

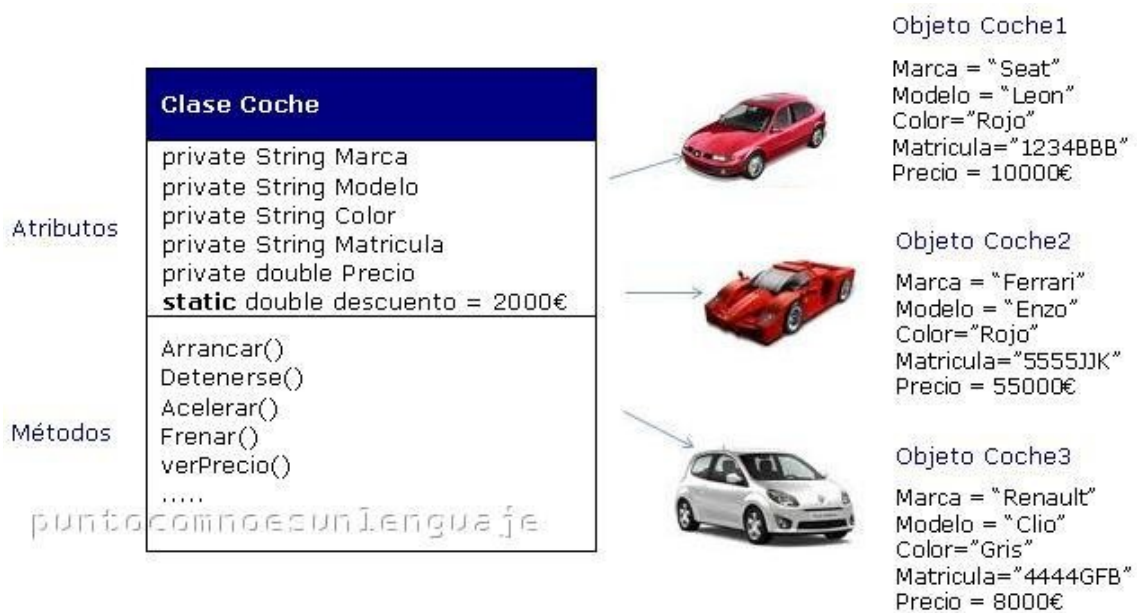
REPASO DE ATRIBUTOS Y MÉTODOS DE CLASE

Los atributos y métodos estáticos también se llaman **atributos de clase** y **métodos de clase**.

Se declaran como **static**.

Supongamos una clase Coche sencilla que se utiliza en un programa de compra-venta de coches usados y de la que se crean 3 objetos de tipo Coche:

La clase contiene 6 atributos: marca, modelo, color, matrícula, precio y descuento. Supongamos que el descuento es una cantidad que se aplica a todos los coches sobre el precio de venta. Como este dato es el mismo para todos los coches y es un valor que se puede modificar en cualquier momento no debe formar parte de cada coche sino que es un dato que deben compartir todos. Esto se consigue declarándolo como **static**.



Un Atributo static:

- No es específico de cada objeto. Solo hay una copia del mismo y su valor es compartido por todos los objetos de la clase.
- Podemos considerarlo como una *variable global* a la que tienen acceso todos los objetos de la clase.
- Existe y puede utilizarse aunque no existan objetos de la clase.

Para acceder a un atributo de clase se escribe:

NombreClase.atributo

Un Método static:

- Tiene acceso solo a los atributos estáticos de la clase.
- No es necesario instanciar un objeto para poder utilizarlo.

Para acceder a un método de clase se escribe:

NombreClase.método()

Ejemplo:

Vamos a escribir una clase Persona que contendrá un atributo contadorPersonas que indique cuantos objetos de la clase se han creado.

contadorPersonas debe ser un atributo de clase ya que no es un valor que se deba guardar en cada objeto persona que se crea, por lo tanto se debe declarar static:

```
static int contadorPersonas;
```

Un ejemplo de uso desde fuera de la clase Persona:

```
System.out.println(Persona.contadorPersonas);
```

Si lo declaramos como private:

```
private static int contadorPersonas;
```

solo podremos acceder al atributo desde fuera de la clase a través de métodos static:

```
public static void incrementarContador(){  
    contadorPersonas++;  
}
```

```
public static int getContadorPersonas() {  
    return contadorPersonas;  
}
```

En este caso un ejemplo de uso puede ser:

```
System.out.println(Persona.getContadorPersonas());
```

Cada vez que se crea una persona se incrementará su valor.

Si no es private, desde otra clase podemos hacerlo así:

```
Persona.contadorPersonas++;
```

Si es private, desde otra clase debemos incrementarlo así:

```
Persona.incrementarContador();
```

//Clase Persona

```
public class Persona {
```

```
    private String nombre;
```

```
    private int edad;
```

```
    private static int contadorPersonas;
```

```
    public Persona() {  
    }
```

```
        public Persona(String nombre, int edad) {  
            this.nombre = nombre;  
            this.edad = edad;  
        }
```

```
        public void setNombre(String nom) {  
            nombre = nom;  
        }
```

```
        public String getNombre() {  
            return nombre;  
        }
```

```
        public void setEdad(int ed) {  
            edad = ed;  
        }
```

```
        public int getEdad() {  
            return edad;  
        }
```

```
        public static int getContadorPersonas() {  
            return contadorPersonas;  
        }
```

```
        public static void incrementarContador() {  
            contadorPersonas++;  
        }
```

```
    }
```

//Clase Principal

```
public class Estatico1 {  
  
    public static void main(String[] args) {  
        Persona p1 = new Persona("Tomás Navarra", 22);  
        Persona.incrementarContador();  
        Persona p3 = new Persona("Jonás Estacio", 23);  
        Persona.incrementarContador();  
        System.out.println("Se han creado: " +  
Persona.getContadorPersonas() + " personas");  
    }  
}
```

En lugar de utilizar el método incrementarContador() cada vez que se crea un objeto, podemos hacer el incremento de la variable estática directamente en el constructor.

El código de la clase Persona y de la clase principal quedaría ahora así:

//Clase Persona

```
public class Persona {  
  
    private String nombre;  
    private int edad;  
    private static int contadorPersonas;  
  
    public Persona() {  
        contadorPersonas++;  
    }  
  
    public Persona(String nombre, int edad) {  
        this.nombre = nombre;  
        this.edad = edad;  
        contadorPersonas++;  
    }  
  
    public void setNombre(String nom) {  
        nombre = nom;  
    }  
  
    public String getNombre() {  
        return nombre;  
    }  
    public void setEdad(int ed) {  
        edad = ed;  
    }  
    public int getEdad() {  
        return edad;  
    }  
}
```

```

    }

    public static int getContadorPersonas() {
        return contadorPersonas;
    }
}

```

//Clase Principal

```

public class Estatico1 {

    public static void main(String[] args) {
        Persona p1 = new Persona("Tomás Navarra", 22);
        Persona p3 = new Persona("Jonás Estacio", 23);
        System.out.println("Se han creado: " +
Persona.getContadorPersonas() + " personas");
    }
}

```