



# Unión Europea

Fondo Social Europeo

*El FSE invierte en tu futuro*

## UD 1 CREACIÓN DE PROCESOS (PSP)

Repaso Java

INSTITUT



GENERALITAT  
VALENCIANA

Conselleria d'Educació,  
Investigació, Cultura i Esport



GOBIERNO  
DE ESPAÑA

MINISTERIO  
DE EDUCACIÓN, CULTURA  
Y DEPORTE

## Continguts

<b>1</b>	<b>Arrays bidimensionales</b>	<b>3</b>
<b>2</b>	<b>Colecciones</b>	<b>3</b>
<b>3</b>	<b>Excepciones</b>	<b>4</b>

## 1 Arrays bidimensionales

1. Escribe un programa que, dada una posición en un tablero de ajedrez, nos diga a qué casillas podría saltar un alfil que se encuentra en esa posición. Como se indica en la figura, el alfil se mueve siempre en diagonal. El tablero cuenta con 64 casillas. Las columnas se indican con las letras de la “a” a la “h” y las filas se indican del 1 al 8.

- Ejemplo:

- Introduzca la posición del alfil: d5
- El alfil puede moverse a las siguientes posiciones: h1 a2 g2 b3 e3 c4 e4 c6 e6 b7 f7 a8 g8

2. Define un array de números enteros de 3 filas por 6 columnas con nombre “num” y asigna los valores según la siguiente tabla:

	Column 0	Column 1	Column 2	Column 3	Column 4	Column 5
Fila 0	0	30	2	0	0	5
Fila 1	75	0	0	0	0	0
Fila 2	0	0	-2	9	0	11

Muestra el contenido de todos los elementos del array dispuestos en forma de tabla como se muestra en la figura.

## 2 Colecciones

1. Crea un ArrayList con los nombres de 6 compañeros de clase. A continuación, muestra esos nombres por pantalla. Utiliza para ello un bucle **for** que recorra todo el **ArrayList** sin usar **ningún índice**.
2. Realiza un programa que introduzca valores aleatorios (entre 0 y 100) en un ArrayList y que luego calcule la suma, la media, el máximo y el mínimo de esos números. El tamaño de la lista también será aleatorio y podrá oscilar entre 10 y 20 elementos ambos inclusive.
3. Introduce 10 palabras y te muestre por pantalla:

**Lista original:** [php, java, c, c++, c#, javascript, python, pascal, SQL, Ruby]

**Lista ordenada alfabéticamente:** [Ruby, SQL, c, c#, c++, java, javascript, pascal, php, python]

### 3 Excepciones

1. Crearemos un programa que vaya pidiendo números enteros en el momento que detecta un número no entero, nos muestre una excepción **try...catch** y nos lo vuelva a pedir.

```
Por favor, vaya introduciendo números enteros.  
N 1: 1  
N 2: 2  
N 3: 3.3  
El dato introducido no es correcto, debe ser un número entero.  
Por favor, inténtelo de nuevo.  
N 3:
```