

# Programación Frontend y Backend

*BLOQUE JAVA*

Utilidades



01

Scanner



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMO

**EQI**

Escuela de  
organización  
industrial



**Unión Europea**  
**Fondo Social Europeo**  
**Iniciativa de Empleo Juvenil**  
El FSE invierte en tu futuro



## SCANNER

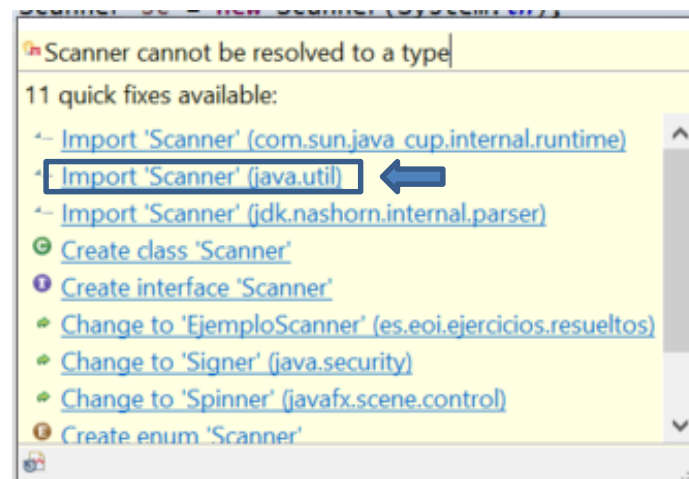
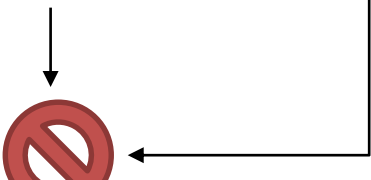
- Scanner es una clase en el paquete java.util utilizada para obtener la entrada de los tipos primitivos y también String. Es la forma más fácil de leer datos en un programa Java

```
Scanner sc = new Scanner(System.in);  
String input = sc.nextLine();
```

API SCANNER: <https://docs.oracle.com/javase/8/docs/api/java/util/Scanner.html>

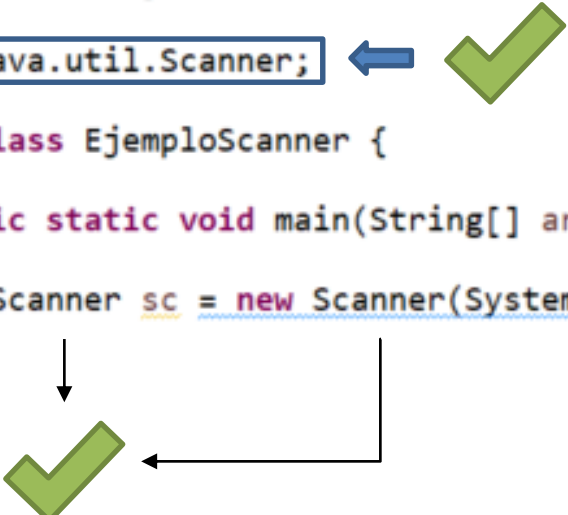
## DECLARACIÓN SCANNER

```
1 package es.eoi.ejercicios.resueltos;  
2  
3 public class EjemploScanner {  
4  
5     public static void main(String[] args) {  
6  
7         Scanner sc = new Scanner(System.in);  
8  
9     }  
10 }  
11  
12 }
```



## DECLARACIÓN SCANNER

```
1 package es.eoi.ejercicios.resueltos;  
2  
3 import java.util.Scanner; ←  
4  
5 public class EjemploScanner {  
6  
7     public static void main(String[] args) {  
8  
9         Scanner sc = new Scanner(System.in);  
10  
11  
12     }  
13  
14 }
```



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMO



Escuela de  
organización  
industrial

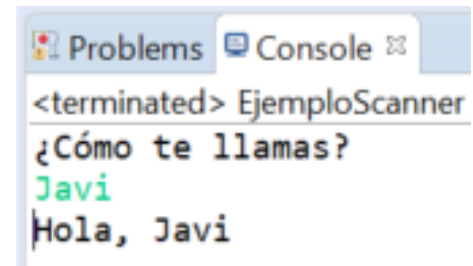


Unión Europea  
Fondo Social Europeo  
Iniciativa de Empleo Juvenil  
El FSE invierte en tu futuro



## SCANNER

```
Scanner sc = new Scanner(System.in);  
  
System.out.println("¿Cómo te llamas?");  
  
String nombre = sc.nextLine();  
  
System.out.println("Hola, "+nombre);  
  
sc.close();
```



The screenshot shows an IDE window with two tabs: 'Problems' and 'Console'. The 'Console' tab is active, displaying the output of the program. The text in the console is as follows:

```
<terminated> EjemploScanner  
¿Cómo te llamas?  
Javi  
Hola, Javi
```

boolean	<b>nextBoolean()</b> Scans the next token of the input into a boolean value and returns that value.
byte	<b>nextByte()</b> Scans the next token of the input as a byte.
byte	<b>nextByte(int radix)</b> Scans the next token of the input as a byte.
double	<b>nextDouble()</b> Scans the next token of the input as a double.
float	<b>nextFloat()</b> Scans the next token of the input as a float.
int	<b>nextInt()</b> Scans the next token of the input as an int.
int	<b>nextInt(int radix)</b> Scans the next token of the input as an int.
<b>String</b>	<b>nextLine()</b> Advances this scanner past the current line and returns the input that was skipped.
long	<b>nextLong()</b> Scans the next token of the input as a long.
long	<b>nextLong(int radix)</b> Scans the next token of the input as a long.
short	<b>nextShort()</b> Scans the next token of the input as a short.
short	<b>nextShort(int radix)</b> Scans the next token of the input as a short.

```
Scanner sc = new Scanner(System.in);  
  
System.out.println("¿Qué edad tienes?");  
  
int edad = sc.nextInt();  
  
System.out.println("Tienes, " + edad + " años.");  
  
sc.close();
```



```
<terminated> EjemploScanner [Java Application]  
¿Qué edad tienes?  
18  
Tienes, 18 años.
```





```
Scanner sc = new Scanner(System.in);  
  
System.out.println("¿Qué edad tienes?");  
  
int edad = sc.nextInt();  
  
System.out.println("Tienes, " + edad + " años.");  
  
sc.close();
```

```
Problems Console  
<terminated> EjemploScanner [Java Application] C:\ej_ico_0\Java\jdk1.8.0_144\bin\javaw.exe (7 nov. 2020 13:42:59)  
¿Qué edad tienes?  
dieciocho  
Exception in thread "main" java.util.InputMismatchException  
    at java.util.Scanner.throwFor(Scanner.java:864)  
    at java.util.Scanner.next(Scanner.java:1485)  
    at java.util.Scanner.nextInt(Scanner.java:2117)  
    at java.util.Scanner.nextInt(Scanner.java:2076)  
    at es.eoi.ejercicios.resueltos.EjemploScanner.main(EjemploScanner.java:13)
```

02

Math



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMO

**EQI**

Escuela de  
organización  
industrial



**Unión Europea**  
**Fondo Social Europeo**  
**Iniciativa de Empleo Juvenil**  
El FSE invierte en tu futuro



## MATH

- La clase Math recoge una buena colección de funciones matemáticas, estas pueden cambiar en función de la versión de Java que estemos utilizando.

**API MATH:** <https://docs.oracle.com/javase/8/docs/api/java/lang/Math.html>

Función matemática	Significado	Ejemplo de uso	Resultado
abs	Valor absoluto	<code>int x = Math.abs(2.3);</code>	<code>x = 2;</code>
atan	Arcotangente	<code>double x = Math.atan(1);</code>	<code>x = 0.78539816339744;</code>
sin	Seno	<code>double x = Math.sin(0.5);</code>	<code>x = 0.4794255386042;</code>
cos	Coseno	<code>double x = Math.cos(0.5);</code>	<code>x = 0.87758256189037;</code>
tan	Tangente	<code>double x = Math.tan(0.5);</code>	<code>x = 0.54630248984379;</code>
exp	Exponenciación neperiana	<code>double x = Math.exp(1);</code>	<code>x = 2.71828182845904;</code>
log	Logaritmo neperiano	<code>double x = Math.log(2.7172);</code>	<code>x = 0.99960193833500;</code>
pow	Potencia	<code>double x = Math.pow(2.3);</code>	<code>x = 8.0;</code>
round	Redondeo	<code>double x = Math.round(2.5);</code>	<code>x = 3;</code>
random	Número aleatorio	<code>double x = Math.random();</code>	<code>x = 0.20614522323378;</code>
floor	Redondeo al entero menor	<code>double x = Math.floor(2.5);</code>	<code>x = 2.0;</code>
ceil	Redondeo al entero mayor	<code>double x = Math.ceil(2.5);</code>	<code>x = 3.0;</code>

GOBIERNO  
DE ESPAÑAMINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMOEscuela de  
organización  
industrial

**Unión Europea**  
**Fondo Social Europeo**  
**Iniciativa de Empleo Juvenil**  
 El FSE invierte en tu futuro



## MATH (REDONDEOS)

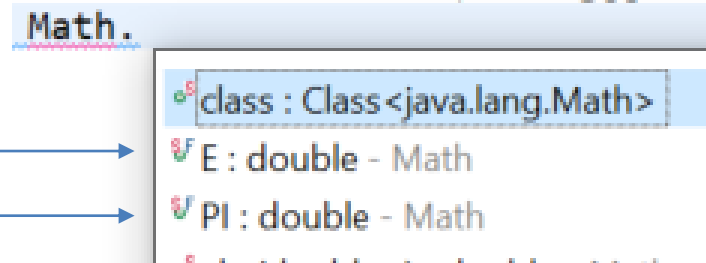
Valor inicial	2.6	-2.6	2.4	-2.4	2.5	-2.5
Resultado con round	3	-3	2	-2	3	-2
Resultado con ceil	3.0	-2.0	3.0	-2.0	3.0	-2.0
Resultado con floor	2.0	-3.0	2.0	-3.0	2.0	-3.0



## MATH (CONSTANTES)

- La clase Math define 2 constantes:

- E
- PI



```
System.out.println("Este es el número PI: " + Math.PI);
```

A screenshot of an IDE's console window. The title bar shows 'Problems' and 'Console'. The output text is: `<terminated> EjemploMath [Java Application] C:\ej_ico` followed by a new line with `Este es el número PI: 3.141592653589793`.

## MATH (FUNCIONES MUY UTILIZADAS)

```
System.out.println("10.0 elevado a 3.5 es:" + Math.pow(10.0, 3.5));
```

```
System.out.println("La raíz cuadrada de 9 es:" + Math.sqrt(9));
```

Problems Console

```
<terminated> EjemploMath [Java Application] C:\ej_ico_
10.0 elevado a 3.5 es:3162.2776601683795
La raíz cuadrada de 9 es:3.0
```

## MATH (NÚMEROS ALEATORIOS)

```
for (int i = 1; i <= 10; i++) {  
    System.out.println("Número aleatorio "+i+": " + Math.random());  
}
```

<terminated> EjemploMath [Java Application] C:\ej\_ico\_0\Java\

Número aleatorio 1: 0.41433414922720224  
Número aleatorio 2: 0.24766849954928272  
Número aleatorio 3: 0.38474416255492594  
Número aleatorio 4: 0.7064028947219297  
Número aleatorio 5: 0.80902815505705  
Número aleatorio 6: 0.32540438096040447  
Número aleatorio 7: 0.23836434129535522  
Número aleatorio 8: 0.8714265042775421  
Número aleatorio 9: 0.8866641758596403  
Número aleatorio 10: 0.4128812439535017

$0 \geq \&\& \leq 1$



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMO



Escuela de  
organización  
industrial



Unión Europea  
Fondo Social Europeo  
Iniciativa de Empleo Juvenil  
El FSE invierte en tu futuro





## MATH (NÚMEROS ALEATORIOS + REDONDEO)

```
for (int i = 1; i <= 10; i++) {  
    double aleatorio = Math.random();  
    System.out.println("Número aleatorio "+i+": (" + aleatorio + ") " + Math.round(aleatorio));  
}
```

<terminated> EjemploMath [Java Application] C:\ej\_ico\_0\Java\jdk1.8

Número aleatorio 1: (0.905852115479154) 1  
Número aleatorio 2: (0.23039552717831002) 0  
Número aleatorio 3: (0.7240898401271527) 1  
Número aleatorio 4: (0.9792183840056303) 1  
Número aleatorio 5: (0.11892373041815174) 0  
Número aleatorio 6: (0.08903841015576952) 0  
Número aleatorio 7: (0.9303567536833459) 1  
Número aleatorio 8: (0.5332604702008462) 1  
Número aleatorio 9: (0.3435944508026181) 0  
Número aleatorio 10: (0.09238572676415135) 0

$0 \geq \&\& \leq 1$

03

String



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMO



Escuela de  
organización  
industrial



**Unión Europea**  
**Fondo Social Europeo**  
**Iniciativa de Empleo Juvenil**  
El FSE invierte en tu futuro



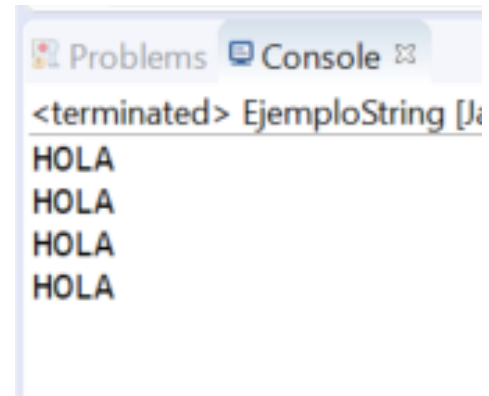
## STRING

- Un String en Java representa una cadena de caracteres no modificable.
- Se puede crear un String de varias formas.

**API STRING:** <https://docs.oracle.com/javase/8/docs/api/java/lang/String.html>

## STRING

```
String str1 = "HOLA";  
System.out.println(str1);  
  
String str2 = "H"+"O"+"L"+"A";  
System.out.println(str2);  
  
String strAux1 = "HO";  
String strAux2 = "LA";  
String str3 = strAux1 + strAux2;  
System.out.println(str3);  
  
char caracteres[] = { 'H', 'O', 'L', 'A' };  
String str4 = new String(caracteres);  
System.out.println(str4);
```



## STRING

- Obtener cada carácter según su índice, como un array.

```
System.out.println(str1.charAt(0));
```

H

- Concatenar varios objetos de tipo String.

```
System.out.println(str1.concat(" JAVI"));
```

HOLA JAVI

- Comprobar si un cierto String es contenido dentro de otro.

```
System.out.println(str1.contains("OLA"));
```

true

```
System.out.println(str1.contains("ola"));
```

false

GOBIERNO  
DE ESPAÑAMINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMOEscuela de  
organización  
industrialUnión Europea  
Fondo Social Europeo  
Iniciativa de Empleo Juvenil  
El FSE invierte en tu futuro

## STRING

- Comprobar si un String empieza o acaba por un cierto carácter o caracteres.

```
System.out.println(str1.startsWith("H")); → true  
System.out.println(str1.endsWith("LA")); → true
```

- Comprobar si dos String son iguales.

```
System.out.println(str1.equals("HOLA")); → true  
System.out.println(str1.equals("hola")); → false  
System.out.println(str1.equalsIgnoreCase("HOLA")); → true  
System.out.println(str1.equalsIgnoreCase("hola")); → true
```

- Devuelve el índice dónde encuentra la primera o última ocurrencia de otro String.

```
System.out.println("SARA".indexOf("A")); → 1  
System.out.println("SARA".indexOf("E")); → -1  
System.out.println("SARA".lastIndexOf("A")); → 3  
System.out.println("SARA".lastIndexOf("E")); → -1
```

## STRING

- Comprobar si un String no tiene valor, está vacío.

```
System.out.println("").isEmpty(); → true  
System.out.println("A").isEmpty(); → false
```

- Comprobar cuantos caracteres tiene el String.

```
System.out.println("").length(); → 0  
System.out.println("A").length(); → 1  
System.out.println("A B C").length(); → 5
```

- Devuelve un nuevo String dónde reemplaza una parte del String con otro (Busca y Reemplaza).

```
System.out.println("MOTO".replace("TO", "NO")); → MONO
```

- Devuelve un array de String[] al trocear un String, dado un delimitador.

```
System.out.println("1A,1B,1C,1D".split(",").length); → 4
```

## STRING

- Devuelve un String con una subcadena de la cadena principal

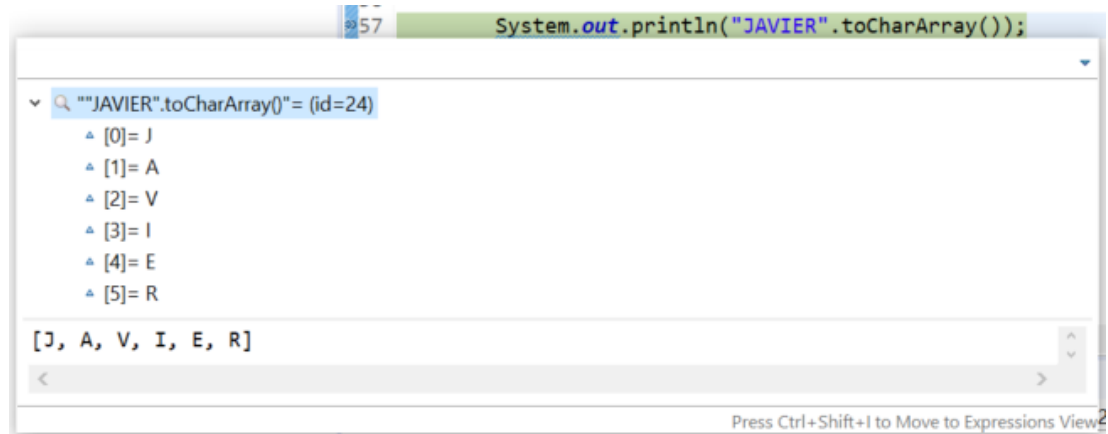
```
System.out.println("JAVIER".substring(2));  
System.out.println("JAVIER".substring(0,4));
```

VIER

JAVI

X	X	X	X		
0	1	2	3	4	5
J	A	V	I	E	R

- Crea un array de char[] dado un String





## STRING

- Devuelve un String sin espacios por la izquierda y la derecha:

```
System.out.println(" HOLA ".trim());
```

—————→ HOLA

- Devuelve un String convertido en mayúsculas o minúsculas:

```
System.out.println(" HOLA ".toLowerCase());
```

—————→ hola

```
System.out.println(" hola ".toUpperCase());
```

—————→ HOLA

04

Java Time



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMO

**EQI**

Escuela de  
organización  
industrial



**Unión Europea**  
**Fondo Social Europeo**  
**Iniciativa de Empleo Juvenil**  
El FSE invierte en tu futuro



## JAVA TIME

- En Java 8 se incluyen clases para la gestión del tiempo:
  - LocalDate
  - LocalTime
  - LocalDateTime

API LOCAL DATE: <https://docs.oracle.com/javase/8/docs/api/java/time/LocalDate.html>

API LOCAL TIME: <https://docs.oracle.com/javase/8/docs/api/java/time/LocalTime.html>

API LOCAL DATE TIME: <https://docs.oracle.com/javase/8/docs/api/java/time/LocalDateTime.html>

## JAVA TIME

```
LocalDate fechaActual = LocalDate.now();  
LocalTime horaActual = LocalTime.now();  
LocalDateTime fechaHoraActual = LocalDateTime.now();
```

```
System.out.println(fechaActual);  
System.out.println(horaActual);  
System.out.println(fechaHoraActual);
```

```
System.out.println(LocalTime.now());  
System.out.println(LocalDate.now());  
System.out.println(LocalDateTime.now());
```

```
<terminated> EjemploDateTime [Java Application] C:\ej_ico_0\Java\jdk1.8.0_144\bin\javaw.exe (7 nov. 2020 18:51:43)
```

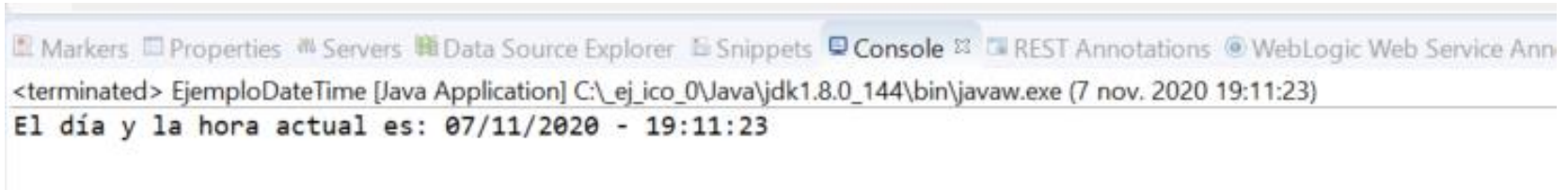
```
18:51:44.470
```

```
2020-11-07
```

```
2020-11-07T18:51:44.477
```

## FORMATO

```
LocalDateTime ahora = LocalDateTime.now();
DateTimeFormatter formatoDiaHora = DateTimeFormatter.ofPattern("dd/MM/yyyy - HH:mm:ss");
System.out.println("El día y la hora actual es: " + ahora.format(formatoDiaHora));
```



The screenshot shows an IDE console window with the following content:

```
<terminated> EjemploDateTime [Java Application] C:\ej_ico_0\Java\jdk1.8.0_144\bin\javaw.exe (7 nov. 2020 19:11:23)
El día y la hora actual es: 07/11/2020 - 19:11:23
```

## LECTURA ATRIBUTOS

07/11/2020 - 19:20:35

```
//Obtenemos ciertos valores (Fecha)
System.out.println(ahora.getDayOfMonth()); → 7
System.out.println(ahora.getDayOfWeek()); → SATURDAY
System.out.println(ahora.getDayOfYear()); → 312
System.out.println(ahora.getMonth()); → NOVEMBER
System.out.println(ahora.getMonthValue()); → 11
System.out.println(ahora.getYear()); → 2020

//Obtenemos ciertos valores (Hora)
System.out.println(ahora.getHour()); → 19
System.out.println(ahora.getMinute()); → 20
System.out.println(ahora.getSecond()); → 35
```

GOBIERNO  
DE ESPAÑAMINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMOEscuela de  
organización  
industrial**Unión Europea**  
**Fondo Social Europeo**  
**Iniciativa de Empleo Juvenil**  
El FSE invierte en tu futuro

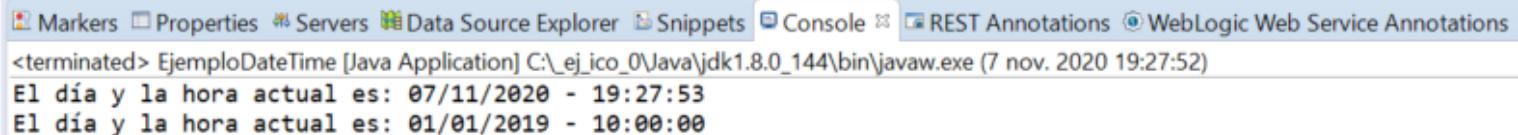
## MODIFICACIÓN ATRIBUTOS

```
LocalDateTime ahora = LocalDateTime.now();
DateTimeFormatter formatoDiaHora = DateTimeFormatter.ofPattern("dd/MM/yyyy - HH:mm:ss");
System.out.println("El día y la hora actual es: " + ahora.format(formatoDiaHora));

ahora = ahora.withDayOfMonth(1);
ahora = ahora.withMonth(1);
ahora = ahora.withYear(2019);

ahora = ahora.withHour(10);
ahora = ahora.withMinute(0);
ahora = ahora.withSecond(0);

System.out.println("El día y la hora actual es: " + ahora.format(formatoDiaHora));
```



Markers Properties Servers Data Source Explorer Snippets Console REST Annotations WebLogic Web Service Annotations

<terminated> EjemploDateTime [Java Application] C:\ej\_ico\_0\Java\jdk1.8.0\_144\bin\javaw.exe (7 nov. 2020 19:27:52)

El día y la hora actual es: 07/11/2020 - 19:27:53

El día y la hora actual es: 01/01/2019 - 10:00:00



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMO



Escuela de  
organización  
industrial



Unión Europea  
Fondo Social Europeo  
Iniciativa de Empleo Juvenil  
El FSE invierte en tu futuro





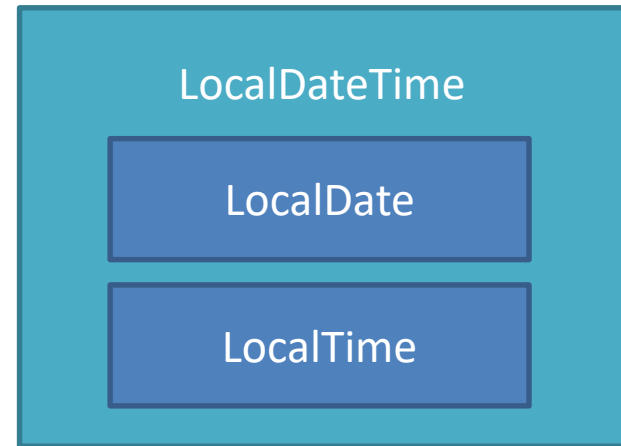
## MODIFICACIÓN ATRIBUTOS

<code>LocalDateTime</code>	<b><code>plusDays(long days)</code></b> Returns a copy of this <code>LocalDateTime</code> with the specified number of days added.
<code>LocalDateTime</code>	<b><code>plusHours(long hours)</code></b> Returns a copy of this <code>LocalDateTime</code> with the specified number of hours added.
<code>LocalDateTime</code>	<b><code>plusMinutes(long minutes)</code></b> Returns a copy of this <code>LocalDateTime</code> with the specified number of minutes added.
<code>LocalDateTime</code>	<b><code>plusMonths(long months)</code></b> Returns a copy of this <code>LocalDateTime</code> with the specified number of months added.
<code>LocalDateTime</code>	<b><code>plusNanos(long nanos)</code></b> Returns a copy of this <code>LocalDateTime</code> with the specified number of nanoseconds added.
<code>LocalDateTime</code>	<b><code>plusSeconds(long seconds)</code></b> Returns a copy of this <code>LocalDateTime</code> with the specified number of seconds added.
<code>LocalDateTime</code>	<b><code>plusWeeks(long weeks)</code></b> Returns a copy of this <code>LocalDateTime</code> with the specified number of weeks added.
<code>LocalDateTime</code>	<b><code>plusYears(long years)</code></b> Returns a copy of this <code>LocalDateTime</code> with the specified number of years added.





## ESTRUCTURA LOCAL DATE TIME

**LocalDate****toLocalDate()**Gets the **LocalDate** part of this date-time.**LocalTime****toLocalTime()**Gets the **LocalTime** part of this date-time.GOBIERNO  
DE ESPAÑAMINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMOEscuela de  
organización  
industrial**Unión Europea**  
**Fondo Social Europeo**  
**Iniciativa de Empleo Juvenil**  
El FSE invierte en tu futuro

## COMPARACIÓN

```
LocalDateTime fecha1 = LocalDateTime.of(2020, 11, 7, 0, 0);  
LocalDateTime fecha2 = LocalDateTime.of(2020, 11, 7, 0, 1);  
  
System.out.println(fecha1.isAfter(fecha2)); → false  
System.out.println(fecha1.isBefore(fecha2)); → true  
System.out.println(fecha1.isEqual(fecha2)); → false  
  
fecha1.getDayOfWeek() == DayOfWeek.MONDAY → false
```

05

## Ejercicios



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMO



Escuela de  
organización  
industrial



**Unión Europea**  
**Fondo Social Europeo**  
**Iniciativa de Empleo Juvenil**  
El FSE invierte en tu futuro



## Ejercicio I

Vamos a crear la clase Calculadora.java, el programa mostrará el siguiente menú:

### **CALCULADORA**

**1 – Sumar**

**2 – Restar**

**3 – Multiplicar**

**4 – Dividir**

**5 – Valor Absoluto**

**6 – Mostrar PI**

**Elige una opción:**

En caso de elegir una opción entre 1 y 4, el programa pedirá introducir dos números, tras introducirlos, calcularemos el resultado de la operación seleccionada para los dos números, se mostrará el siguiente mensaje: **{numero1} {símbolo} {numero2} = {resultado}** (Utilizaremos el símbolo adecuado en cada ocasión)

En el caso de seleccionar la opción 5, únicamente pedirá un número en vez de dos, mostraremos: **El valor absoluto de {numero} es: {valorAbsoluto}** y en caso de seleccionar la 6, mostraremos el valor de PI.

## Ejercicio II

Vamos a crear la clase Poema.java, el programa contendrá un String con el siguiente contenido:

***No entres dócilmente en esa buena noche.  
La vejez debería delirar y arder cuando se acaba el día.  
Rabia, rabia, contra la luz que se esconde.  
Aunque el sabio cerca del fin a la tiniebla no haga reproche,  
dado que a su verbo ningún rayo ha confiado vigor,  
no entra dócilmente en esa buena noche.  
Rabia, rabia, contra la luz que se esconde.***

Cuando ejecutemos el programa deberemos mostrar los siguientes datos por pantalla:

- Nº Vowels – A/a: x
- Nº Vowels – E/e: x
- Nº Vowels – I/i: x
- Nº Vowels – O/o: x
- Nº Vowels – U/u: x
- Nº Caracteres Totales: x

## Ejercicio III

Vamos a crear la clase Cadenas.java, el programa nos pedirá que introduzcamos una cadena de texto, debe tener un tamaño mínimo de 30 caracteres, de no ser así el programa se lo notificará al usuario y volverá a pedirla hasta que el usuario introduzca una cadena de texto mayor o igual a 30 caracteres.

Tras haber introducido la cadena correctamente, se mostrarán las siguientes opciones:

### **CADENAS**

- 1 – Convertir en Mayúsculas**
- 2 – Contar Vocales** (Cómo en el ejercicio anterior)
- 3 – Mostrar subcadena** (Pediremos índiceDesde e índiceHasta)
- 4 – Cambiar palabra** (Pediremos la palabra a reemplazar y la nueva palabra)
- 5 – Visualizar texto.**
- 6 - SALIR**

Deberemos controlar que el índiceDesde y el índiceHasta son inferiores al tamaño de la cadena, si son mayores mostraremos un mensaje al usuario indicando el error, también deberemos controlar que el índiceHasta no sea inferior al índiceDesde.

El menú se repetirá infinitamente hasta que seleccionemos la opción 6.

## Ejercicio IV

Vamos a crear un ejercicio que liste los días del mes en el que nos encontramos:

**01/11/2020**

**02/11/2020**

...

No podemos asignar nosotros el mes, se debe obtener automáticamente con lo visto en este tema.

A continuación escribiremos la siguiente salida:

### NOVEMBER

LU	MA	MI	JU	VI	SA	DO
						01
02	03	04	05	06	07	08
09	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30						



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE INDUSTRIA, COMERCIO  
Y TURISMO



Escuela de  
organización  
industrial



Unión Europea  
Fondo Social Europeo  
Iniciativa de Empleo Juvenil  
El FSE invierte en tu futuro

