



APRENDERAPROGRAMAR.COM

TIPOS ENUMERADOS  
(ENUM) EN JAVA. EJEMPLO  
DE CÓDIGO CON ENUM.  
ERROR ENUM TYPES MUST  
NOT BE LOCAL. (CU00681B)

Sección: Cursos

Categoría: Curso “Aprender programación Java desde cero”

Fecha revisión: 2029

**Resumen:** Entrega nº81 curso Aprender programación Java desde cero.

Autor: Alex Rodríguez

## TIPOS ENUMERADOS (ENUM) EN JAVA

Un tipo enumerado en Java es, de partida, similar al de otros lenguajes de programación. No obstante, incluye una serie de características que lo hacen muy potente. Un tipo enumerado restringe los posibles valores que puede tomar una variable. Esto ayuda a reducir los errores en el código y permite algunos usos especiales interesantes.



Los tipos enumerados se declaran con esta sintaxis tipo:

```
Tipo Público/Privado enum nombreTipoEnumerado { ELEMENTO1, ELEMENTO2, ELEMENTO3, ..., ELEMENTOn };
```

Dentro de las llaves se declaran las variables (objetos) de que consta el tipo enumerado. Por convención, sus nombres se escriben en letras mayúsculas para recordarnos que son valores fijos (que en cierto modo podemos ver como constantes). Una vez declarado el tipo enumerado, todavía no existen variables hasta que no las creamos explícitamente, de la misma manera que ocurre con cualquier tipo Java. Para crear una variable de tipo enumerado lo haremos con una declaración simple que recuerda a la creación de una variable tipo primitivo: *nombreTipoEnumerado miNombreElegido;*

Esta forma de creación de variables de tipo enumerado se justifica porque los tipos enumerados en principio no tienen constructores (más adelante veremos que sí los pueden tener). Los valores de un tipo enumerado son objetos propiamente dichos, ¿de qué tipo? Del tipo enumerado cuyo nombre se haya indicado. No se pueden crear más objetos variantes del tipo enumerado que los especificados en su declaración. Tener en cuenta, porque es una confusión habitual, que los tipos enumerados no son enteros, ni cadenas (aunque a veces podamos hacer que se comporten de forma similar a como lo haría un entero o una cadena). Cada elemento de un enumerado es un objeto único disponible para su uso.

La identificación de cada objeto del tipo se hace con la sintaxis del punto, es decir, nos referimos a un elemento concreto como nombreDelTipoEnumerado.ELEMENTO1. Esta sintaxis nos quiere recordar lo que sería un campo de una clase pero no es así: en este caso es un objeto de un tipo enumerado.

**Un tipo enumerado puede ser declarado dentro o fuera de una clase**, pero no dentro de un método. Por tanto no podemos declarar un enum dentro de un método main (programa principal); si lo hacemos nos saltará el error de compilación *"enum types must not be local"* (los tipos enumerados no deben ser locales a un método). Por tanto el tipo enumerado lo declararemos o bien antes del *public class...* o bien después del *public class...* pero fuera de cualquier método o constructor. Veamos un primer ejemplo de uso:

```
/**
 * Descripción para javadoc antes de la clase. Clase de ejemplo aprenderaprogramar.com para tipos enumerados
 * @author Alex Rodríguez
 */

public class TestEnum {
    enum TipoDeMadera { ROBLE, CAOBA, NOGAL, CEREZO, BOJ };
    public static void main (String[ ] Args) {
        TipoDeMadera maderaUsuario;
        maderaUsuario = TipoDeMadera.ROBLE;
        System.out.println ("La madera elegida por el usuario es " + maderaUsuario.toString().toLowerCase() );
        System.out.println ("¿Es la madera elegida por el usuario caoba? Resultado: " +
(maderaUsuario==TipoDeMadera.CAOBA) );
        System.out.println ("¿Es la madera elegida por el usuario roble? Resultado: " +
(maderaUsuario==TipoDeMadera.ROBLE) );
    } //Cierre del main
} //Cierre de la clase
```

Ejecuta el anterior código. El resultado de la ejecución será similar a este:

```
La madera elegida por el usuario es roble
¿Es la madera elegida por el usuario caoba? Resultado: false
¿Es la madera elegida por el usuario roble? Resultado: true
```

Hemos usado el método toString() que es un método disponible para la mayoría de las clases en Java y utilizado para representar un objeto en una cadena de texto. Ten en cuenta que usamos este método porque el enumerado no es texto.

El enumerado lo hemos incluido como código dentro de la clase. En realidad, podríamos haberlo dispuesto como una clase independiente que llevara únicamente este código:

```
public enum TipoDeMadera { ROBLE, CAOBA, NOGAL, CEREZO, BOJ }
```

## EJERCICIO

Crema una clase Vehiculo donde definas un tipo enumerado MarcaDeVehiculo cuyos valores posibles serán FORD, TOYOTA, SUZUKI, RENAULT, SEAT. Establece como atributos de la clase matricula (tipo String) y marca (tipo MarcaDeVehiculo) y crea los métodos getters (para obtener) y setters (para establecer el valor) de los atributos. Escribe un pequeño programa de prueba donde verifiques que los métodos funcionan correctamente. Puedes comprobar si tu código es correcto consultando en los foros aprenderaprogramar.com.

**Próxima entrega:** CU00682B

**Acceso al curso completo** en aprenderaprogramar.com -- > Cursos, o en la dirección siguiente:

[http://www.aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=68&Itemid=188](http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=68&Itemid=188)