**LABORATORIO NRO. 4**

**JEISSON CASALLAS Y CAMILO MURCIA**

**Disfaces**

EN BLUEJ

**PRACTICANDO MDD y BDD con EXCEPCIONES**

[En lab04.doc, CostumeShop.asta y BlueJ costumes]

En este punto vamos a aprender a diseñar, codificar y probar usando excepciones. Para esto se van a trabajar algunos métodos de la clase Costume

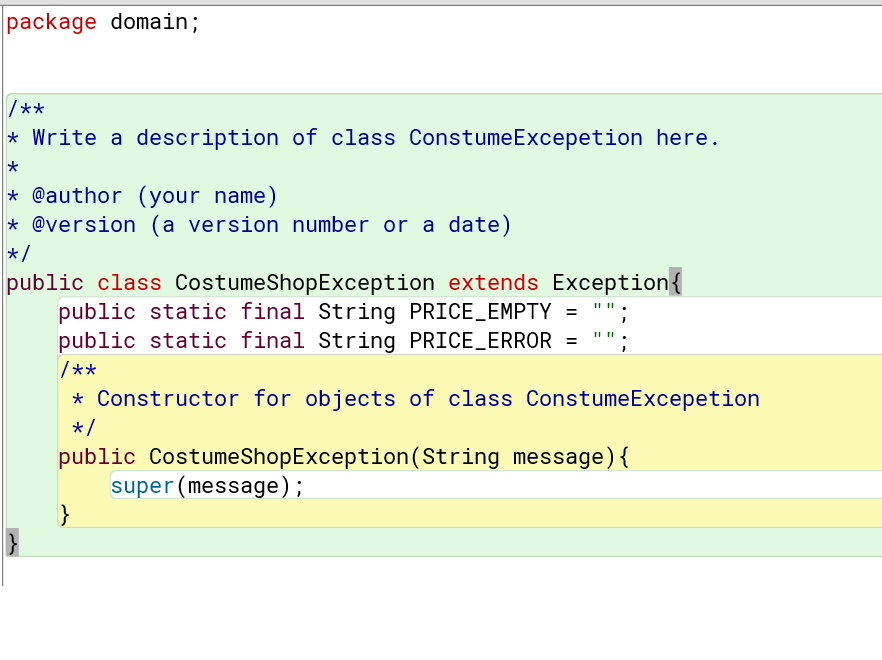
**1. En su directorio descarguen los archivos contenidos en costumes.zip revisen el contenido y estudien el diseño estructural de la aplicación (únicamente la zona en azul).**

**2. Expliquen por qué el proyecto no compila. Realicen las adiciones necesarias para lograrlo.**

El proyecto no compila debido a que no hay aun una clase de excepcion, y para que funcione debemos crear esta clase.

PANTALLAZO

**3. Dado el diseño y las pruebas, documenten y codifiquen el método price().**



**4. Dada la documentación y el diseño, codifiquen y prueben el método price(type).**

**5. Documenten, diseñen, codifiquen y prueben el método price(makeUp).**

**CostumeShop**

EN CONSOLA

El objetivo de esta aplicación es mantener un catálogo de los disfraces ofrecidos por una tienda en Halloween en el proyecto CostumeShop. En este proyecto se ofrecen diferentes tipos de disfraces:básicos y completos.

**Conociendo el proyecto CostumeShop [En lab04.doc]**

**No olviden respetar los directorios bin docs src**

1. En su directorio descarguen los archivos contenidos en costumeShop.zip, revisen el contenido. ¿Cuántos archivos se tienen? ¿Cómo están organizados? ¿Cómo deberían estar organizados?

Se tienen tres archivos, customShop, customShopGUI, Log, no estan organizados, solo se encuentran los .java en la carpeta, deberian estar estar en una carpeta de src.

2. Estudien el diseño del programa: diagramas de paquetes y de clases. ¿cuántos paquetes tenemos? ¿cuántas clases tiene el sistema? ¿cómo están organizadas? ¿cuál es la clase ejecutiva?

Solo tenemos un paquete, y tenemos 4 clases(Costume, Basic, Complete, CustomeShopExeption) y una de pruebas, estan organizadas mediante la documentacion(en la carpeta doc), y domain, en el que tenemos el resto de archivos

3. Prepare los directorios necesarios para ejecutar el proyecto. ¿qué estructura debe tener? ¿qué clases deben tener? ¿dónde están esas clases? ¿qué instrucciones debe dar para ejecutarlo?

4. Ejecute el proyecto, ¿qué funcionalidades ofrece? ¿cuáles funcionan?

5. Revisen el código y la documentación del proyecto. ¿De dónde salen los disfraces iniciales? ¿Qué clase pide que se adicionen? ¿Qué clase los adiciona?