno de NetBeans picamos sobre la clase , New File y luego habilitamos la herramienta habilitada para la realización del ejercicio

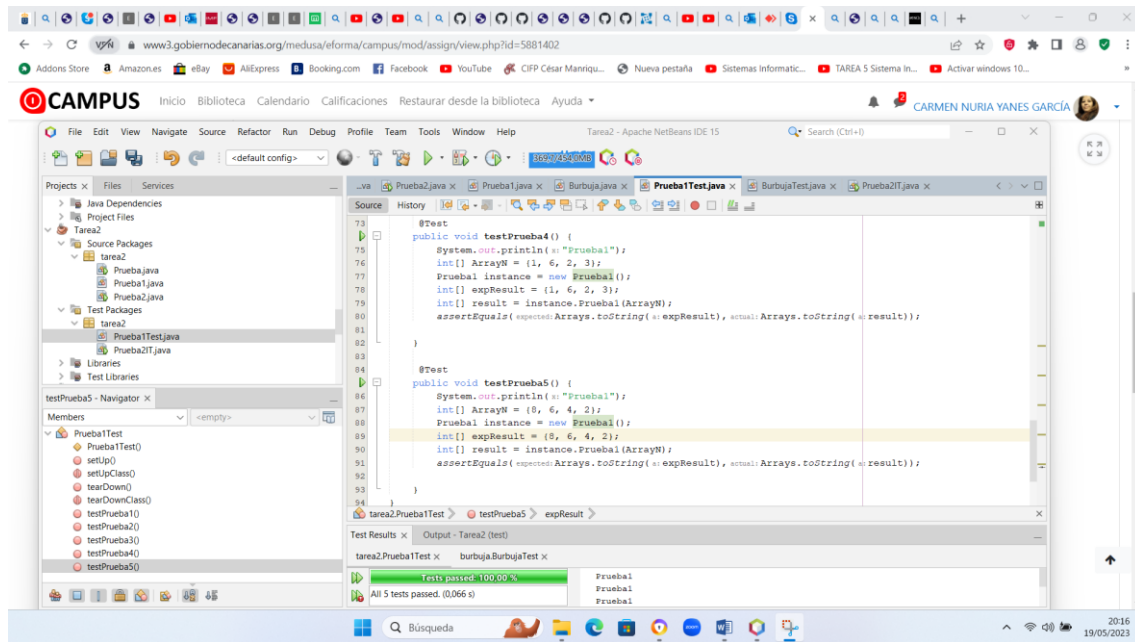




Automáticamente se importan los métodos que necesitamos







En los dos últimos casos pasamos, esta vez, el código no esperado y vemos como al pasar el test nos da error

