



Taller I

Introducción a Java

Competencias a evaluar

El estudiante deberá ser capaz de construir un programa en Java utilizando los conocimientos aprendidos en el curso, para ello el estudiante requerirá tener dominio en lo siguiente:

- Despliegue e ingreso de datos por Consola.
- Lectura y escritura de archivos.
- Uso de menú.
- Arreglos, sean unidimensionales (vectores) o bidimensionales (matrices)
- Ordenamiento, inserción y búsqueda en arreglos.
- Subprogramas, métodos y funciones.

1. Introducción

“Atangana”, es momento de la acción al ras de la lona, el evento más grande de la lucha libre, Wrestlemania™, está a la vuelta de la esquina y para celebrarlo **Vince McMahon** (¿Quién más, eh?) le pide a usted desarrollar un pequeño programa en Java para mantener la base de datos de los luchadores de la WWE, si esto no se desarrolla bien...

You...

are...

FIRE!



“No chance... that's what you've got”

2. Funcionalidad

Al ejecutar el programa, primero se debe leer el archivo “RosterWWE.txt”, el cual contiene la información de todos los luchadores que han pasado por la empresa (*la estructura y el detalle de este archivo se encuentra en la sección Archivos*), para las consultas a la base de datos se debe crear un menú con las siguientes características.



a. Menú Principal

El menú principal debe contener las siguientes opciones:

- **Desplegar Luchadores**
- **Ordenar Base de Datos**
- **WWE WarGame Alfa**
- **Cerrar Programa**

Según la opción que elija el usuario, se realizará una acción diferente.

b. Desplegar Luchadores

Al ingresar a la primera opción entrará a un submenú con las siguientes opciones de despliegue

- **Desplegar todos los luchadores actuales**

Despliega todos los luchadores que se encuentran actualmente en la empresa.

- **Desplegar por marca**

Despliega todos los luchadores según la marca en la que participan (Puede ser RAW, Smackdown o NXT).

Ejemplo, si selecciono RAW y la base de datos esta ordenada alfabéticamente A-Z, los 3 primeros resultados serían:

Adam Rose – Jobber – POW 106 TGH 101 SPD 104 CHA 103 – Especialidad: POW
Big E – Midcarter – POW 108 TGH 107 SPD 101 CHA 104 – Especialidad: POW
Big Show – Midcarter – POW 110 TGH 108 SPD 103 CHA 105 – Especialidad: POW

- **Desplegar según su estatus**

Despliega todos los luchadores según el estatus que tienen en la empresa (Leyenda, Superestrella, Midcarter, Jobber)

Ejemplo, si selecciono Leyenda y la base de datos esta ordenada alfabéticamente A-Z, los 3 primeros resultados serían:

Andre The Giant – POW 112 TGH 111 SPD 105 CHA 108 – Especialidad: POW
Batista – POW 111 TGH 110 SPD 107 CHA 104 – Especialidad: POW
Big Boss Man – POW 111 TGH 113 SPD 106 CHA 108 – Especialidad: TGH

- **Salir**

c. Ordenar Base de Datos

Al ingresar a la segunda opción entrará a un submenú con las siguientes opciones

- Ordenar luchadores alfabéticamente por Nombre (A-Z)
- Ordenar luchadores según “stat” (De menos a mas)



Cuando se selecciona la opción de ordenar según “stat”, estos pueden ser: **POW, TGH, SPD y CHA** (Poder, Dureza, Velocidad y Carisma respectivamente).

d. WWE WarGame Alfa

Esta opción contiene una pequeña versión alfa del juego que **WWE Games** quiere desarrollar, al inicio elegirá 3 luchadores distintos (ingresando su nombre), al terminar, se desplegarán en consola junto con sus “stats”, además, se desplegará el tipo de pelea que se va a realizar (Batalla de Poder, de Dureza, de Velocidad o de Carisma, **cualquiera de estas será seleccionada al azar**), al seleccionar un luchador, el juego seleccionará otro luchador (al azar) al cual se enfrentará, quien tenga el “stat” más alto gana la pelea.

Este juego tiene ciertas reglas:

- Se enfrentarán luchadores del mismo género solamente (luchador con luchador, diva con diva).
- Si la batalla es de la especialidad del luchador (Ej.: *Un luchador es especialista en poder y se enfrenta en una batalla de poder*), se activa su habilidad especial lo que le hace ganar temporalmente 15 puntos en ese stat.
- Al terminar la batalla se debe almacenar el detalle de esta lucha, esto debe ser: El nombre del luchador que seleccionaste, el nombre del luchador oponente, el tipo de lucha y el resultado (Victoria, Empate o Derrota)

e. Cerrar programa

Al seleccionar esta opción, se despliega un mensaje indicándole al usuario que se ha finalizado el programa, además se deben almacenar todas las luchas realizadas durante la ejecución en un archivo llamado “EstadisticaWarGames.txt”. El almacenamiento de este se detalla en el apartado de archivos.

NOTA: Cada vez que el usuario elija la opción Salir dentro de un submenú este debe retornar al usuario al menú principal, si un despliegue no es exitoso se debe desplegar un mensaje de “No se encuentran datos para la opción seleccionada”, y si la opción que el usuario digita no es válida debe desplegarse un mensaje de error al usuario.

3. Archivos

- **Entrada**

El archivo de entrada contiene el siguiente formato:

Nombre, Genero (Luchador o Diva), Estatus, Marca, POW, TGH, SPD, CHA, Especialidad

Ejemplo:

Brock Lesnar,Luchador,Superestrella,RAW,113,110,108,105,POW
John Cena,Luchador,Leyenda,Smackdown,111,108,106,113,CHA
Paige,Diva,Midcarter,RAW,108,110,105,107,TGH

Donde los campos que acompañan al luchador son

- **POW** – Poder
- **TGH** – Dureza
- **SPD** – Velocidad
- **CHA** – Carisma



- **Salida**

Como fue antes mencionado, el archivo de salida es uno solo, el cual sus registros deben contener todo el detalle de las luchas realizadas según el siguiente formato:

Nombre del Luchador Seleccionado, Nombre del Oponente, Tipo de Lucha, Resultado.

Ejemplo:

Brock Lesnar, John Cena, CHA, Derrota
John Cena, Randy Orton, POW, Victoria
Emma, Paige, SPD, Derrota

4. Observaciones

Sobre el trabajo:

- La resolución del taller puede ser realizado, como máximo, en grupos de dos personas.
- Cualquier duda, consulta y/o inconveniente enviar un correo al mail wth001@alumnos.ucn.cl

Sobre el código (cada punto de este ítem se evaluará):

- Su código debe estar **documentado** con el estándar **JavaDoc**.
- Su código debe estar **identado** y con **variables nemotécnicas**.
- Su código **debe tener las validaciones que se requieran**, tanto para la lectura de los datos desde el archivo, de los parámetros de entrada, como de los posibles cálculos que se pueden requerir.

Sobre la entrega del trabajo:

- La entrega del taller será realizada por medio del sistema Educ@ en la sección trabajos, el cual debe ir con el siguiente formato "Apellidos1_Apellidos2.zip", este archivo debe contener la carpeta completa del proyecto en .zip o .rar, incluyendo los archivos proporcionados al inicio del taller.
- En el taller se utilizará la librería **StdUCN**.
- Se descuenta 0.002 puntos de la nota final por cada minuto de atraso a partir de la fecha de entrega.

Tener presente que:

- La copia de talleres será sancionada con nota 1.0 y **reprobación automática del curso**.
- Si su entrega tiene errores de compilación, la nota será un 2.0.

5. Fechas de Entrega

- **Entrega N°1 – 20% – Viernes 24 de Marzo hasta las 23:59 horas por Educ@.**
 - Informe en PDF, este debe contener:
 - Estructura de Datos utilizada y su Justificación
 - Estructura del Programa
 - Supuestos del Problema
- **Entrega N°2 – 80% – Martes 4 de Abril hasta las 23:59 horas por Educ@.**
 - Código Java, este debe:
 - Ser entregado como proyecto de Eclipse o Netbeans
 - Cumplir las observaciones descritas en el punto 4.