



PARA LOS MANITAS: Forzar la conversión de tipos puede hacer que se pierda parte del valor de un cálculo o de una variable. Utilícelo siempre estando seguro que el resultado es realmente el que desea.

En la conversión de tipos existe un tipo especial de expresiones que involucra a los valores del tipo `char`. De hecho, un tipo `char` siempre se puede utilizar en una expresión junto con números enteros como se muestra en el siguiente ejemplo:

```
char c = 'A';
int n;
n = c + 2;
```

El resultado en la variable `n` es el valor 67. Si el resultado que se deseaba obtener es el carácter dos posiciones posterior al carácter 'A', es decir, el carácter 'C', habría que haber convertido el resultado al tipo `char`, de la siguiente forma:

```
char c = 'A';
c = (char) (c + 2);
```



IMPORTANTE: Aunque los caracteres en Java se pueden emplear en cualquier expresión entera, no es un tipo de operaciones que sea recomendable realizar, salvo que conozca bien qué está haciendo. Este tipo de operaciones no generan ningún error del compilador.

En este caso el resultado hubiese sido de manera esperada el carácter 'C'. De todas formas recuerde que los caracteres en Java siguen la ordenación Unicode, aunque los primeros caracteres coinciden en orden y número con la tabla ASCII.

Enumerados

Como ha visto hasta ahora, es habitual utilizar los tipos anteriores en cualquier programa en Java. También es habitual utilizar elementos más complejos que verá cómo se definen en los próximos capítulos. Los enumerados son conjuntos de valores constantes para los que no existe un tipo predefinido. Por ejemplo, no existe ningún tipo predefinido para representar los días de la semana, las estaciones del año, los meses del año, los turnos de clases, etc.

Para definir un tipo enumerado con sus valores se haría de la siguiente forma:

```
enum DiaSemana {LUNES, MARTES, MIÉRCOLES, JUEVES, VIERNES, SABADO, DOMINGO}
enum TurnoDeClase {MAÑANA, TARDE}
enum TipoDeClase {TEORIA, LABORATORIO, SEMINARIO, CHARLA, EXPERIMENTO}
```

En el Ejemplo 1.3 puede ver un programa que maneja los valores de los días de la semana.

```

public class Dias {
    public enum DiaSemana {LUNES, MARTES, MIÉRCOLES, JUEVES, VIERNES,
    SABADO, DOMINGO}

    public static void main(String[] args) {
        DiaSemana hoy = DiaSemana.JUEVES;
        DiaSemana ultimo = DiaSemana.DOMINGO;

        System.out.println("Hoy es " + hoy);
        System.out.println("El ultimo día es " + ultimo);
    }
}

```

Ejemplo 1.3. Programa que utiliza valores enumerados.

Fijese que para declarar que un conjunto de valores pertenece a un enumerado se pone el nombre que define al enumerado, en este caso, `DiaSemana` y, a continuación, entre llaves la lista de valores separados por coma. Cada uno de los valores es un identificador. Tras la definición no se ha puesto punto y coma, aunque puede llevarlo.

Como podrá comprobar más adelante, la definición de un enumerado es en realidad la declaración de una clase que tiene ese nombre y un conjunto de atributos con los nombres de los valores.

Petición de datos al usuario

Como ya habrá imaginado, lo habitual en un programa es que solicite datos al usuario para realizar los cálculos del programa. En el Ejemplo 1.4 se presenta de nuevo el ejemplo del cálculo del volumen de un cilindro en el que se piden al usuario los datos del radio y la altura.

```

/**
 * Programa en Java que pide al usuario los datos
 * del radio y la altura de un cilindro y calcula su volumen
 */
import java.util.Scanner;
public class PedirDatos {
    public static void main(String[] args) {
        /* El valor del numero pi */
        final double PI = 3.1415926536;
        double radio;
        double altura;

        Scanner teclado = new Scanner(System.in);

        System.out.println("Introduzca los datos del cilindro:");
        System.out.print("Radio: ");
        radio = teclado.nextDouble();
        System.out.print("Altura: ");
        altura = teclado.nextDouble();
        System.out.print("El área del cilindro es: ");
        System.out.println(PI * radio * radio * altura);
    }
}

```

Ejemplo 1.4. Programa con petición de datos al usuario.

