PROGRAMACIÓN ORIENTADA A OBJETOS
Excepciones
2024-1
Laboratorio 4/6

OBJETIVOS

1. Perfeccionar el diseño y código de un proyecto considerando casos especiales y errores.
2. Construir clases de excepción encapsulando mensajes.
3. Manejar excepciones considerando los diferentes tipos.
4. Registrar la información de errores que debe conocer el equipo de desarrollo de una aplicación en producción.
5. Vivenciar las prácticas ***Designing*** - *Simplicity*.

Refactor whenever and wherever possible.

ENTREGA

- ➔ Incluyan en un archivo .zip los archivos correspondientes al laboratorio. El nombre debe ser los dos apellidos de los miembros del equipo ordenados alfabéticamente.
- ➔ Deben publicar el avance al final de la sesión y la versión definitiva en la fecha indicada, en los espacios preparados para tal fin.

Disfaces

EN BLUEJ

PRACTICANDO MDD y BDD con EXCEPCIONES

[En lab04.doc, activities.asta y BlueJ activities]

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

1. En su directorio descarguen los archivos contenidos en `activities.zip` revisen el contenido y estudien el diseño parcial que está en el diagrama de clases. ¿Qué estructura de datos es Actividad? Justifique la respuesta.
2. Expliquen por qué el proyecto no compila. Realicen las adiciones necesarias para lograrlo.
3. Dadas las pruebas, documenten, diseñen y codifiquen el método `time()`.
   Adicionen pruebas análogas a cada una de las dadas para una actividad más profunda (nivel 3)
4. Dada la documentación y el diseño, codifiquen y prueben el método `time(unknown, error, empty)`.
5. Documenten, diseñen, codifiquen y prueben el método `price(estimate)`.
6. Documenten, diseñen, codifiquen y prueben el método `price(activity)`.