



Taller II

Programación Avanzada

Competencias a evaluar

El estudiante deberá ser capaz de construir un programa en Java haciendo uso de los siguientes elementos:

- Modelo de dominio.
- Contratos.
- Diagramas de clases.
- Conceptos de clase, objeto, atributo, método y mensaje.
- Contenedores implementados con arreglos

1. Enunciado del Taller

La biblioteca de la universidad “ABC” está solicitando ayuda de los alumnos de Programación Avanzada, para desarrollar un sistema que facilite la gestión de la unidad, ya que el actual sistema informático no entrega esas facilidades. Con este objetivo, los administradores de la biblioteca exportaron a un archivo de texto todos los datos que han sido almacenados hasta el momento. Los archivos son **clientes.txt**, **libros.txt**, **prestamos.txt**.

A partir de los datos que se han exportado y de los requerimientos que han sido identificados, se conoce que existen clientes (rut, nombre, apellido, número de celular y el correo electrónico de contacto). Además, se han exportado todos los libros que están presente en la biblioteca. De cada libro se conoce, el identificador ISBN, el título, el nombre del autor y el año de publicación, además de estar asociado a una lista de préstamos.

Cada cliente puede pedir un préstamo de algún libro que esté presente en biblioteca, por lo que tendrá una lista de préstamos asociada. Estos préstamos se registraron en el archivo **prestamos.txt**; donde cada préstamo posee un código único, la identificación del cliente que ha pedido el préstamo, el libro pedido, la fecha del préstamo, la fecha de devolución estimada y la fecha de devolución real. Uno de los aspectos que necesita conocer de la biblioteca es la deuda de cada cliente (que pudo haber sido pagada o no), la cual se calcula de la siguiente forma:

La deuda se calcula como el **N° de días * factor de la deuda**, donde el factor de la deuda se calcula como la diferencia en días entre la fecha de devolución real y la fecha de devolución estimada, y si:

1. Si lo ha entregado en 1 a 7 días, entonces el factor de la deuda es \$100.
2. Si lo ha entregado en 8 días a 14 días, entonces el factor de la deuda es \$300.
3. Si lo ha entregado en 15 días o más, entonces el factor de la deuda es de \$700.

No todos los clientes que han pedido préstamos han devuelto los libros, por lo que ese préstamo no registrará ninguna deuda asociada. Para saber si un cliente ha devuelto el libro, se registrará en el archivo **prestamos.txt** la siguiente información:

Registro de libro pedido y devuelto

1, 17679133-0, 123-4, 31-08-2017, 05-09-2017, 08-09-2017 ← Devuelto 3 días después (deuda de \$300)

Registro de libro pedido y no devuelto

1, 17679133-0, 123-4, 31-08-2017, 05-09-2017, 0 ← Aún no ha devuelto el libro. (todavía no registra deuda)



Requisitos funcionales

La biblioteca requiere que el sistema desarrollado en Java cumpla con los siguientes requisitos funcionales:

RF1: Dado el rut de un cliente ingresado por consola, imprimir a) Para cada préstamo que genera una deuda a dicho cliente (si el libro lo devuelve cuando corresponde o aún no lo devuelve, entonces no genera deuda), el código del préstamo, la fecha del préstamo, la fecha de devolución estimada, la fecha de devolución real, la diferencia en días entre la fecha de devolución real y la estimada, y la deuda asociada. b) La deuda total registrada.

RF2: Dado un rango de fechas que se han ingresado por consola (Por ej. “26-09-2017” y “29-09-2017”), imprimir el libro que más ha sido solicitado durante ese rango; el ISBN del libro, el título, el nombre de su autor, el año de publicación y la cantidad de veces que ha sido solicitado.

RF3: Generar el archivo **devoluciones_pendientes.txt** que contenga: el código único del préstamo, la fecha del préstamo, la fecha de devolución estimada, el apellido del cliente, el nombre del cliente, el n° de celular y correo electrónico del cliente, y el título del libro, para todos los préstamos en que aún no se ha devuelto el libro. Debe estar ordenado por fecha de devolución estimada de la más antigua a la más reciente.

RF4: Generar un menú con opciones para cumplir con RF1 - RF3 y una opción para finalizar la ejecución del programa.

2. Consideraciones del Taller

En relación al trabajo a realizar:

- El taller puede ser resuelto por un máximo de dos personas, **ambos del mismo paralelo**.
- Dudas o consultas directamente con el ayudante.
- La copia en el taller supone la reprobación automática del curso.
- Si el taller no compila, la nota máxima a la que puede optar es un 4.0.

En relación al código fuente del taller:

- El código fuente debe estar documentado siguiendo el estándar JavaDoc.
- El código fuente debe estar debidamente indentado utilizando variables y estructuras nemotécnicas.
- El programa realizado debe contener todas las validaciones de los datos de entrada, ya sea desde consola o desde archivos.

En relación a las entregas del taller:

- Las entregas parciales del taller serán realizadas por medio del sistema Educ@ en la sección Trabajos. Deben ir con el siguiente formato: “Apellidos1_Apellidos2.zip”. El archivo final debe contener la carpeta completa del proyecto en .zip o .rar.
- En el taller se debe trabajar con la librería StdUCN.
- Se descuenta 1 punto (10 décimas) por cada hora de atraso.
- En caso de que la plataforma Educ@ presente problemas, las entregas deben ser enviadas al ayudante por correo electrónico, con copia al profesor de la asignatura. No es excusa enviar un taller tarde por problemas del Educ@.



Fechas de entrega

- 1. Entrega 1 (15%) viernes 6 de octubre antes de las 23:50 horas vía Educ@**
 - a. Informe en formato PDF debe contener:
 - i. Modelo de dominio.
 - ii. Contratos.
 - iii. Cualquier consideración adicional pertinente que contribuya a la comprensión de su solución.
 - iv. En caso de estar trabajando en pareja, incluir un detalle de la distribución de tareas.
- 2. Entrega 2 (20%) viernes 20 de octubre antes de las 23:50 horas vía Educ@**
 - a. Informe en formato PDF debe contener:
 - i. Diagrama de clases.
 - ii. Cualquier consideración adicional pertinente que contribuya a la comprensión de su solución.
 - iii. En caso de estar trabajando en pareja, incluir un detalle de la distribución de tareas.
- 3. Entrega 3 (65%) viernes 27 de octubre antes de las 23:50 horas vía Educ@**
 - a. Proyecto completo, debe cumplir con:
 - i. Ser un proyecto completo entregado en Eclipse o Netbeans.
 - ii. Cumplir con las observaciones realizados en la sección 2 de este documento.

Consultas del taller: Rubén Sánchez lic.rubensanchezacosta@gmail.com