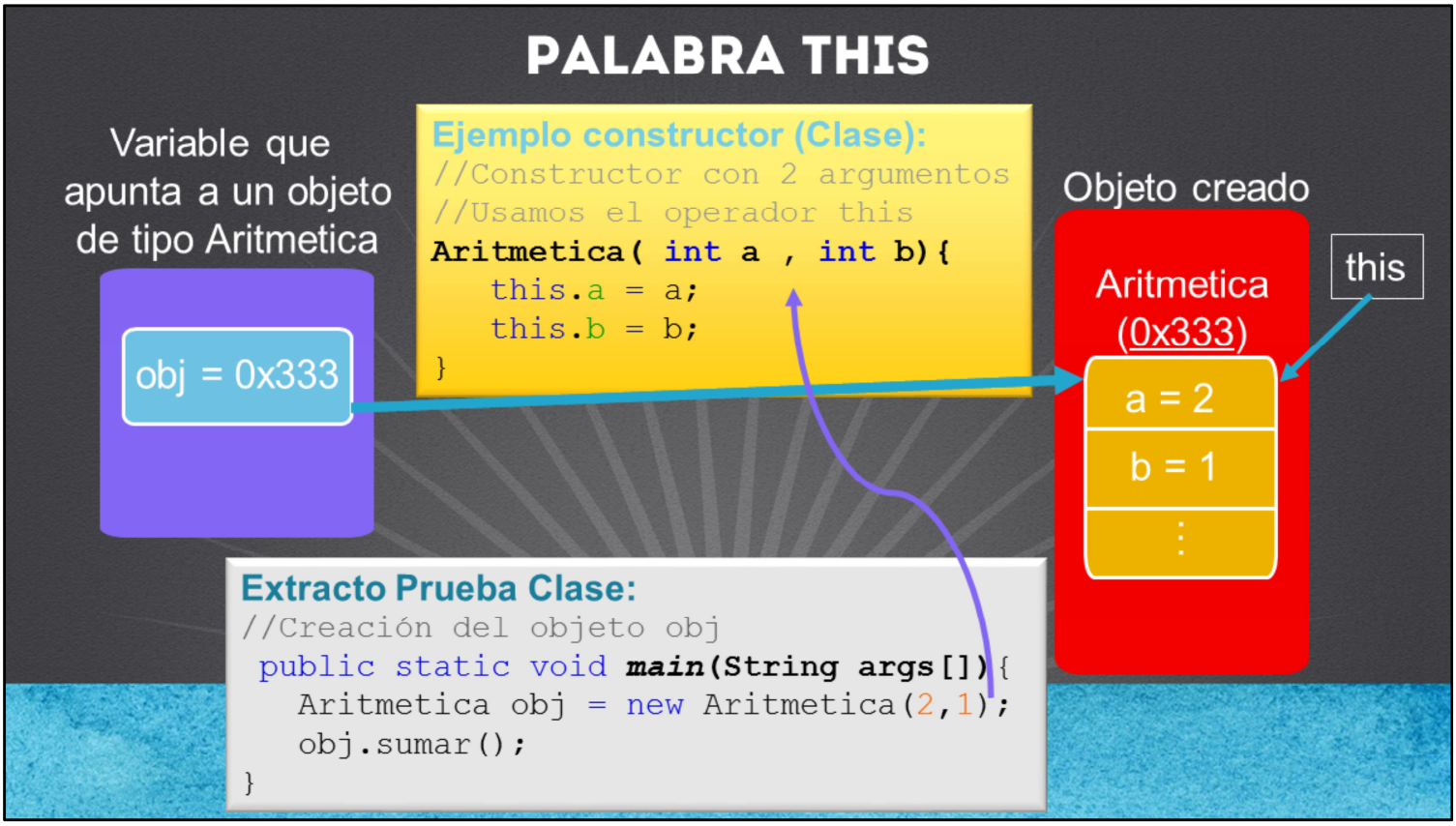
Hola, te saluda nuevamente Ubaldo Acosta. Espero que estés listo para comenzar con esta lección.

Vamos a estudiar el tema de métodos en Java.

¿Estás listo? ¡Vamos!



En ocasiones un método necesita hacer referencia al objeto con el que estamos trabajando actualmente. Para esta tarea Java agregó la palabra reservada this. La palabra this es un operador el cual nos permite acceder al objeto actual (la clase con la cual estamos trabajando), y nos servirá, entre otras cosas, para acceder a los atributos o métodos de una clase. Con esto podemos hacer una diferencia entre los argumentos recibidos en un método y los atributos de una clase.

Como podemos observar en la lámina, tenemos un código que tiene un atributo llamado a y b, y también el Constructor de la clase recibe dos argumentos llamados a y b. Para hacer diferencia entre estas dos variables (atributos de la clase y los argumentos recibidos en el método) podemos utilizar la palabra this como sigue:

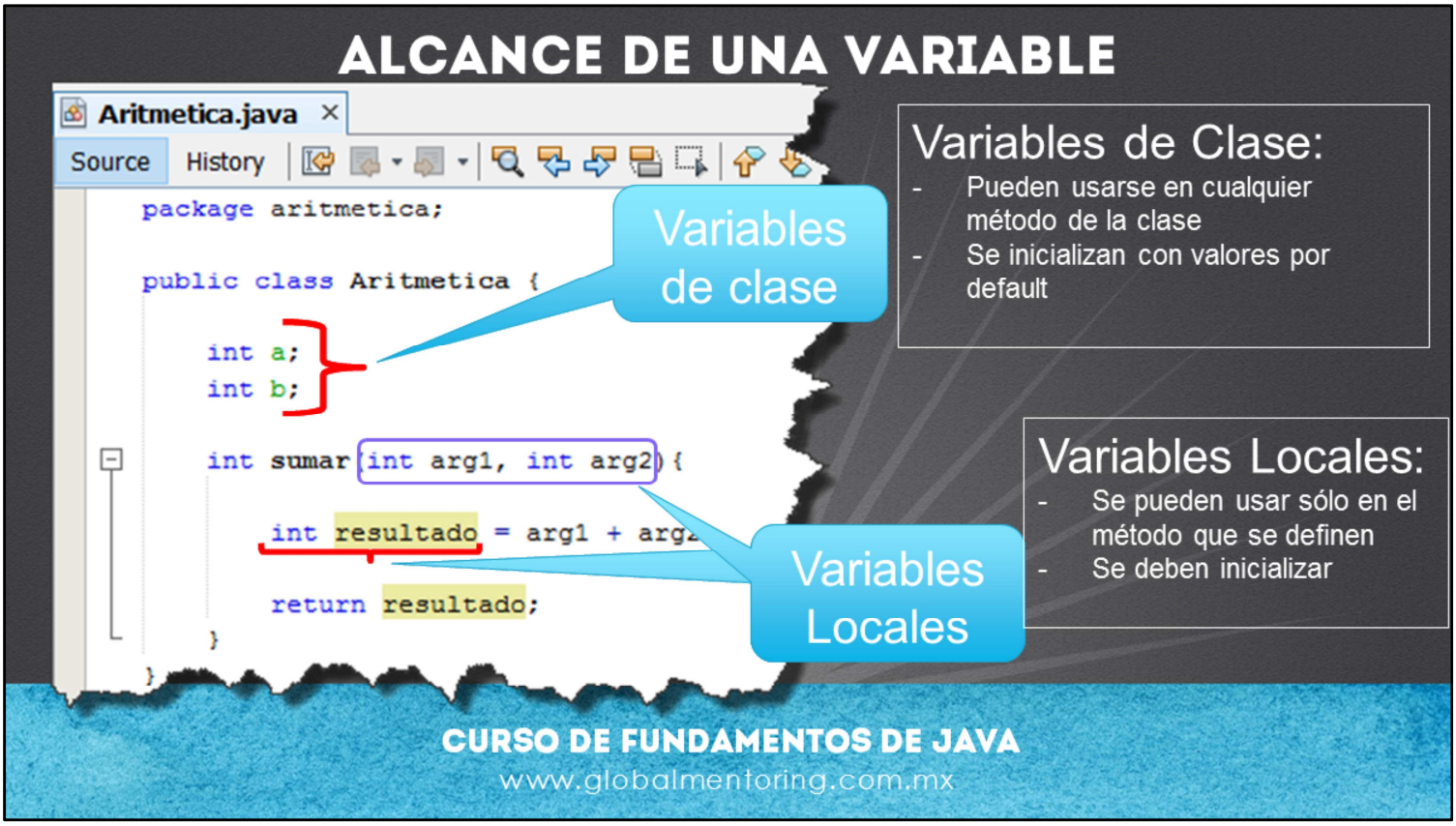
```
//Constructor con 2 argumentos. Usamos el operador this
Aritmetica( int a , int b){
  this.a = a;
  this.b = b;
}
```

Como podemos observar, para hacer diferencia entre los valores recibidos en el Constructor y los atributos de la clase, podemos utilizar el operador this. Esto es solo un ejemplo del uso del operador this, pero básicamente nos permite acceder a los atributos y métodos del objeto actual con el cual estamos trabajando.

Esto puede ser un confuso en un inicio, ya que estamos trabajando con el código de nuestra clase, es decir, la plantilla, y este código no se ejecutará hasta que hayamos creado un objeto y mandemos llamar este constructor, será en este momento cuando enviemos los argumentos y el constructor los reciba y los procese. Así que debemos acostumbrarnos a pensar en dos momentos, la creación de nuestras clases o plantillas, y la creación de nuestros objetos, que será cuando realmente se ejecute el código que hemos programado en nuestras clases o plantillas.

Aunque el uso del operador this en ocasiones parecerá redundante, es un buena práctica el usarlo para hacer referencia a los atributos de la clase en la que estamos trabajando, ya que al leer nuestro código rápidamente reconoceremos qué variables son atributos de una clase y cuales no lo son.

Dentro de los constructores o métodos de una clase, el operador this hará siempre referencia al objeto que fue invocado. Si observamos el código, tanto el argumento recibido como el atributo de la clase se llaman exactamente igual, por lo tanto toma prioridad el argumento sobre el atributo de la clase, a esto se le conoce como ocultamiento del atributo de la clase. Y para resolver este problema basta con utilizar el operador this antes de la variable, tal como si accediéramos al atributo de una clase por medio del operador punto.



En Java tenemos distintos tipos de variables, como son variables de Clase y variables Locales.

Dependiendo de dónde se defina la variable será la duración de la misma, y a esto se le conoce como el Alcance de una Variable.

Si una variable la definimos como un atributo de una clase, esta variable se le conoce como variable de Clase, y existirá durante todo el tiempo que exista el objeto en memoria. Estas variables se inicializan con su valor por default de manera automática, por ejemplo un tipo entero se inicializa con el valor de 0, una variable tipo bool con el valor de false, y un tipo Object con el valor null.

Por otro lado, las variables locales son cualquier variable definida dentro de un método, incluyendo los argumentos que recibe una función. Estas variables tiene un tiempo de vida más corto, ya que se crean al momento de ejecutarse el método y se eliminan de la memoria apenas haya terminado la ejecución de dicho método. Además las variables locales es necesario inicializarlas con algún valor, de lo contrario el compilador marcará un error debido a la falta de inicialización de este tipo de variables.

Las variables locales, ocultan a las variables de clase, y si queremos utilizar las variables de clase en un método que ha definido variables locales con el mismo nombre, entonces debemos utilizar el prefijo this para poder acceder a las variables de clase en lugar de los atributos locales.


EJERCICIOS CURSO FUNDAMENTOS DE JAVA
ABRIR LOS ARCHIVOS DE EJERCICIOS EN PDF.
EJERCICIO: Ejercicio Creación del proyecto Artimetica_V3
CURSO DE FUNDAMENTOS DE JAVA
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:

- Fundamentos de Java
- 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
Email: informes@globalmentoring.com.mx