

Clase 32

Lunes, 13 Noviembre 2023

[Indice](#)

Recordatorios

- Habitualmente una función de objeto, que lleva **'void'** no tiene parametros de entrada.
- Al revés, una función de clase, que lleva **'static'** recibe todos los datos de entrada por parámetros.
- Utilizamos tres barras **'///'** para documentar funciones. Le indicamos el uso de la función, los parámetros que recibe y el retorno.
- La instrucción **'break'** coloca la linea de compilación fuera del bucle más inmediato. Los bucles son **'while'** y **'for'**.
- Podemos averiguar el valor numérico de un **'char'** mediante un casting de **'int'**.

```
charExample = 'b';  
int characterNumber = (int)charExample;
```

A la inversa también funciona, averiguar qué char es un numero en concreto.

Clase 'Datetime'

- Diseño de una nueva clase, llamada **'Datetime'**. Tiene los siguientes atributos:

```
DATETIME  
-----  
-day: int  
-month: int  
-year: int  
-seconds: int  
-minutes: int  
-hours: int  
-----  
+Datetime()  
+Datetime(day, month, year)  
+Datetime(day, month, year, seconds, minutes, hours)  
-----  
+Clone(): Datetime  
+Equals(Datetime): bool  
+IsValid(): bool  
+IsLeap(): bool  
+IsLeap'static'(year: int): bool  
+ToString(): string  
+GetDaysCount'static'(year, month): int  
+IncrementDay(): int  
+IncrementSecond(): int  
+GetDayOfWeek(): DayOfTheWeek
```

- Datetime mide los momentos en los que ocurre algún hecho.
- Las fechas se almacenan con enteros. Todas se miden contando los segundos desde la fecha: 1 de Enero de 1970
- Los constructores que mantienen un mismo orden son más accesibles.
- La función **'static'** del año bisiesto requiere de un parametro, ya que es imposible acceder desde un objeto año.