

# CURSO DE FUNDAMENTOS DE JAVA

## OBJETOS EN JAVA



Por el experto: Ing. Ubaldo Acosta



**CURSO DE FUNDAMENTOS DE JAVA**

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

Hola, te saluda Ubaldo Acosta. Bienvenida o bienvenido nuevamente. Espero que estés listo para comenzar con esta lección.

Vamos a estudiar el tema de Clases y Objetos, esto es la base de la programación orientada a objetos.

¿Estás listo? Ok, ¡Vamos!



Un objeto es la instancia de una clase, es decir que a partir de nuestra plantilla, es posible crear objetos en concreto con los cuales podremos trabajar, es decir, cambiar sus valores y llamar a sus métodos.

Más adelante veremos que hay casos en los que podremos trabajar directamente con la clase, a esto se le llama contexto estático, pero de momento estaremos estudiando el contexto dinámico, es decir, necesitamos crear una instancia de una clase para poder trabajar con ellas.

Como podemos observar en la figura, tenemos la clase Persona, y a partir de esta plantilla podemos crear objetos concretos para poder trabajar con sus características y atributos. Por ejemplo hemos creado dos objetos de tipo Persona, y cada uno tiene características similares pero con valores distintos.

Las características son por ejemplo: Nombre, Apellido, Color, Género, escolaridad, etc.

Y cada uno de los objetos se les asigna estas características, de esta manera la clase nos sirve sólo para indicar que características tendrán los objetos que vamos a crear. Veamos ahora la sintaxis para crear un objeto.

# CREACIÓN DE UN OBJETOS EN JAVA

## Creación de un objeto Persona:

```
public class PersonaPrueba {

    public static void main(String args[]){
        //Creacion de un objeto
        Persona p1 = new Persona();

        //Modificar valores
        p1.nombre = "Armando";
        p1.apellidoPaterno = "Esparza";
        p1.apellidoMaterno = "Lara";
    }
}
```

## CURSO DE FUNDAMENTOS DE JAVA

www.globalmentoring.com.mx

Para definir un objeto vamos a utilizar la palabra reservada new. Esta palabra en Java significa en términos simples que vamos a crear una nueva variable del tipo Persona. Para crear un objeto vamos a utilizar una línea de código como sigue:

```
Persona p1 = new Persona();
```

Después de que esta línea se ejecuta, la variable p1 podrá acceder a un objeto de tipo Persona. Cada vez que se manda llamar la palabra new se creará un nuevo objeto de tipo Persona, y contendrá sus propios valores. Por ejemplo:

```
Persona p2 = new Persona();
```

Esta nueva variable p2, apunta a un nuevo objeto de tipo Persona, y por lo tanto contendrá valores distintos a la variable p1. Así, cada nuevo objeto contendrá valores distintos en los atributos de nombre, apellidoPaterno y apellidoMaterno. La sintaxis new Persona() hace referencia al nombre de la clase de la cual queremos crear una nueva instancia, es decir, un objeto. Además, estamos haciendo uso del concepto de constructor de la Clase, este concepto lo estudiaremos más adelante, aquí solo debemos notar que para llamar al constructor de la clase se pone el nombre de la clase y posteriormente se utilizan paréntesis, en este caso un constructor sin argumentos, pero veremos posteriormente que podemos enviar argumentos al momento de crear un objeto.

El nombre de la variable puede ser cualquiera que elijamos, pero no debe tener caracteres especiales, o espacios. Para acceder a los atributos de nuestras clases vamos a utilizar el operador . A través de este operador se liga el nombre del objeto con el nombre del atributo de la clase. Por ejemplo, para asignar el valor de Armando al atributo nombre del objeto p1, sería como sigue:

```
p1.nombre = "Armando";
```

En lecciones posteriores veremos en términos más técnicos lo que hace la palabra new, pero para esta lección es suficiente con saber que nos permite crear nuevos objetos.

## MÁS DE LA PALABRA NEW

### Sentencias

### Efecto en la Memoria

```
//Declaración de una variable
Persona p1;
```



```
//Creacion de un objeto
Persona p2 = new Persona ();
```



Objeto Persona

## CURSO DE FUNDAMENTOS DE JAVA

www.globalmentoring.com.mx

La palabra `new` solicita espacio de memoria de manera dinámica para poder almacenar un objeto. Con la sintaxis generalmente como sigue:

```
variableQueAlmacenaReferencia = new TipoClaseAInstanciar();
```

En la figura podemos observar que en primer lugar declaramos una variable de tipo de la clase `Persona`, llamada `p1`. Podemos observar que el efecto que esto tiene en la memoria Java es que se crea espacio para esta variable llamada `p1`. Pero esta variable aún no tiene ningún valor asignado, ya que sólo se declaró la variable.

En la segunda sentencia, podemos observar, que además de declarar una variable de tipo `Persona` llamada `p2`, se le asigna la referencia de un objeto de tipo `Persona`. Para la creación de este objeto de tipo `Persona` se utiliza la palabra `new`, seguida del nombre de la clase de la cual queremos crear el objeto y finalmente los paréntesis es el tipo de constructor que vamos a utilizar para la creación de este objeto.

El concepto de constructor lo estudiaremos a detalle posteriormente, pero solo pensemos en el constructor como la forma en que queremos crear nuestro objeto de manera inicial, podemos crear por ejemplo un objeto de tipo `Persona` ya con valores como son `nombre`, `apellido`, etc, o como en este caso, podemos simplemente crear el objeto de tipo `Persona`, sin proporcionar los valores de `nombre` y `apellido` y posteriormente podremos cambiar y/o asignar estos valores a los atributos de nuestro objeto de tipo `Persona`.

Debemos aclarar que los tipos primitivos como `int`, `long`, etc, no son objetos, por lo tanto para declarar variables de estos tipos no se necesita utilizar la palabra reservada `new`.

Recordemos que la palabra `new` lo que hace internamente es solicitar espacio de memoria para la creación de un nuevo objeto, tantos objetos como necesitemos, sin embargo, la memoria es finita, y por lo tanto podríamos recibir un error al momento de crear demasiados objetos en tiempo de ejecución de nuestro programa. En temas posteriores veremos el manejo de errores y como tratar los más comunes, como la falta de memoria en Java.

En conclusión, una clase define un nuevo tipo, y un objeto es una instancia de este nuevo tipo que ya se ha creado previamente. Podemos crear tantos objetos del tipo creado como necesitemos, esto con ayuda de la palabra `new`.

## USO DE LA CLASE STRING

**NO ES UN  
TIPO  
PRIMITIVO**

**TRATAMIENTO  
ESPECIAL**



**NO SE  
NECESITA LA  
PALABRA NEW**

Ej. `String saludo = "Hola Mundo";`

**CURSO DE FUNDAMENTOS DE JAVA**

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

Si recordamos en lecciones previas, mencionamos que el tipo String es una clase de Java, pero con un tratamiento especial. Debido a que este tipo de datos se utiliza en repetidas ocasiones en nuestros programas, es posible omitir el uso de la palabra new, y por tanto simplificar la creación de una variable de tipo String.

Entonces podemos crear un objeto de tipo String de la manera clásica:

```
String nombre = new String("Armando");
```

Pero lo utilizaremos más comúnmente sin el uso de la palabra new, este es el método simplificado:

```
String nombre = "Armando";
```

Ambas opciones son válidas, pero utilizaremos más la segunda opción debido a que se simplifica el código. Sólo no hay que olvidar que el tipo String es una clase y no un tipo primitivo. Este tipo de dato es la única excepción a la regla para crear objetos utilizando el operador new. Todas las demás clases deberán utilizar el operador new para poder crear un objeto.

Empezaremos a utilizar este tipo en nuestros siguientes ejemplos, y así familiarizarnos rápidamente con el uso del tipo String en Java.

## EJERCICIOS CURSO FUNDAMENTOS DE JAVA

- **ABRIR LOS ARCHIVOS DE EJERCICIOS EN PDF.**
- **EJERCICIO:** Ejercicio Creación de Objetos en Java.

**CURSO DE FUNDAMENTOS DE JAVA**

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

# CURSO ONLINE

# FUNDAMENTOS DE JAVA

Por: Ing. Ubaldo Acosta



**CURSO DE FUNDAMENTOS DE JAVA**

[www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

En Global Mentoring promovemos la Pasión por la Tecnología Java. Te invitamos a visitar nuestro sitio Web donde encontrarás cursos Java Online desde Niveles Básicos, Intermedios y Avanzados, y así te conviertas en un experto programador Java.

Además agregamos nuevos cursos para que continúes con tu preparación como programador Java profesional. A continuación te presentamos nuestro listado de cursos:

- ✔ Lógica de Programación
- ✔ Fundamentos de Java
- ✔ Programación con Java
- ✔ Java con JDBC
- ✔ HTML, CSS y JavaScript
- ✔ Servlets y JSP's
- ✔ Struts Framework
- ✔ Hibernate Framework
- ✔ Spring Framework
- ✔ JavaServer Faces
- ✔ Java EE (EJB, JPA y Web Services)
- ✔ JBoss Administration
- ✔ Android con Java
- ✔ HTML5 y CSS3

#### Datos de Contacto:

Sitio Web: [www.globalmentoring.com.mx](http://www.globalmentoring.com.mx)

Email: [informes@globalmentoring.com.mx](mailto:informes@globalmentoring.com.mx)

