

Calendar

Es una clase que integra el paquete `java.util` a partir de Java 5 SE sustituyó a la clase `Date`.

Una instancia de la clase `Calendar` representa una fecha y una hora.

La clase `Calendar` trae implementados distintos métodos que nos permiten manipular fechas.

Para obtener una instancia de `Calendar` debemos ejecutar:

```
Calendar hoy = Calendar.getInstance();
```

De esta forma la variable de tipo `Calendar` **hoy** va a estar inicializada con la fecha y hora actuales para la máquina en la cual se ejecuta la instrucción.

Si quisiéramos ver estos valores en pantalla debemos ejecutar:

```
System.out.printf("Día: %1$td Mes: %1$tm Año: %1$tY \n", hoy);
```

`1$` es un índice que indica que cada uno de los formatos que estamos especificando corresponde aplicarlos sobre el primer parámetro con datos.

`%t` es el especificador de formato de **printf** que debemos utilizar para imprimir fechas y horas.

%t siempre va seguido de un carácter de sufijo de conversión que especifica el formato, caso contrario produce un error en tiempo de ejecución.

Los caracteres de sufijo de conversión más utilizados con `%t` son los siguientes:

<code>%tc</code>	Muestra los datos completos. Ejemplo: mar abr 14 19:30:45 GMT-03:00 2020
<code>%tr</code>	Sólo hora con AM/PM. Ejemplo: 07:30:45 PM
<code>%tR</code>	Sólo hora formato 24 hs. Ejemplo: 19:30:45
<code>%tA</code>	Sólo nombre completo del día de la semana Ejemplo: martes
<code>%ta</code>	Sólo nombre abreviado del día de la semana Ejemplo: mar
<code>%tB</code>	Sólo nombre completo del mes Ejemplo: abril
<code>%tb</code>	Sólo nombre abreviado del mes Ejemplo: abr
<code>%td</code>	Sólo el día en 2 dígitos Ejemplo: 14
<code>%tm</code>	Sólo el mes en 2 dígitos Ejemplo: 04
<code>%tY</code>	Sólo el año en 4 dígitos Ejemplo: 2020
<code>%ty</code>	Sólo el año en 2 dígitos Ejemplo: 20

La única forma de obtener una instancia de `Calendar` es enviándole el mensaje **.getInstance()**. Si necesitamos una instancia de `Calendar` pero con valores diferentes para la fecha deberíamos primero limpiarla usando **.clear()** y luego asignarle los nuevos valores.

Cuando se ejecuta **.clear()** sobre una instancia de *Calendar* el objeto es inicializado con un conjunto de valores por default denominado `Lenient`.

Para evaluar si una instancia de fecha está seteada con estos valores y por lo tanto se la debe considerar como fecha “vacía”, hay que ejecutar **.isLenient()** que devuelve *true* ó *false*.

Para asignar valores a los atributos de hay que usar **.set(índice, valor)**.

Donde *índice* es la posición que ocupa el atributo que deseamos cambiar.

Para obtener la posición que ocupa un atributo debemos recuperar el *valor* asignado al atributo de nuestro interés.

Supongamos:

```
Calendar fecha = Calendar.getInstance( );  
fecha.clear( );
```

fecha. <i>DAY_OF_MONTH</i>	Devuelve la posición del atributo día
fecha. <i>MONTH</i>	Devuelve la posición del atributo mes
fecha. <i>YEAR</i>	Devuelve la posición del atributo año
fecha. <i>HOURL</i>	Devuelve la posición del atributo horas
fecha. <i>MINUTE</i>	Devuelve la posición del atributo minutos
fecha. <i>SECOND</i>	Devuelve la posición del atributo segundos

Para asignar a la variable *fecha* los valores que representan al 12/09/2000 deberíamos ejecutar:

```
fecha.set(fecha.DAY_OF_MONTH, 12);  
fecha.set(fecha.MONTH, 9);  
fecha.set(fecha.YEAR, 2000);
```

Para recuperar en las variables dd, mm, aaaa los valores que contiene la variable *fecha* de tipo *Calendar* deberíamos ejecutar:

```
int dd = fecha.get(fecha.DAY_OF_MONTH);  
int mm = fecha.get(fecha.MONTH) + 1;  
int aaaa = fecha.get(fecha.YEAR);
```