

Tarea Final de la Unidad PROG05

Realizar un ejercicio en JAVA sobre el desarrollo de clases.

Entrega: Deberás crear una carpeta comprimida en formato 'zip' con el nombre "Apellido_Nombre_Tarea_Final_PROG05". Dentro de esta carpeta, incluye el proyecto o el paquete donde están las clases del ejercicio.

Finalmente, sube el archivo comprimido en la tarea correspondiente del aula virtual.

Enunciado del Problema

Título: GESTIÓN DE DISPOSITIVOS MÓVILES

Descripción: Desarrollar un programa en Java que gestione dos dispositivos móviles, simulando la funcionalidad básica de los mismos. Uno de los dispositivos contará inicialmente con una tarjeta SIM, y el otro no. El programa permitirá realizar llamadas, gestionar tarjetas SIM, y mostrar información detallada de los móviles, utilizando métodos estáticos para mejorar la estructura y funcionalidad del código.

Clase Movil

1. Atributos:

- id: Identificador único para cada móvil. Comenzará en 0 y se incrementará en una unidad para cada móvil creado.
- marca, modelo, color: Información básica del móvil.
- contadorLlamadas: Número total de llamadas realizadas desde el móvil.
- tarjeta: Objeto de la clase Tarjeta, que representa la SIM insertada en el móvil. Se inicializa como null.

2. Constructores:

- Constructor vacío: Inicializa los atributos básicos con valores predeterminados, asigna el identificador automáticamente y el atributo tarjeta a null.

- Constructor parametrizado: Recibe valores para los atributos marca, modelo, y color, asigna el identificador automáticamente y deja la tarjeta como null.

3. Métodos:

- llamar(String nombre, String numero): Realiza una llamada al número especificado, siempre que:
 - El móvil tenga una tarjeta insertada.
 - El número comience con "6" o "9". Si alguna de las condiciones no se cumple, muestra un mensaje de error. En caso de éxito, incrementa el contador de llamadas y muestra un mensaje de confirmación.
- comprobarTarjeta(): Devuelve un mensaje indicando si el móvil tiene tarjeta o no.
- @Override public String toString(): Devuelve una cadena con toda la información del móvil, incluyendo si tiene tarjeta y el contador de llamadas.
- **Método estático** getTotalMoviles(): Devuelve el número total de móviles creados.

Clase Tarjeta

1. Atributos:

- compania: Nombre de la compañía telefónica.
- numero: Número de teléfono asociado a la tarjeta.

2. Constructores:

- Constructor vacío: Inicializa los atributos con valores predeterminados.

- Constructor parametrizado: Inicializa los atributos con valores proporcionados.

3. Métodos:

- **@Override** public String toString(): Devuelve una cadena con toda la información de la tarjeta.
- **Método estático** validarNumero(String numero): Valida si el número de teléfono es válido (debe comenzar con 6-9 y tener 9 dígitos).

Programa Principal

1. **Crear los Móviles:** Al iniciar el programa, se crean dos objetos de la clase Movil. Uno de ellos dispondrá de una tarjeta SIM configurada, mientras que el otro no tendrá tarjeta.
2. **Menú Interactivo:** Incluye las siguientes opciones:
 - **Poner Tarjeta:** Permite añadir una tarjeta SIM a un móvil que no tenga tarjeta. Si el móvil tiene tarjeta, se mostrará un mensaje indicando que no es posible añadir más tarjetas. El usuario deberá proporcionar el nombre de la compañía telefónica y el número de teléfono. Antes de asignar la tarjeta, se validará el número utilizando el método estático Tarjeta.validarNumero.
 - **Quitar Tarjeta:** Permite quitar la tarjeta SIM de un móvil que la tenga. Si el móvil no tiene tarjeta, se mostrará un mensaje de error.
 - **Llamar:** El usuario deberá proporcionar:
 - El nombre del destinatario.
 - El número de teléfono al que desea llamar.

En las tres opciones anteriores, hay que solicitar al usuario que indique con qué móvil quiere realizar la acción, validar si es necesario.

- **Mostrar Información:** Muestra toda la información de ambos móviles, incluyendo:
 - Atributos básicos.
 - Si tienen tarjeta o no.
 - Contador de llamadas.
 - El número total de móviles creados, utilizando el método estático `Movil.getTotalMoviles`.
- **Salir:** Finaliza el programa.

Ayuda:

- Se proporciona el método para validar el número utilizando una expresión regular, devolviendo cierto o falso tras la comprobación:

```
public static boolean validarNumero(String numero) {  
  
    return numero.matches("[6-9]\\d{8}");  
  
}
```
- Para comprobar si un número de teléfono comienza por un determinado carácter, se puede utilizar `startsWith` o `charAt`:

```
numero.startsWith("6") o numero.charAt(0) == '6'
```

Rúbrica de Evaluación

Criterio	Puntos
Clase Movil	
- Implementación correcta de atributos.	0.5
- Implementación de constructores.	0.5
- Métodos llamar, comprobarTarjeta, toString.	1.0
- Método estático getTotalMoviles().	0.5
Clase Tarjeta	
- Implementación de atributos y constructores.	0.5
- Método toString.	0.5
- Método estático validarNumero().	0.5
Programa Principal	
- Creación de los dos móviles.	1.0
- Menú Interactivo	1.0
- Función Poner Tarjeta.	1.0
- Función Quitar Tarjeta.	1.0
- Función Llamar.	1.0
- Función Mostrar Información.	1.0
Total	10 puntos