

TECNOLOGÍA DE PROGRAMACIÓN 2016/2017

miércoles, 15 de febrero de 2017

Práctica 1: Venta de entradas de cine multisala (versión 1)

Especificaciones

Escribir la primera versión de un programa Java para la venta de entradas en un cine multisala de acuerdo a las siguientes especificaciones:

- El programa se usará por personal del cine en la ventanilla de venta.
- El cine tiene varias salas de exhibición de películas en sesiones con diferentes horarios.
- Cada sala tiene un número, una platea con asientos y sesiones.
- Una sesión tiene una película y un horario de exhibición.
- La configuración (número y disposición de los asientos) de la platea de cada sala estará guardada en uno o más ficheros de texto.
- La información sobre las sesiones estará guardada en uno o más ficheros de texto.
- Se podrá efectuar la venta de entradas numeradas para una sesión entre los asientos libres en ese momento. La entradas se generarán en forma de ficheros de texto, un ejemplo puede ser:

```
Cine Rialto
Sala 2
Master and Commander
Sesión 17:00 - 19:30 - 25/02/2017
Asiento fila 2 número 7
```

- Se podrá conocer el mapa de ocupación de la platea para una sesión, así como el número de asientos totales y libres, un ejemplo puede ser:

```
Cine Rialto
Sala 2
Master and Commander
Sesión 17:00 - 19:30 - 25/02/2017
Asientos totales: 230 libres: 214

- - - - -
- - - X X - X X - - - - - X X - - - - -
- - - - X X X X X X X X X - X X - X X - - - - -
- - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - -
- - - - - - - - - - - - - - - - - - - - - - -
```

Especificaciones de implementación

- El diseño OO deberá evitar dependencias innecesarias entre clases con el fin de tener un diseño reutilizable y extensible. Las clases no podrán tener métodos estáticos.
- En esta versión no existirá ningún tipo de interfaz de usuario. En su lugar se implementará una clase de test con un método `main` donde se crearán objetos y se invocarán métodos para probar adecuadamente las clases diseñadas.
- Se utilizarán vectores para el almacenamiento de datos en memoria.
- Se capturarán y tratarán adecuadamente las posibles excepciones.
- Se sobrescribirán los métodos `equals`, `hashCode` y `toString` de cada clase.
- Se cumplirá lo indicado en el documento de Normas de entrega de prácticas (disponible en Moodle), en particular el seguimiento de *Google Java Style Guide* [1].

Referencias

[1] <https://google.github.io/styleguide/javaguide.html>