



## PRÁCTICA 3: MIS AMIGOS LOS ENUMERADOS

### 1. Introducción

Los tipos numerados son una gran facilidad que ofrecen los lenguajes de programación para definir tipos que representen un conjunto finito (y corto) bien definido de individuos, los cuales se pueden representar mediante valores literales. Hay muchos casos donde el uso de tipos enumerados es de utilidad. Por ejemplo, son útiles para representar los días de la semana, los baremos de calificación de un examen, los colores del arco iris, los palos de una baraja de cartas, los estados de un cierto objeto, etc. Por fortuna o desgracia, los tipos enumerados con la llegada de los lenguajes modernos tipo Java han ido quedando parcialmente en el olvido, siendo muy pocos los programadores actuales que los usan en su actividad diaria. No obstante, los tipos enumerados aportan bastante seguridad y robustez a los programas, dado que es imposible usar valores incorrectos en aquellos lugares de un programa donde se solicitan valores de tipo enumerado. Sin embargo, su definición puede requerir, dependiendo del lenguaje, cierto esfuerzo inicial que la mayoría de los programadores prefieren obviar. Por tanto, la razón para el no uso de los tipos enumerados en la mayoría de los casos es la pereza del programador, justificación en absoluto admisible.

### 2. Objetivos

El objetivo de esta práctica es implementar un tipo enumerado básico con objeto de conocer las técnicas y problemas propios de la implementación de tipos enumerados en lenguajes modernos. Tras la realización de la práctica, el alumno debe ser capaz de implementar tipos enumerados básicos.

### 3. Actividades

Basándose en la implementación del tipo enumerado “DiaSemana”, disponible en la plataforma moodle, crear el tipo enumerado “NotaMusical”, que soporte las siguientes operaciones:

- (1) `Do():NotaMusical`, `Re():NotaMusical`, `Mi():NotaMusical`,  
`Fa():NotaMusical`, `Sol():NotaMusical`, `La():NotaMusical`,  
`Si():NotaMusical`. No aceptan ningún parámetro y devuelven el valor del tipo enumerado que corresponde al nombre de la función.
- (2) `Siguiente(n: NotaMusical) : NotaMusical`. Devuelve la nota siguiente a la que se pasa como parámetro de acuerdo al orden la escala musical. La nota siguiente al Si es un Do.
- (3) `Anterior(n: NotaMusical) : NotaMusical`. Devuelve la nota anterior a la que se pasa como parámetro de acuerdo al orden la escala musical. La nota anterior al Do es un Si.
- (4) `LeeNota(s : CadenaCaracterres): NotaMusical`. Muestra el mensaje `s` por pantalla y lee una nota por teclado. Se puede solicitar tantas veces como sea necesario el valor de la nota por teclado.



- (5) `aCadena(s: NotaMusical): CadenaCaracteres`. Devuelve una cadena de caracteres con el valor de una nota de acuerdo al sistema latino (Do, Re, Mi Fa, Sol, La, Si).
- (6) `aCadenaIngles(s: NotaMusical): CadenaCaracteres`. Devuelve una cadena de caracteres el valor de una nota de acuerdo al sistema inglés (C, D, E, F, G, A, B) donde C se corresponde con Do, D con Re, y así sucesivamente.

#### 4. Criterios de Evaluación y Aclaraciones

Para la realización de la práctica se proporciona un ejemplo del uso de los enumerados en Java a través de la plataforma moodle. Se recomienda inspeccionar dicho ejemplo antes de realizar la práctica.

La práctica se entregará a través de la plataforma *moodle* siguiendo las instrucciones en ella proporcionadas.

*Pablo Sánchez Barreiro.*