



APRENDERAPROGRAMAR.COM

GENERAR NÚMEROS  
ALEATORIOS EN JAVA.  
CLASE RANDOM Y MÉTODO  
NEXTINT. EJERCICIO  
EJEMPLO RESUELTO.  
(CU00672B)

Sección: Cursos

Categoría: Curso “Aprender programación Java desde cero”

Fecha revisión: 2029

**Resumen:** Entrega nº72 curso Aprender programación Java desde cero.

Autor: Alex Rodríguez

## GENERAR NÚMEROS ALEATORIOS EN JAVA

Hay varias maneras de generar números aleatorios en Java. Entre ellas, **métodos previstos por las clases Math y Random**. La terminología puede resultarnos inicialmente confusa, pues la clase Math dispone de un método denominado random(), mientras que Random es a su vez el nombre de otra clase.



La generación de números aleatorios por ordenador no es tan sencilla como pueda parecer debido a que la operación de los ordenadores es determinística: se apoya en el hecho de que todo cálculo es predecible y repetible. Por ello conviene tener presente que sería más apropiado usar el término “pseudoaleatorio” para referirnos a los números aparentemente aleatorios que podemos obtener de un ordenador.

Vamos a ver un extracto de la documentación de la clase Random (ten en cuenta que pueden existir pequeñas diferencias según la versión de Java que estés utilizando). Para utilizarla habremos de indicar en cabecera `import java.util.Random;` o de forma más general `import java.util.*;`. Para generar números (pseudo) aleatorios vamos a tener que crear un objeto de tipo Random y luego invocar un método sobre ese objeto que nos devuelva el (pseudo) aleatorio. Veamos el extracto de documentación:

`java.util`

### Class Random

[java.lang.Object](#)

└ `java.util.Random`

Un objeto de esta clase se usa para generar una secuencia (stream) de números pseudoaleatorios a partir de un valor inicial o “semilla”. Si dos instancias de la clase Random son creadas con la misma “semilla”, y se ejecutan las mismas instrucciones para ambas instancias, se generarán y devolverán idénticas secuencias de números.

### Constructor Summary (se omiten algunos constructores)

[Random](#) ( )      Crea un generador de números aleatorios. Usando este constructor, en cada invocación Java se encarga de que la “semilla” para generación de aleatorios sea siempre distinta.

## Method Summary (sólo algunos métodos, otros se omiten)

boolean	<a href="#">nextBoolean</a> ( ) Devuelve un valor true o false de forma pseudoaleatoria. La probabilidad de obtener un valor u otro es de aproximadamente la mitad.
double	<a href="#">nextDouble</a> ( ) Devuelve un pseudoaleatorio de tipo double, uniformemente distribuido entre 0.0 y 1.0.
float	<a href="#">nextFloat</a> ( ) Devuelve un pseudoaleatorio de tipo float, uniformemente distribuido entre 0.0 y 1.0.
int	<a href="#">nextInt</a> ( ) Devuelve un pseudoaleatorio de tipo int con valores entre 0 y $2^{32}$ producidos con aproximadamente la misma probabilidad.
int	<a href="#">nextInt</a> ( int n ) Devuelve un pseudoaleatorio de tipo int comprendido entre cero (incluido) y el valor especificado (excluido).
long	<a href="#">nextLong</a> ( ) Devuelve un pseudoaleatorio de tipo long.
Otros	<b>Existen más métodos (consultar la documentación del API de Java para ampliar información)</b>

Vamos a escribir un programa que utiliza la clase Random. Además también utiliza el print en lugar del println y el carácter de escape \n para forzar cambio de línea y retorno de carro. Utilizaremos la clase EntradaDeTeclado que ya hemos escrito y usado en programas anteriores, aunque con una pequeña variación en el constructor. Escribe el código de las siguientes clases.

```
import java.util.Scanner; //Importación de la clase Scanner desde la biblioteca Java
public class EntradaDeTeclado { // Definimos la clase EntradaDeTeclado aprenderaprogramar.com
    private String entradaTeclado; //Variable de instancia (campo) del método

    //Constructor
    public EntradaDeTeclado () {
        entradaTeclado=""; } //Cierre del constructor

    public void pedirEntrada () { //Método de la clase
        Scanner entradaEscaner = new Scanner (System.in);
        entradaTeclado = entradaEscaner.nextLine ();
    } //Cierre del método pedirEntrada

    public String getEntrada () {
        return entradaTeclado;
    } //Cierre del método getEntrada

} //Cierre de la clase
```

```

import java.util.ArrayList; import java.util.Random; //Ejemplo aprenderaprogramar.com

//Esta clase define objetos que contienen tantos enteros aleatorios entre 0 y 1000 como se definan
public class SerieDeAleatorios {
    //Campos de la clase
    private ArrayList<Integer> serieAleatoria;

    //Constructor
    public SerieDeAleatorios (int numeroltems) {
        serieAleatoria = new ArrayList<Integer> ();
        //Inicializamos el ArrayList lleno de ceros
        for (int i=0; i<numeroltems; i++) { serieAleatoria.add(0); }
        System.out.println ("Serie inicializada. El número de elementos en la serie es: " + getNumeroltems() );
    } //Cierre del constructor

    public int getNumeroltems() { return serieAleatoria.size(); } //Cierre del método

    public void generarSerieDeAleatorios () {
        Random numAleatorio;
        numAleatorio = new Random ();
        for (int i=0; i < serieAleatoria.size(); i++) {
            serieAleatoria.set(i, numAleatorio.nextInt(1000) );
        }
        System.out.print ("Serie generada! ");
    } //Cierre del método

    public void mostrarSerie() {
        System.out.print ("Procedemos a mostrar la serie: ");
        for (Integer tmpObjeto : serieAleatoria) { //Uso de for each
            System.out.print (" " + tmpObjeto.toString() ); } //Cierre del for extendido
        } //Cierre del método
    } //Cierre de la clase

// Programa Test de obtención de números pseudoaleatorios aprenderaprogramar.com
public class TestPseudoAleatorios {
    public static void main (String [ ] Args) {
        Integer tmpInteger = 0;
        EntradaDeTeclado entradaMain = new EntradaDeTeclado();
        char tecla = 'S';
        while (tecla == 'S') {
            System.out.print ("Por favor introduzca el número de elementos en la serie de números aleatorios: ");
            entradaMain.pedirEntrada();
            tmpInteger = tmpInteger.valueOf (entradaMain.getEntrada() );
            SerieDeAleatorios serieDePrueba = new SerieDeAleatorios ( (int) tmpInteger);
            serieDePrueba.generarSerieDeAleatorios();
            serieDePrueba.mostrarSerie();
            tecla = ' ';
            System.out.println ("\n¿Generar otra serie (S/N):");
            while (tecla != 'n' && tecla != 'N' && tecla != 's' && tecla != 'S') {
                entradaMain.pedirEntrada();
                //Ojo tenemos que comparar usando equals porque los string son objetos!!!!
                if (entradaMain.getEntrada().equals("n") || entradaMain.getEntrada().equals("N")) { tecla = 'N'; }
                else if (entradaMain.getEntrada().equals("s") || entradaMain.getEntrada().equals("S")) { tecla = 'S'; }
            } //Cierre del while interior
        } //Cierre del while exterior
        System.out.println ("Gracias por utilizar el programa");
    } } //Cierre del main y de la clase

```

El resultado del programa será similar a este:

```
Por favor introduzca el número de elementos en la serie de números aleatorios: 3
Serie inicializada. El número de elementos en la serie es: 3
Serie generada! Procedemos a mostrar la serie: 745 777 24
¿Generar otra serie (S/N):
S
Por favor introduzca el número de elementos en la serie de números aleatorios: 7
Serie inicializada. El número de elementos en la serie es: 7
Serie generada! Procedemos a mostrar la serie: 314 520 402 944 71 248 839
¿Generar otra serie (S/N):
N
Gracias por utilizar el programa
```

## EJERCICIO

Crea un programa Java que permita “jugar a adivinar un número” como se expone a continuación. El programa debe iniciarse indicando “Se ha generado un número aleatorio entero entre 1 y 100, intente adivinarlo”. El usuario introducirá un número y si el número aleatorio generado por el ordenador es menor deberá indicarse “No has acertado: el número es menor. Prueba otra vez”. Si el usuario introduce un número menor que el número aleatorio deberá indicarse “No has acertado: el número es mayor”. El programa terminará cuando el usuario introduzca el número aleatorio que había escogido el ordenador. Puedes comprobar si tu código es correcto consultando en los foros [aprenderaprogramar.com](http://aprenderaprogramar.com).

**Próxima entrega:** CU00673B

**Acceso al curso completo** en [aprenderaprogramar.com](http://aprenderaprogramar.com) --> Cursos, o en la dirección siguiente:

[http://www.aprenderaprogramar.com/index.php?option=com\\_content&view=category&id=68&Itemid=188](http://www.aprenderaprogramar.com/index.php?option=com_content&view=category&id=68&Itemid=188)