

# Evaluación Parcial N°2

Sigla	Nombre Asignatura	Tiempo Asignado	% Ponderación
DSY1102	Desarrollo Orientado a Objetos	120 minutos	40%

## 1. Situación evaluativa

<input checked="" type="checkbox"/>	Ejecución práctica	<input type="checkbox"/>	Entrega de encargo	<input type="checkbox"/>	Presentación
-------------------------------------	--------------------	--------------------------	--------------------	--------------------------	--------------

## 2. Agente evaluativo

<input checked="" type="checkbox"/>	Heteroevaluación	<input type="checkbox"/>	Coevaluación	<input type="checkbox"/>	Autoevaluación
-------------------------------------	------------------	--------------------------	--------------	--------------------------	----------------

### 3. Tabla de Especificaciones

Resultado de Aprendizaje	Indicador de Logro (IL)	Ponderación Indicador Logro
RA2 Aplica conceptos avanzados del paradigma orientado a objetos en componentes de software, para dar respuesta a requerimientos de clientes.	IL 2.1 Aplica las sentencias de ciclos, para ser representadas en la solución según el requerimiento del usuario en un caso de negocios.	15%
	IL 2.2 Aplica conceptos de encapsulación avanzada permitiendo cambios internos sin afectar otros componentes del sistema del software.	15%
	IL 2.3 Configura una colección para almacenar información temporal en el programa según lo solicitado por el usuario.	30%
	IL 2.4 Aplica los conceptos de herencia en la programación orientada a objetos que permitan dar solución a un problema planteado un caso de negocios.	20%
	IL 2.5 Utiliza abstracción y polimorfismo de manera efectiva, para permitir la creación de interfaces flexibles y genéricas que puedan adaptarse a diferentes situaciones.	20%
<b>Total</b>		<b>100%</b>

## 4. Instrucciones generales para el/la estudiante

Esta es una evaluación que corresponde a una ejecución práctica y tiene un 40% de ponderación sobre la nota final de la asignatura. El tiempo para desarrollar esta evaluación es de **90 minutos** y se realiza de manera **individual** en laboratorio PC avanzado.

Esta evaluación consiste en la resolución de un caso donde el estudiante debe aplicar colecciones y herencia. Por otro lado, realizar métodos abstractos, polimorfismos, sobrecarga e implementación de interface, entre otros.

Para dejar registro deberá realizar los siguientes puntos:

- Realizar el diagrama de clases.
- Programar en el lenguaje JAVA utilizando el IDE NetBeans el diagrama de clases generado.
- Deberá dejar registro del trabajo realizado por medio de la plataforma de Blackboard, para ello debe comprimir la carpeta de su proyecto y subirla a la actividad correspondiente.

## 5. Evaluación

**Lea el documento atentamente antes de iniciar la creación de la solución.**

## ENUNCIADO

Una tienda de entretenimiento lanzará un nuevo departamento vintage para la venta de producciones audiovisuales en formato físico, por lo que es requerido desarrollar una aplicación que permita agilizar el registro de estos y los respectivos precios de venta.

Inicialmente, la tienda partirá con 2 tipos de producciones: series y películas.

Los datos por considerar para cada uno de estos tipos son:

- Las series tienen cantidad de capítulos, duración promedio y temporadas.
- Las películas tienen duración y clasificación TE, TE+7, 14 y 18.



Determine la **SUPERCLASE** y sus **SUBCLASES**, que permitan resolver el problema planteado.

Todas las clases que representen entidades deben tener los siguientes métodos implementados: constructores, accesadores y mutadores.

Los requisitos para realizar son los siguientes:

- Implementar una **INTERFACE** IParametros, en la súper clase con 3 porcentajes:

- IVA: 19%
- DESCUENTO\_PELICULA: 5%
- DESCUENTO\_SERIE: 10%
- Definir 2 Métodos:
  - obtenerDescuento, aplicará el descuento según el tipo de producción a partir del precio neto y constantes definidas.
  - totalDeIVA, calculará el IVA por pagar.

- La **SUPERCLASE** debe considerar:

- Implementar la Interface y ser abstracta.
- Atributos: código, nombre, año de estreno, formato disponible (Laser Disc, DVD, VHS, Blue Ray o BetaMax) y precio neto (sin iva).
- Un método abstracto con nombre **imprimirBoleta**, que permita visualizar: el costo neto, descuento (si los hay), el IVA y total a pagar. Este método deberá llamar los métodos de descuento y cálculo de IVA.

• Las **SUBCLASE** deben considerar:

- **No** debe permitir tener subclases.
- atributos mencionados para cada una.
- Aplicar sobrecarga al método **obtenerDescuento**, según el valor que determine el usuario por parámetro o el que está por defecto en la **interface**.
- Debe aplicar sobreescritura al método **imprimirBoleta**.

• Debe existir una clase gestora que contenga una ArrayList para las producciones en stock, el cual, deberá permitir:

- Ingresar una producción, validando que el nombre no se repita. Nombre de método: **agregar**.
- Listar todas las producciones registradas de forma ordenada. Nombre de método: **listar**.
- Contar las producciones según su tipo, donde se mostrará cuantas series y películas existen. Nombre de método: **contar**.
- Eliminar producción según nombre. Nombre de método: **eliminar**.

Desde la clase que contiene el **main** crear un menú que permita:

- Crear dos películas y una serie. (agregar datos por código).
- Agregarlas a la gestora.
- Buscar y Mostrar el total a pagar para una película. Invocando los métodos necesarios para calcular el pago final.
- Listar todas las producciones disponibles.
- Eliminar 1 película mediante el nombre.
- Mostrar las cantidades de cada tipo de producción.

**Usar SOLO los códigos y reglas vistas en clases.  
PLAGIOS serán evaluados con nota mínima.**