

## EJERCICIO GUIADO. JAVA: VARIABLES GLOBALES

### Variables Globales / Propiedades de la Clase

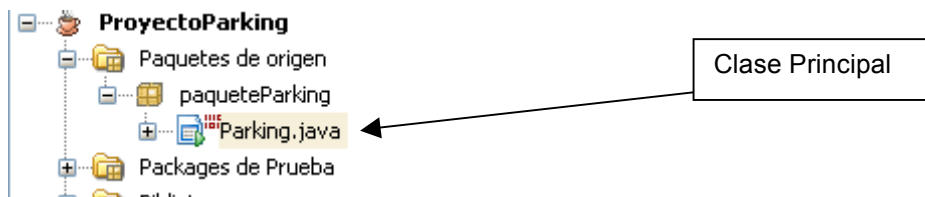
Las propiedades de la clase en java es el equivalente a las variables globales en lenguajes estructurados como el C.

Una propiedad es una variable que puede ser accedida desde cualquier evento programado. Esta variable se inicializa a un valor cuando se ejecuta el programa y los distintos eventos pueden ir cambiando su valor según se necesite.

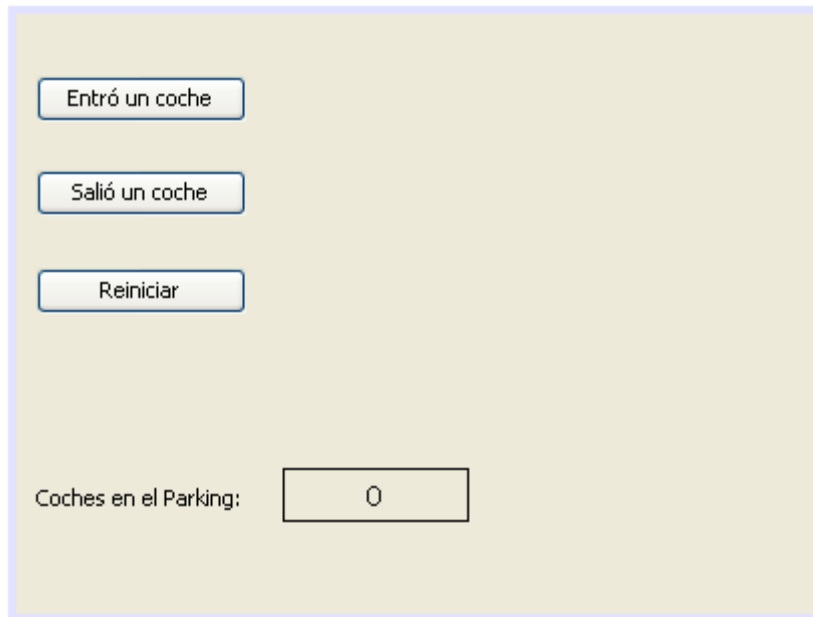
Veamos un ejemplo para entender el funcionamiento de las propiedades de la clase / variables globales.

### Ejercicio guiado 1

1. Crea un nuevo proyecto llamado *ProyectoParking*. Dentro de él añade un paquete llamado *paqueteParking*. Y finalmente añade un JFrame llamado *Parking*. El aspecto de tu proyecto será el siguiente:



2. Tu clase principal es la clase *Parking*.
3. Se pretende hacer un pequeño programa que controle los coches que van entrando y van saliendo de un parking. En todo momento el programa debe decir cuantos coches hay dentro del parking. Para ello debes crear una ventana como la que sigue:



4. Esta ventana contiene lo siguiente:

- Un botón “Entró un coche” llamado btnEntro.
- Un botón “Salió un coche” llamado btnSalio.
- Un botón “Reiniciar” llamado btnReiniciar.
- Una etiqueta con el texto “Coches en el Parking”.
- Una etiqueta con borde llamada etiCoches que contenga un “0”.

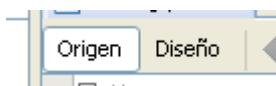
5. Se pretende que el programa funcione de la siguiente forma:

- a. Cuando el usuario pulse el botón “Entró un coche”, el número de coches del parking aumenta en 1, mostrándose en la etiqueta etiCoches.
- b. Si el usuario pulsa el botón “Salió un coche”, el número de coches del parking disminuye en 1, mostrándose en la etiqueta etiCoches.
- c. El botón Reiniciar coloca el número de coches del parking a 0.

6. Para poder controlar el número de coches en el Parking, es necesario que nuestra clase principal *Parking* tenga una propiedad (variable global) a la que llamaremos *coches*. Esta variable será entera (int).

Esta variable contendrá en todo momento los coches que hay actualmente en el *Parking*. Esta variable podrá ser usada desde cualquier evento.

7. Para crear una variable global haz clic en el botón “Origen” para acceder al código:



8. Luego busca, al comienzo del código una línea que comenzará por

```
public class
```

Seguida del nombre de tu clase principal “*Parking*”.

Debajo de dicha línea es donde se programarán las propiedades de la clase (las variables globales)

```

/**
 *
 * @author didact
 */
public class Parking extends javax.swing.JFrame {

    /**
     * Creates new form Parking
     */
    public Parking() {
        initComponents();
    }

```

Aquí se declaran las variables globales, también llamadas propiedades de la clase.

9. En dicho lugar declararás la variable *coches* de tipo *int*:

```

public class Parking extends javax.swing.JFrame {

    int coches;

    /**
     * Creates new form Parking
     */
    public Parking() {

```

Declaración de una variable global *int* llamada *coches*.

10. Cuando el programa arranque, será necesario que la variable global *coches* tenga un valor inicial. O dicho de otra forma, será necesario inicializar la variable. Para inicializar la variable iremos al constructor. Añade lo siguiente al código:

```

public class Parking extends javax.swing.JFrame {

    int coches;

    /**
     * Creates new form Parking
     */
    public Parking() {
        initComponents();
        coches=0;
    }

    /** This method is called from within the construct

```

Inicialización de la propiedad *coches*.

Inicializamos a cero ya que se supone que cuando arranca el programa no hay ningún coche dentro del parking.

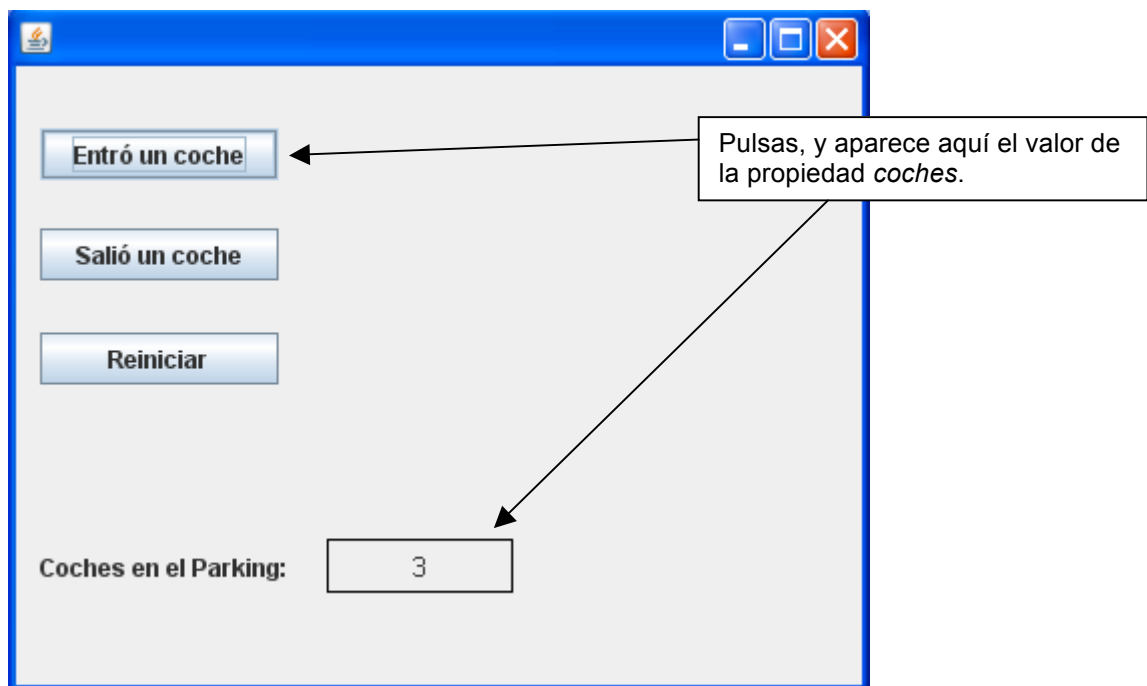
11. Ahora que ya tenemos declarada e inicializada la variable global *coches*, esta puede ser usada sin problemas desde cualquier evento que programemos.

Por ejemplo, empezaremos programando la pulsación del botón “Entró un coche”. Acceda a su evento *actionPerformed* y programe esto:

```
coches=coches+1;  
etiCoches.setText(""+coches);
```

Como ves, se le añade a la variable *coches* uno más y luego se coloca su valor actual en la etiqueta.

12. Ejecuta el programa y prueba este botón varias veces.



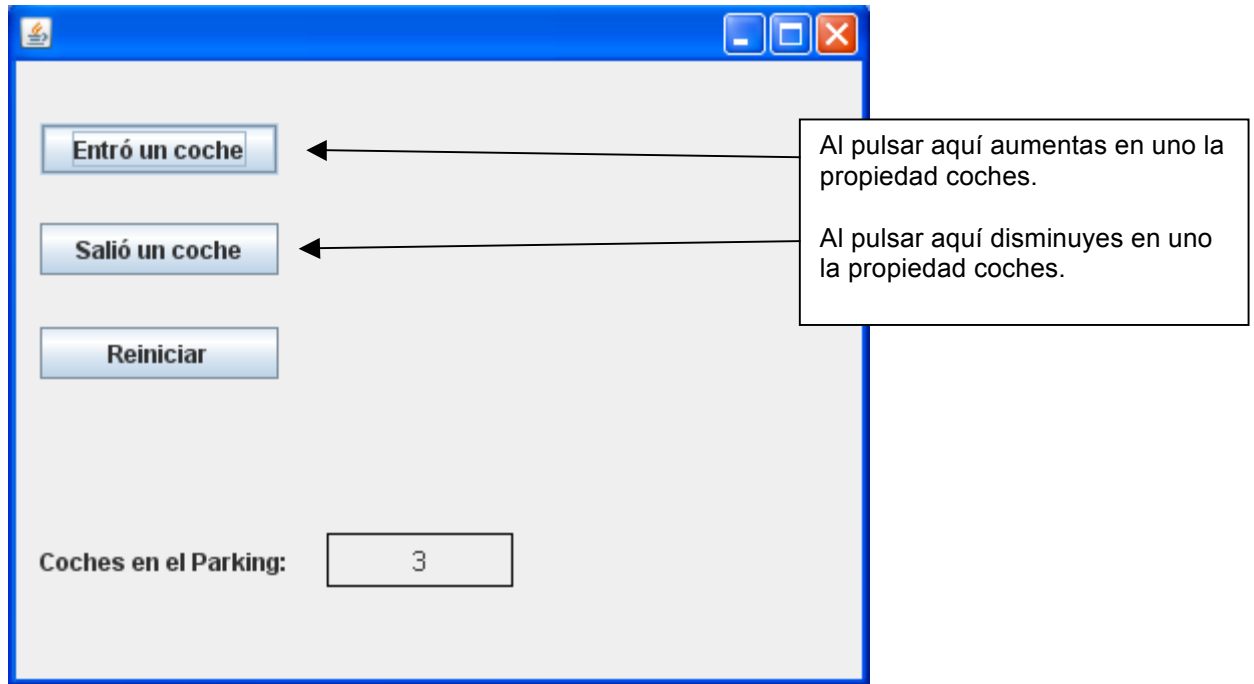
13. Ahora programaremos el botón “Salió un coche” de la siguiente forma:

```
if (coches>0) {  
    coches=coches-1;  
    etiCoches.setText(""+coches);  
}
```

Como ves, se accede igualmente a la propiedad *coches* pero esta vez para restarle una unidad. Luego se muestra el valor actual de *coches* en la etiqueta correspondiente.

Se usa un if para controlar que no pueda restarse un coche cuando el parking esté vacío. (Si hay cero coches en el parking no se puede restar uno)

14. Ejecuta el programa y prueba los dos botones. Observa como la cantidad de coches del parking aumenta o disminuye.



Lo realmente interesante de esto es que desde dos eventos distintos (desde dos botones) se puede usar la variable *coches*. Esto es así gracias a que ha sido creada como variable global, o dicho de otro modo, ha sido creada como propiedad de la clase *Parking*.

15. Finalmente se programará el botón *Reiniciar*. Este botón, al ser pulsado, debe colocar la propiedad *coches* a cero. Programa dentro de su *actionPerformed* lo siguiente:

```
coches=0;  
etiCoches.setText("0");
```

Simplemente introduzco el valor cero en la variable global y actualizo la etiqueta.

16. Ejecuta el programa y comprueba el funcionamiento de este botón.

## CONCLUSIÓN

**Las variables globales, también llamadas propiedades de la clase, son variables que pueden ser usadas desde cualquier evento del programa. Estas variables mantienen su valor hasta que otro evento lo modifique.**

**Las variables globales se declaran justo después de la línea *public class*.**

**La inicialización de estas variables se realiza en el constructor.**