



**la clase Math**

**El tipo String**

**Leer caracteres y alfanuméricos**

**Obtener números aleatorios**

**Obtener la fecha del sistema**

**El método `System.out.printf( )`**

**Argumentos en la línea de comandos**

# **MISCELÁNEA I**

**CONCEPTOS, IDEAS, TÉCNICAS...  
QUE VAN SALIENDO EN LOS EJERCICIOS**

# Algunos métodos predefinidos. La clase Math

---

## ■ Las constantes E y PI

**Math.E**=2.7182818284590452354

**Math.PI**=3.14.159265358979323846

## ■ Algunos métodos de cálculo:

- **pow(x,y)** : calcula x elevado a y
- **random()** : genera un número (pseudo)aleatorio entre 0.0 y 1.0
- **sqrt(x)** : calcula la raíz cuadrada de x ( x no negativo)
- **abs(x)** : calcula el valor absoluto de x (entero o real)
- **exp(x)** : calcula e elevado a x (x es real)
- **log(x)** : calcula el logaritmo natural de x (x real y no negativo)
- **max(x,y)** : compara los números x e y (enteros o reales) y devuelve el mayor
- **min(x,y)** : compara los números x e y (enteros o reales) y devuelve el menor

# Algunas funciones predefinidas. La clase Math

---

## ■ Métodos de redondeo, con $x$ de tipo double:

- $\text{ceil}(x)$  : devuelve el número entero más pequeño que es mayor o igual a  $x$
- $\text{floor}(x)$  : devuelve el número entero más grande que es menor o igual a  $x$
- $\text{round}(x)$  : convierte el real  $x$  al entero más próximo

## ■ Métodos trigonométricos

- $\sin(x)$  : calcula el seno del ángulo (en radianes)  $x$
- $\cos(x)$  : calcula el coseno del ángulo (en radianes)  $x$
- $\text{asin}(x)$  : calcula el arco seno del ángulo  $x$  ( $x$  entre -1 y 1)
- $\text{acos}(x)$  : calcula el arco coseno del ángulo  $x$  ( $x$  entre -1 y 1)
- $\text{atan}(x)$  : calcula el arco tangente del ángulo  $x$

# Tipo String. Uso sencillo

---

- El tipo String permite representar secuencias de caracteres
- Ejemplo:

```
String frase, palabra, linea;
```

```
frase="En un lugar de la Mancha de cuyo nombre..."
```

```
//.....solicitamos al usuario una frase
```

```
//..... y la Frase de entrada es: Oh! es terrible
```

```
palabra=tcl.next(); //palabra = Oh!
```

```
linea=tcl.nextLine(); // linea = Oh! es terrible
```

# Leer caracteres y alfanuméricos

- Para solicitar un carácter o un alfanumérico al usuario en nuestro programa, puedo hacer lo siguiente:

```
String resp;  
//Declaro una variable resp de tipo String ...0 más correcto  
//creo un objeto de la clase String y lo referencio con resp  
....  
Scanner lector = new Scanner(System.in);  
resp = lector.nextLine( );  
//leo una tira de caracteres que guardo en resp  
...  
if ( resp.equalsIgnoreCase("Si") == 0 )  
//comparo la tira introducida con otra. En el ejemplo con "SI" ó "si"  
If (resp.charAt(0) == 's' )  
//comparo el primer carácter introducido con otro carácter
```

**IMPORTANTE:** será un error escribir algo tipo

```
if ( resp == "si" )
```

# Obtener números aleatorios. Una opción

---

- El método `Math.random( )` devuelve un número `double` pseudoaleatorio entre 0 y 1
  - A partir de este método ajusto los límites y convertimos a entero
  - Para obtener un número entero entre dos valores **valorInicial** y **valorFinal**, ambos incluidos, debemos usar la fórmula:

`( int ) ( Math.random() * (valorFinal-valorInicial+1) + valorInicial )`

Por ejemplo, para obtener un número del 1 al 6

```
dado = (int) ( Math.random( ) * 6 + 1 );
```

# Obtener números aleatorios. Otra opción

---

- Para que Java genere números aleatorios podemos usar la clase Random

```
import java.util.*;
...
Random rnd = new Random( );
//creamos un objeto Random
...
int numero =rnd.nextInt(1000);
//solicitamos un número entero entre 0 y 1000. Lo guardamos en la
variable numero
```

Nota:

```
// podemos re-establecer la semilla a partir de la cual se
generarán los números pseudo-aleatorios a partir de la hora del
sistema
rnd.setSeed(new Date().getTime());
```

# Obtener la fecha del sistema

---

```
import java.util.*;
```

```
GregorianCalendar fecha= new GregorianCalendar();
```

```
int diaHoy= fecha.get(GregorianCalendar.DAY_OF_MONTH);
```

```
int mesHoy= fecha.get(GregorianCalendar.MONTH)+1;
```

```
int anyoHoy=fecha.get(GregorianCalendar.YEAR)  ;
```

```
System.out.printf("%d/%d/%d \n\n" ,diaHoy,mesHoy,anyoHoy);
```



# El método `System.out.printf( )`

---

- Podemos usar `System.out.printf( )` para formatear los datos de salida

`printf ( Cadena con formato, argumentos sustituidos en la cadena)`

- Usaremos `%d` para valores enteros (en sistema decimal)

`System.out.printf("Al sumar %d mas %d, resulta: %d", num1, num2, suma);`

- En este caso el primer `%d` (`%d` significa numero entero) se sustituye con el primer valor ingresado después de la cadena (`num1`), el segundo `%d` se sustituirá por el segundo entero y así sucesivamente

- Usaremos `%f` para valores float o double

- Usaremos `%x.yf` para float o double con formato 'x' dígitos 'y' decimales:

`System.out.printf("valor pi: %6.4f", Math.PI);`

// Salida "valor pi: 3,1416" con 6 dígitos y 4 decimales

- Más ejemplos:  
<http://www.java2s.com/Code/JavaAPI/java.lang/System.out.printf.htm>

# Argumentos para la línea de comandos

- Puedo ejecutar desde NetBeans un programa que espera argumentos en la línea de comandos:
  - Menu Run -> Set Project Configuration -> Customize  
Main Class: el nombre de la clase que quiero ejecutar  
Arguments: Escribo los argumentos
  - Ojo! He de ejecutar el proyecto (run ->Project) y no el fichero (Run -> File)

