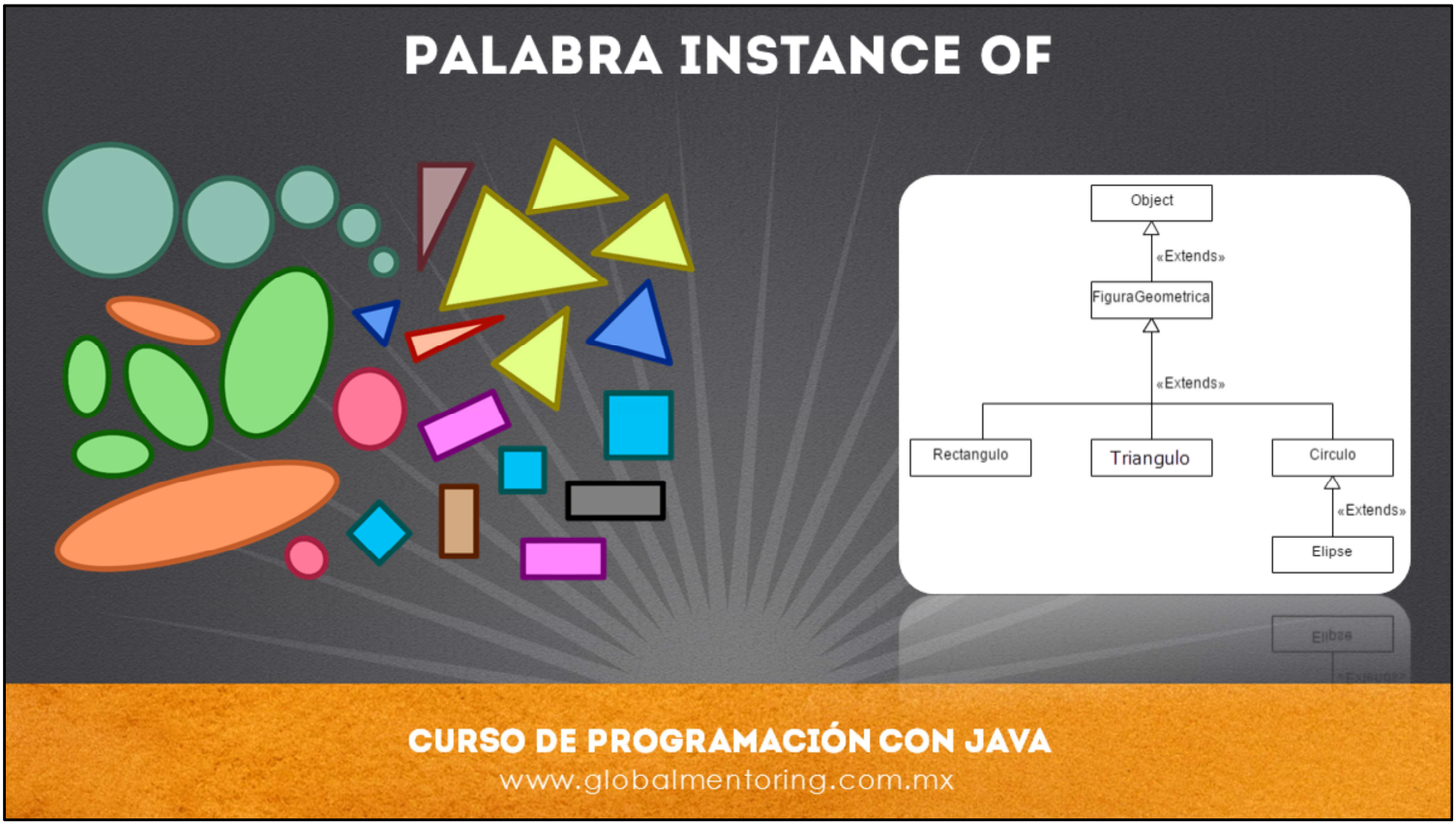


Hola, te saluda nuevamente Ubaldo Acosta. Espero que estés listo para comenzar con esta lección..

Vamos a estudiar el tema instance of en Java.

¿Estás listo? ¡Vamos!

© Todos los Derechos Reservados Global Mentoring



La palabra instace of básicamente responde a la pregunta si una clase tiene alguna relación con alguna otra clase que indiquemos.

Por ejemplo en la figura, podemos observar que todos los elementos podrían ser parte de una clase llamada FiguraGeométrica, es decir, esta sería la super clase o clase padre de todas las figuras geométricas mostradas. Y como subclases tendríamos la clase Circulo, Triangulo, Rectangulo, Elipses, etc.

Y a su vez cada una de estas clases serían a su vez un tipo de figura geométrica. Por lo tanto si preguntáramos si un circulo es de tipo figura geométrica la respuesta debería ser que si, de igual manera si hiciéramos esta pregunta de un rectángulo o de un triangulo. Para realizar esta pregunta existe precisamente la palabra reservada instanceof, la cual significa ¿es un tipo de este otro tipo indicado?

Vemos también un diagrama de clases con el cual podemos agregar una jerarquía de clases y así poder hacer la pregunta si una clase es de cierto tipo, y vemos que si hay una jerarquía de clases involucrada entonces lo lógico debería ser que sí son del tipo del cual heredan. Veamos a continuación un ejemplo basado en este diagrama de clases, de cómo hacer esta pregunta en código Java y para qué nos sirve realizar esta pregunta.

© Todos los Derechos Reservados Global Mentoring

USO DE INSTANCE OF EN JAVA

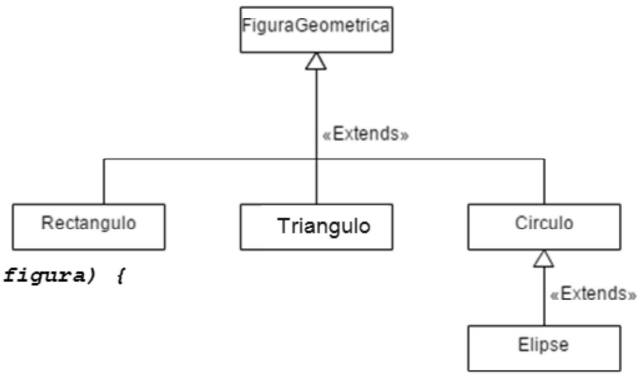
```
public class EjemploInstanceOf {

    public static void main(String[] args) {

        FiguraGeometrica figura;

        figura = new Elipse();
        determinaTipo(figura);
    }

    private static void determinaTipo(FiguraGeometrica figura) {
        if (figura instanceof Elipse) {
            //Procesa algo particular de la Elipse
            System.out.println("Es una Elipse");
        }
        if(figura instanceof Circulo){
            //Procesa algo particular del Circulo
            System.out.println("Es un Circulo");
        }
        if(figura instanceof FiguraGeometrica){
            //Procesa algo particular de la Figura Geometrica
            System.out.println("Es una Figura Geometrica");
        }
    }
}
```




```
Output - EjemploInstanceOf (run)
run:
Es una Elipse
Es un Circulo
Es una Figura Geometrica
```

El operador “instanceof” sirve para conocer si un objeto es una instancia de una clase determinada en tiempo de ejecución. Si corresponde con el tipo de dato regresa “true”, de lo contrario “false”.

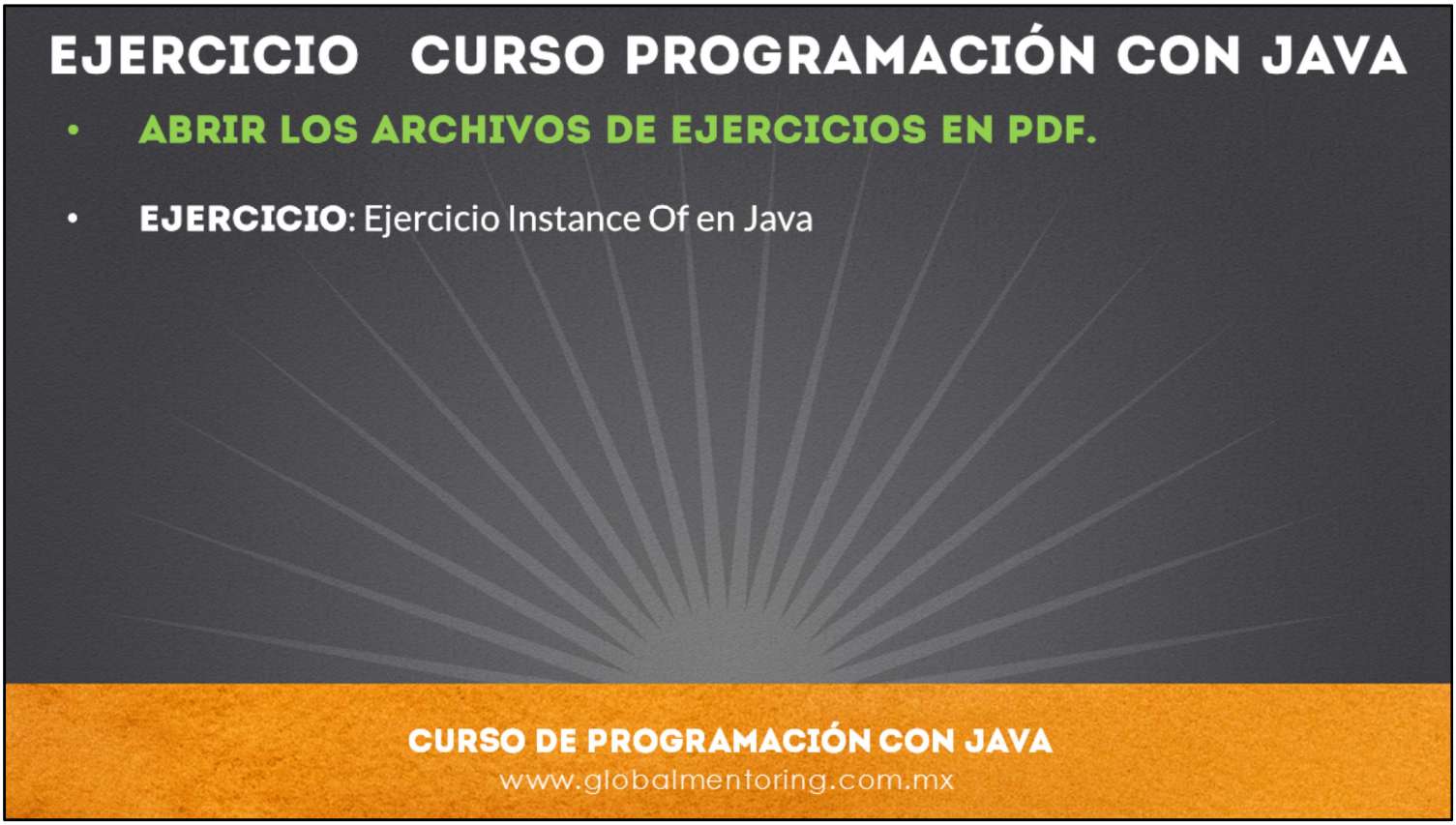
Como podemos observar en el código, el tipo Elipse debido a la jerarquía de clases que hemos definido responder que SI es del tipo para varios tipos, por ejemplo responde que si es de tipo Elipse, Circulo y FiguraGeometrica, incluso si preguntamos si es de tipo Object la respuesta será true, debido a que todas las clases de manera indirecta heredan en primer término de la clase Object.

Por ello, si vamos a hacer una revisión de varios tipos en la jerarquía de clases, se recomienda que se revise primero las clases de menor jerarquía y hasta el final las clases de mayor jerarquía, esto debido a que una clase podría dar como resultado “true” si es un tipo compatible, que puede promoverse a un tipo de dato. Este concepto de promoción lo veremos más adelante, pero básicamente si preguntamos primero por clases de mayor jerarquía nos regresa true, y podría ya no revisarse los casos de clases inferiores en la jerarquía de clases.

Si queremos que solo coincida con el tipo más interno, en lugar de utilizar la estructura if, podemos utilizar la estructura if-else, con la cual sólo entrará a un caso al estar comprobando si es de cierto tipo. Cabe mencionar que el operador instance of solo puede usarse en variables de tipo Object, pero no en tipos primitivos.

Ahora, para que nos sirve saber si un tipo es de cierto tipo en una jerarquía de clases. Y la respuesta es que dependiendo del tipo que sea una clase, podemos realizar ciertas operaciones por ser de cierto tipo o mandar a llamar algún método para cierto tipo, y esto sólo lo haríamos si responde como verdadero a un tipo de dato, de lo contrario no ejecutaríamos ese código. Así que es una forma de empezar a crear código más complejo dependiendo de la jerarquía de clases que tengamos. De alguna manera es como programar manualmente el polimorfismo, pero aquí no se limita a ejecutar un método de cierto tipo, sino podemos ejecutar cualquier código, o mandar a llamar varios métodos, etc, así que el operador instanceof nos permite saber el tipo de dato del objeto en tiempo de ejecución y así poder realizar alguna acción debido al tipo que hayamos detectado.

© Todos los Derechos Reservados Global Mentoring


EJERCICIO CURSO PROGRAMACIÓN CON JAVA
• ABRIR LOS ARCHIVOS DE EJERCICIOS EN PDF.
• EJERCICIO: Ejercicio Instance Of en Java
CURSO DE PROGRAMACIÓN CON 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:

- Programación con 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

© Todos los Derechos Reservados Global Mentoring