

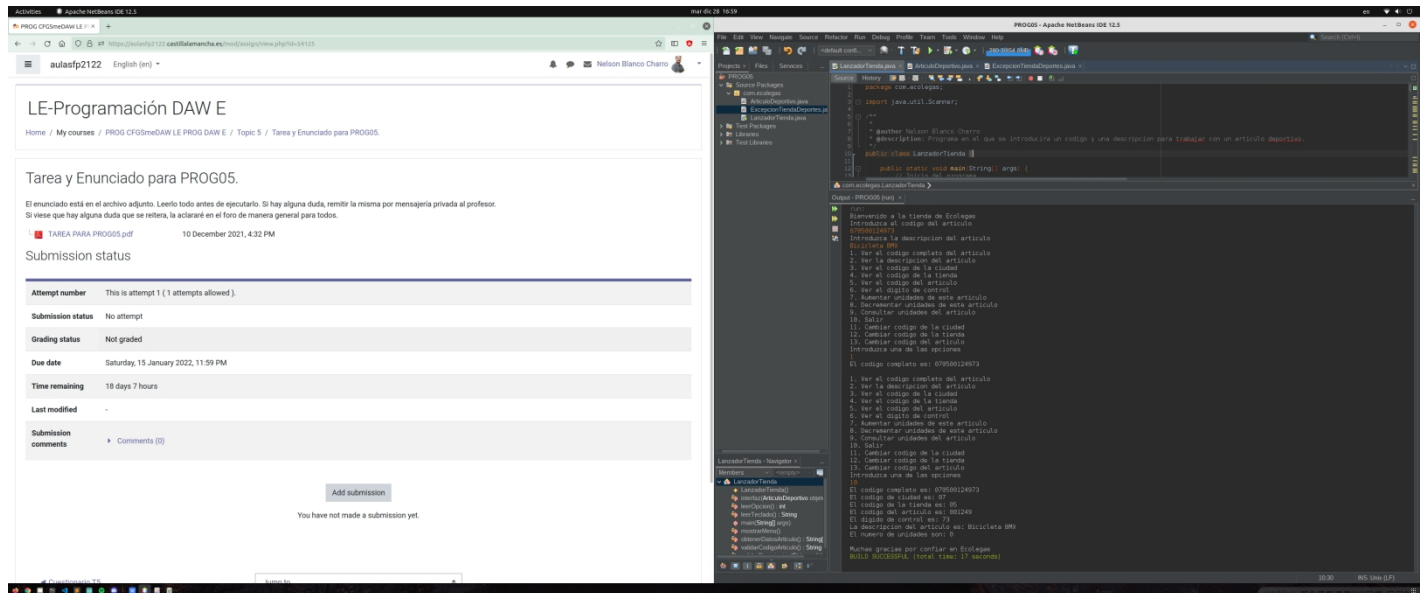
Dudas

Aunque la tarea funciona correctamente tengo una duda con respecto a la captura de excepciones:

La captura de excepciones de forma manual (ExcepcionTiendaDeportes.java) no me queda claro, consiste en lanzar un mensaje de error en el método y luego, fuera del método capturarlo y mostrarlo?

Por ejemplo, yo en esta tarea en la clase “ArticuloDeportivo.java” en cada uno de los metodos lanzaba errores cuando yo sabia donde podía fallar (al introducir un numero negativo o al introducir una letra en lugar de un numero) y en el Main capturaba esta y otras excepciones, solo con un try-catch, pero no me queda claro si esta es la forma adecuada de utilizar las excepciones.

Captura de pantalla donde se puede ver el correcto funcionamiento del programa



Aclaraciones sobre el programa

En el enunciado aparecen 10 opciones, he añadido 3 mas para hacer uso de los metodos de cambiar ciudad, tienda y código del articulo, recalculando el dígito de control y el código completo del articulo.

En el método “validarCodigoArticulo” he optado por validar el código de 12 dígitos mediante un string ya que creo que es mas sencillo calcular la longitud de un string que de un numero ($\text{int length} = (\text{int})(\text{Math.log}_{10}(\text{numero})+1)$); además, trabajando con números es posible perder el 0 de la izquierda del todo (070500124973) y que obtengamos errores, usando un string se mantendrían siempre los 12 dígitos.

En los metodos iniciales de introducción del código del articulo y descripción, he utilizado condicionales IF para poder salir del programa, en el que si el usuario escribe “s” o “S” en cualquiera de las opciones, el programa finaliza. Me parece correcto utilizar un método de finalización del programa antes incluso de mostrar el menú, porque, por ejemplo, se podría entrar en un bucle cuando el usuario no tenga el código de 12 dígitos y el programa lo solicite continuamente y no poder cerrar el programa.

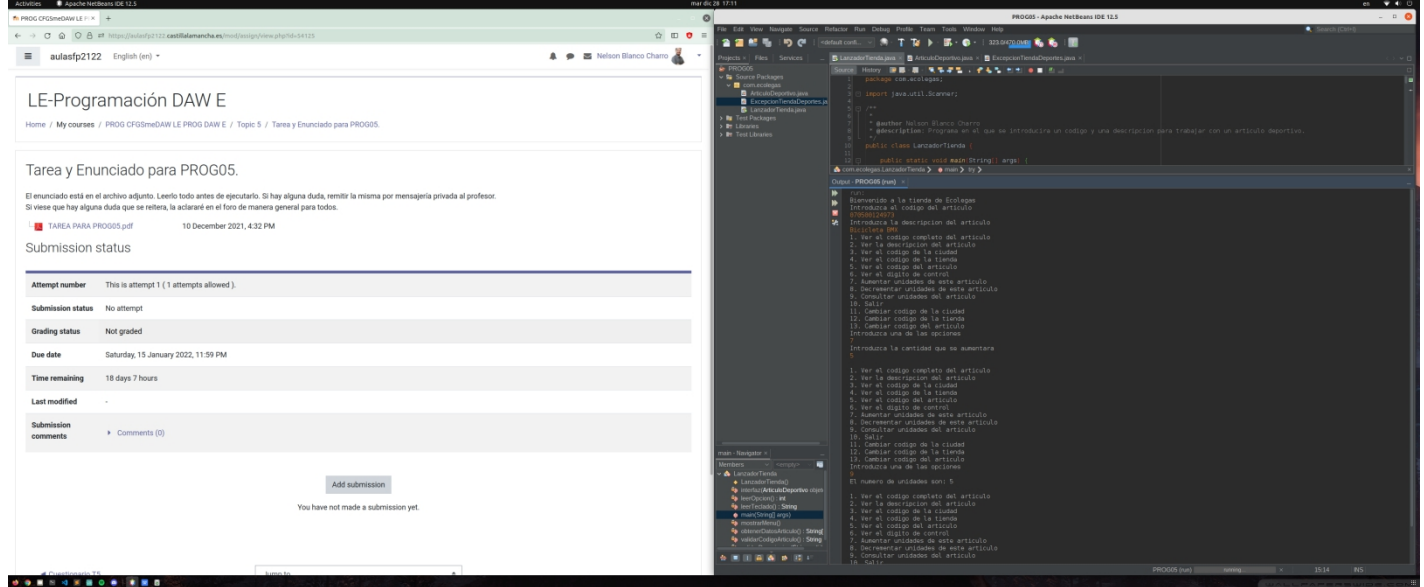
En los metodos de validar el código del articulo y de la descripción, he creado bucles WHILE para solicitar la información correcta cuando el usuario se equivoque o introduzca datos no validos.

En el método toString() de la clase “ArticuloDeportivo.java”, que yo he llamado verTodo(), en lugar de dar una única cadena de texto con la información del articulo, he reutilizado el código ya escrito en otros metodos (verDescripcion(), verCiudad(), etc) ya que me parece mas correcto reutilizar código en lugar de repetir el código, y dado el caso de cambiar algo, se tendría que cambiar en un sitio y no en los dos.

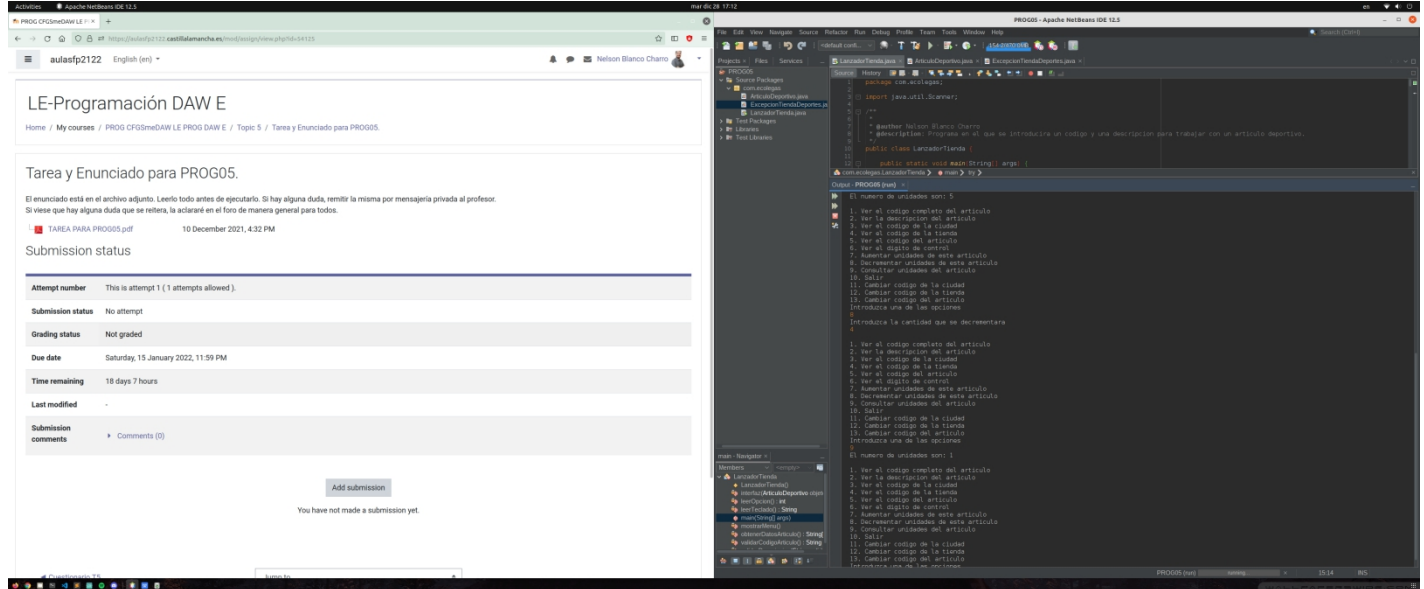
En el método “decrementarUnidades()” de la clase “ArticuloDeportivo.java”, el enunciado dice que establezcamos el valor de las unidades a 0 y mostremos un mensaje alertando cuando se decrementan mas unidades de las que tiene el articulo. Yo creo que es mejor no cambiar el numero de unidades y mostrar una excepción cuando se eliminan mas unidades de las que hay, pero como tengo dudas con respecto a las excepciones no tengo claro si es la mejor opción.

Funcionamiento del programa

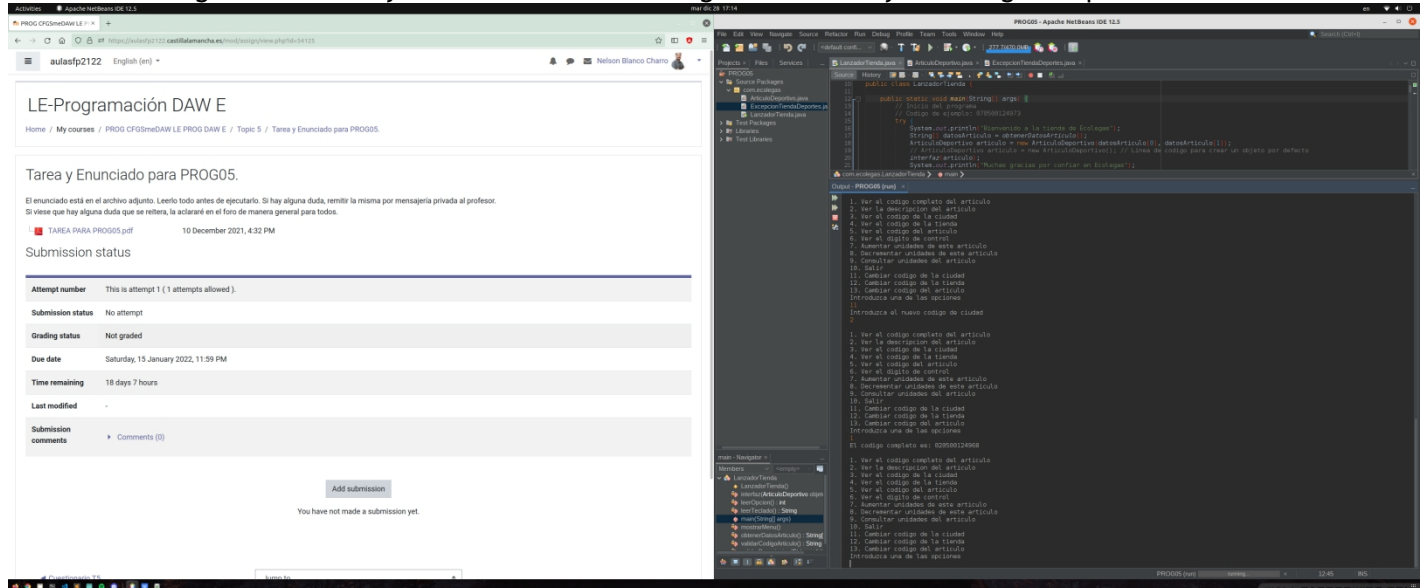
Aumentar el número de unidades del artículo



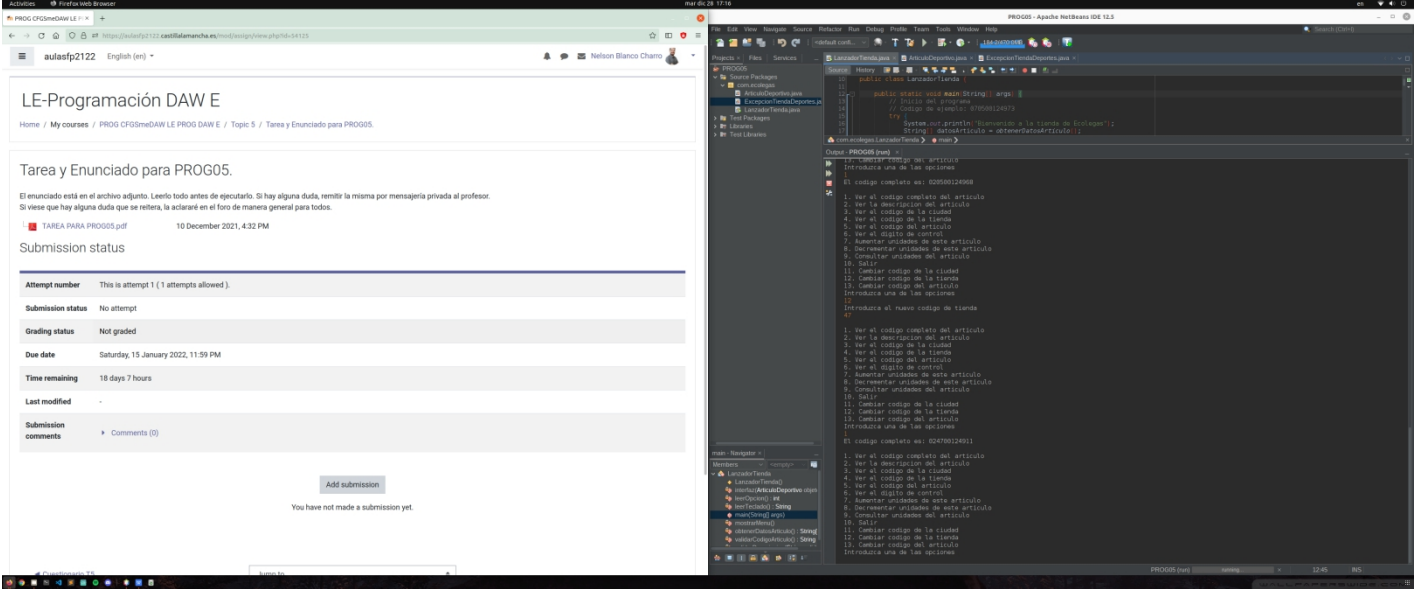
Disminuir el número de unidades del artículo



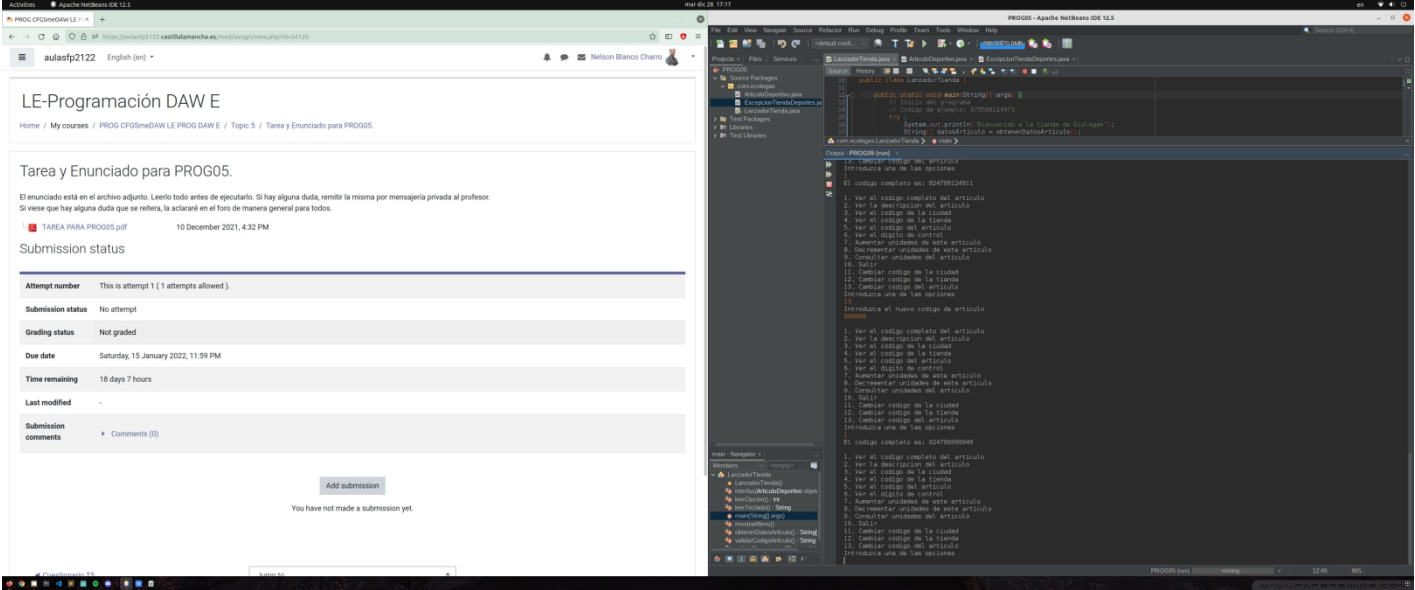
Cambiar el código de la ciudad y mostrar el cambio del dígito de control y del código completo



Cambiar el código de tienda y mostrar el cambio del dígito de control y del código completo



Cambiar el código del artículo y mostrar el cambio del dígito de control y del código completo



Salida del programa y mostrar toda la información del artículo

