

### Soluciones de ejercicios 1 y 2

```
U.println("Calcular velocidad de un móvil");
double d=U.readDouble("distancia(metros)?"),
      t=U.readDouble("tiempo(segundos)?");
U.println("velocidad="+ (d/1000)/(t/3600)+"km/hora");
alternativamente:
U.println("velocidad="+ 3.6*d/t + "km/hora");

U.println("ingresar nº alumnos de colegios:");
int a=U.readInt("públicos?"),
    b=U.readInt("subvencionados?"),
    c=U.readInt("particulares?");
double factor=100.0/(a+b+c);
U.println("públicos=" + a*factor + "% " +
        "subvencionados=" + b*factor + "% " +
        "particulares=" + c*factor + "%");
```

### Solución de ejercicio 3

```
//obtener nº
int n=U.readInt("Ingresar un nº de 2 dígitos?");
//calcular primer dígito
int d1=n/10; //ej:73/10=7
//calcular segundo dígito
int d2=n-d1*10; //ej:73-7*10=3
//mostrar resultado
U.println("nº invertido=" + (10*d2+d1) );
```

**Mejor** (muestra bien los terminados en 0)  
U.println("nº invertido=" + d2 + d1);

**equivalencia:**  
U.print("nºinvertido=");U.print(d2);U.println(d1);

### Solución de ejercicio 3 con operador % (resto o residuo de división entre enteros)

ejemplo: 7 % 3 entrega 1  
sintaxis: `exp1 % exp2` (ambas enteras)  
semántica: `exp1 - exp1/exp2*exp2`  
prioridad: misma que \* y /

```
//separar los dígitos
int d1=n/10, d2=n%10;
U.println("Resultado=" + d2 + d1);
```

#### o directamente:

```
U.print("Resultado=" + n%10 + n/10);
```

### Problema

Escribir las instrucciones principales de un programa que establezca el siguiente diálogo:

```
Perímetro y area de triangulo de lados a,b,c
a ? ____
b ? ____
c ? ____
Perímetro = nº
Area = nº
```

Nota. El área se calcula como  $\sqrt{s(s-a)(s-b)(s-c)}$   
con  $s=\frac{a+b+c}{2}$  (semi-perímetro)

### Programa

```
U.println("perímetro y area de "
+ " triangulo de lados a,b,c");
//obtener lados
double a=U.readDouble("a ? "),
      b=U.readDouble("b ? "),
      c=U.readDouble("c ? ");

//calcular y mostrar perimetro y area
U.println("Perímetro = " + (a+b+c));
double s=(a+b+c)/2; //semi-perimetro
U.println("Area = " +
    Math.sqrt(s*(s-a)*(s-b)*(s-c)));
o
    Math.pow(s*(s-a)*(s-b)*(s-c), 0.5));
```

### Uso (invocación) de métodos (funciones) predefinidas ejemplos

```
Math.sqrt(s*(s-a)*(s-b)*(s-c))
Math.pow(s*(s-a)*(s-b)*(s-c), 0.5)
```

#### sintaxis

Math.nombre(argumentos)  
¿argumentos? cero o más expresiones aritméticas (separadas por comas)

#### semántica

1º evaluar argumentos. Ejemplo: `s*(s-a)*(s-b)*(s-c)`  
2º evaluar función (método) con/en valores de argumentos  
3º recibir resultado de la función (en el lugar o punto de invocación)

## Funciones predefinidas en el lenguaje Java (métodos de clase predefinida Math)

función	significado	ejemplo	resultado
<b>sqrt(x)</b>	$\sqrt{x}$ , $x \geq 0$	sqrt(4.0)	2.0
<b>pow(x,y)</b>	$x^y$	pow(2.0,3.0)	8.0
<b>exp(x)</b>	$e^x$	exp(1.0)	2.71...
<b>log(x)</b>	log <sub>e</sub> x	log(Math.E)	1.0
<b>sin(x)</b>	seno de angulo x	sin(Math.PI)	0.0
<b>cos(x)</b>	coseno de x	cos(Math.PI)	-1.0
<b>tan(x)</b>	tangente de x	tan(Math.PI)	0.0
<b>asin(x)</b>	arco-seno de x	asin(1.0)	PI/2
<b>acos(x)</b>	arco-coseno de x	acos(1.0)	0.0
<b>atan(x)</b>	arco-tangente de x	atan(0.0)	0.0

- todas las funciones reciben y entregan un double
- un argumento double admite un int (int es assignable a double)

## Funciones predefinidas en el lenguaje Java (métodos de clase Math)

función	significado	arg	result	ejemplo	result
<b>abs(x)</b>	x	int double	int double	abs(-3) abs(-3.0)	3 3.0
<b>max(x,y)</b>	mayor entre x e y	int double	int double	max(4, 6) max(4.1, 6.5)	6 6.5
<b>min(x,y)</b>	menor entre x e y	int double	int double	min(4, 6) min(4.1, 6.5)	4 4.1
<b>random()</b>	Nº al azar en [0,1[		double	random( )	0.x....

**Prob.** Mostrar el mayor entre 3 enteros entre 1 y 100 generados al azar  
 Números = 26 72 37  
 Mayor = 72

```
//generar 3 nª reales en rango [0,1[: 0.ddd..
double a=Math.random(),
       b=Math.random(),
       c=Math.random();
//generar y mostrar 3 enteros en rango [1,100]
int i=1+(int)(100*a),
    j=1+(int)(100*b),
    k=1+(int)(100*c);
U.println("Numeros= " + i + " " + j + " " + k);
//determinar y mostrar el mayor
int l=Math.max(i,j), //mayor entre i y j
    m=Math.max(l,k); //mayor entre l y k
U.println("Mayor= " + m);
```

```
int i=1+(int)(100*a);
```

- multiplica por 100 n° 0.d<sub>1</sub>d<sub>2</sub>d<sub>3</sub>... Resultado n°=d<sub>1</sub>d<sub>2</sub>d<sub>3</sub>...  
ej: 57.8...
- convierte d<sub>1</sub>d<sub>2</sub>d<sub>3</sub>... a entero, truncando decimales.  
ej: 57.8.. a 57
- suma 1 a entero y asigna  
porque no se permite asignar un real a un entero: int i=1+100\*a;

### coerción de tipos (casting)

**sintaxis:** (tipo)(expresión)

### semántica

1. evaluar expresión
2. convertir resultado al tipo indicado
3. entregar resultado

## Solución 2. Con Funciones definidas por el programador

```
class Programa
{
  static public void main(String[]args){
    int i=azar(1,100), j=azar(1,100), k=azar(1,100);
    U.println("Numeros="+i+" "+j+" "+k);
    U.println("Mayor= " + mayor(i,j,k));
  }
  static public int azar(int x,int y){
    return x + (int)(Math.random()*(y-x+1));
  }
  static public int mayor(int x,int y,int z){
    int m=Math.max(x,y);
    return Math.max(m,z);
  }
}
```

## Funciones definidas por el programador

### sintaxis

```
static public tipo-resultado nombre(parámetros)
{
  instrucciones
  return expresión;
}
¿parámetros? tipo nombre, ...
```

### semántica

- 1º copiar (asignar) argumentos de llamada en parámetros
- 2º ejecutar las instrucciones (si es que existen)
- 3º devolver valor de la expresión (del tipo-resultado) en el punto de invocación (uso)

**Problema.** Escribir los siguientes métodos:

```
class Programa
{
    //minutos(x): convierte x (nº HHMM) a minutos
    static public int minutos(int x){
        ...
    }
    //diferencia en minutos entre 2 instantes de tiempo
    static public void main(String[]args)
    throws IOException{
        ...
    }
}
```

El método main debe usar la función minutos y establecer el diálogo del sge ejemplo:

```
Tiempo 1 (HHMM)? 1045
Tiempo 2 (HHMM)? 2115
Diferencia=630 minutos
```

```
class Programa
{
    //minutos(x): convierte x (nº HHMM) a minutos
    static public int minutos(int x)
    {
        return x/100*60 + x%100;
    }
    //diferencia en minutos entre 2 instantes de tpo
    static public void main(String[]args)
    throws IOException
    {
        int t1=U.readInt("Tiempo 1 (HHMM)?");
        int t2=U.readInt("Tiempo 2 (HHMM)?");
        int t=Math.abs( minutos(t2)- minutos(t1) );
        U.println("Diferencia="+t+" minutos");
    }
}
```

### Funciones en clases independientes

```
class U{ //en archivo U.java
...
static public int azar(int x,int y){
    return x + (int)(Math.random()*(y-x+1));
}
static public int mayor(int x,int y,int z){
    return Math.max(Math.max(x,y),z);
}
static public int menor(int x,int y,int z){
    return Math.min(Math.min(x,y),z);
}
}
```

### Uso de funciones de otra clase

```
class MayorDe3{ //en archivo MayorDe3.java
{
    static public void main(String[]args)
    {
        int i=U.azar(1,100),
        k=U.azar(1,100),
        k=U.azar(1,100);
        U.println("Numeros="+i+" "+j+" "+k);

        U.println("Mayor=" + U.mayor(i,j,k));
    }
}
Sintaxis: nombreClase.nombreFunción(argumentos)
```

```
class Math{ //Clase predefinida Math
static public //antes de cada función
double sqrt(double x){...}
double pow(double x,double y){...}
double exp(double x){...}
double log(double x){...}
double sin/cos/tan(double x){...} //en radianes
double asin/acos/atan(double x){...}
double abs(double x){...}
int abs(int x){...}
double max(double x,double y){...}
int max(int x,int y){...}
double min(double x,double y){...}
int min(int x,int y){...}
double random(){...}
static public final double PI=3.14...; //cte
static public final double E=2.71...; //cte
}
```

### Tarea

1. nº de 3 dígitos? 123  
resultado=321
2. **Cajero automático**  
¿Cuánto dinero necesita? 38000  
Retire los siguientes billetes:  
3 de \$10.000  
1 de \$5.000  
1 de \$2.000  
1 de \$1.000
3. int hhmm(int minutos)  
//Ej:hhmm(630)=1030
4. int aaaammdd(int ddmmaaaa)  
//Ej: aaaammdd(13032008)=20080313
5. Números al azar=45 23 67  
Ordenados=23 45 67
6. static public int mayor(int x,int y)  
sin usar Math.max