05 - EJERCICIO DE PROGRAMACIÓN JAVA

Parking.

Práctica con Excepciones

Control de Ocupación de un Parking

Se pide realizar una aplicación sencilla que permita controlar la ocupación en tiempo real de un Parking.

Para ello, se necesitará una clase Parking con las características que se indican a continuación.

Todo el control de errores se hará lanzando y capturando Excepciones.

Clase Parking

Atributo matrículas

La clase Parking tendrá como atributo un ArrayList de String llamado **matrículas**.

Este ArrayList representará las distintas plazas del parking, y contendrá las matrículas de los coches aparcados en cada plaza.

Por ejemplo, si el coche con matrícula 1234-KW aparca en la plaza 23, entonces introduciremos en el ArrayList la cadena “1234-KW” en la posición 23 del array.

Si la plaza 45 estuviera libre, entonces la posición 45 del array contendrá el valor null.

Atributo nombre

Este otro atributo será un String y se corresponderá con el nombre del parking.

Constructor

El constructor de la clase Parking recibirá el nombre del parking y un entero, indicando el número de plazas del parking, y construirá el ArrayList matrículas con ese tamaño e inicializando cada posición a null.

Métodos

public String getNombre()

Devuelve el nombre del parking

public void entrada(String matricula, int plaza)

Este método introduce el coche con la matricula indicada en la plaza del parking indicada.

Control de errores:

- Si la matrícula tiene un tamaño menor de 4 caracteres o es null, se lanzará una excepción con el mensaje “Matrícula incorrecta”
- Si la plaza ya estuviera ocupada se lanzará una excepción con el mensaje “Plaza ocupada”
- Si la matrícula existiera dentro del parking se lanzará una excepción con el mensaje “Matrícula repetida”

public int salida(String matricula)

Este método hace que el coche con la matrícula indicada salga del parking, liberando la plaza correspondiente. Devolverá el número de la plaza liberada.

Control de errores:

- Si la matrícula no existe dentro del parking se lanzará una excepción con el mensaje “Matrícula no existente”

public int getPlazasTotales()

Este método devuelve las plazas totales del parking.

public int getPlazasOcupadas()

Este método devuelve las plazas ocupadas del parking.

public int getPlazasLibres()

Este método devuelve las plazas libres del parking.

public String toString()

La sobrecarga de este método devolverá una cadena que contendrá el nombre del parking y un listado de cada plaza y la matrícula que la ocupa. Por ejemplo:

```
Parking Avenida
----------------
Plaza 0: (vacía)
Plaza 1: 3322AB
Plaza 2: 5342HW
Plaza 3: (vacía)
…
```

Clase principal

En la clase principal realice un programa de prueba que haga lo siguiente:

Crear un objeto Parking:

- Crea un objeto Parking llamado “Parking Centro” con 10 plazas.

Menú de opciones:

- Después de crear el objeto Parking, se mostrará al usuario un menú con cuatro opciones: 1) Entrada de coche y 2) Salida de coche, 3) Mostrar parking y 4) Salir del programa.

- Se le pedirá al usuario que introduzca un valor entero, entre 1 y 4, correspondiente a una de las opciones (use la clase Scanner y el método nextLine, convirtiendo la entrada a entero)

Opción 1. Entrada de coche:

- La opción Entrada de coche le pedirá al usuario que introduzca por teclado la matrícula del coche que entra y la plaza donde se colocará, y a continuación se introducirá el coche en el parking en dicha plaza. La opción mostrará luego el número de plazas totales, ocupadas y disponibles después de la entrada.

- Para pedir la plaza, use el método nextLine de la clase Scanner y convierta el dato introducido por teclado a entero.

- Se capturarán las excepciones si las hubiera, y en ese caso se mostraría el mensaje de error y la entrada no se llevaría a cabo.

Opción 2. Salida de coche.

- La opción Salida de coche le pedirá al usuario que introduzca por teclado la matrícula del coche que sale y a continuación se mostrarán las plazas totales, libres y ocupadas del parking después de la salida.

- Se capturarán las excepciones si las hubiera, y en ese caso se mostraría el mensaje de error y la entrada no se llevaría a cabo.

Opción 3. Mostrar parking.

- Esta opción muestra todas las plazas y las matrículas que las ocupan (se usará el método toString de la clase Parking)

Opción 4. Salir del programa

- El programa mostrará el menú repetidas veces hasta que se pulse esta opción.

Mejoras Finales

1. La entrada de la opción numérica del menú puede causar una caída del programa si el usuario introduce un texto en vez de un entero. Evite esto usando un try…catch en la línea donde se pide la opción del menú elegida.

2. Es posible crear clases de Excepciones propias. Cree una clase ParkingException que construya una excepción con un mensaje y una matrícula, y contenga un método getMensaje que devuelva el mensaje y otro método getMatricula que devuelva la matrícula del coche que ha producido el error.

   Nota: esta clase heredará de la clase Exception

   Una vez realizada esta clase, úsela para lanzar las excepciones de la clase Parking.

   Modifique también el programa principal de forma que los try… catch de la entrada de coches y la salida sean capaz de capturar excepciones de tipo ParkingException y de otros tipos.