

## Ejercicio sobre polimorfismo

---

### Introducción

En su nuevo puesto de trabajo como analista de seguridad se le ha encargado desarrollar un sistema de autenticación. Este sistema controla acceso a un conjunto de datos mediante un esquema de reto-respuesta. Los potenciales usuarios serán Empleados, Gerentes y Proveedores. Todos los tipos de usuario son Retables por el sistema, aunque el reto debe ser diferente por tipo. Todos los tipos de usuario también tienen un nombre y una ID. Los Empleados saben el nombre de su jefe inmediato, los Gerentes no tienen jefe inmediato. Los Empleados y los Gerentes tienen un departamento. Los Proveedores conocen el nombre de quien que los atiende.

Los retos siempre son una pregunta emitida por el sistema y una subsecuente comparación de la respuesta del usuario contra una respuesta esperada. Algunos ejemplos de retos en la industria:

- Requisito de contraseña.
- Completar o alterar una secuencia de números.
- Pregunta secreta.
- Generación de *tokens*.

A los usuarios Retables también se les puede fichar o marcar. Cuando un usuario falla su reto tres veces, es fichado o marcado. Los Empleados, al ser fichados, despliegan su nombre, su ID y su jefe inmediato. Los Gerentes despliegan su nombre, ID y departamento. Los Proveedores despliegan su nombre, ID y persona que les atiende.

Una vez autenticados, el sistema identifica a los usuarios en dos categorías: *Normies* y Administradores. Los *Normies* y los Administradores pueden consultar los datos, pero sólo los Administradores pueden modificarlos. Los Gerentes y los Empleados son Administradores; los Proveedores son *Normies*.

Los Empleados y Gerentes pueden leer toda la información, pero los Proveedores sólo pueden leer la información almacenada en registros impares. Los Gerentes pueden modificar toda la información, pero los Empleados sólo pueden modificar los registros pares.

### Instrucciones

Represente el conjunto de datos con un `ArrayList` de *strings*. Desarrolle un diagrama de clases que represente adecuadamente los tipos de usuario y sus diferentes atribuciones ante el sistema. Incluya en su diagrama las clases y relaciones necesarias para incluir un simulador que permita al usuario seleccionar un tipo... eh... de usuario, y luego autenticarse y usar el sistema.

A los Gerentes se les debe presentar un menú que permita:

1. Leer cualquier posición de los datos.
2. Modificar cualquier posición de los datos.
3. Agregar un nuevo dato.
4. Consultar la cantidad de datos.

A los Empleados se les presenta un menú similar, pero mostrando un mensaje de error al usuario si intenta modificar una posición inválida. Además, el Empleado no puede agregar datos. A los Proveedores simplemente se les pide una posición a consultar, restringida a las posiciones permitidas para Proveedores.

Finalmente, implemente su sistema y su simulador en Java.

Debe entregar en Canvas:

- **[50 pts.]** Diagrama de clases.
  - **[10 pts.]** Sintaxis.
  - **[20 pts.]** Correcto uso de interfaces.
  - **[20 pts.]** Representación correcta de todos los componentes del sistema y el simulador.
- **[50 pts.]** Programa que implemente su diseño.
  - **[20 pts.]** Correcto uso de polimorfismo.
  - **[20 pts.]** Cumplimiento de requisitos funcionales.
  - **[10 pts.]** Comentarios y encabezados.