

¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.
 Ese apoyo que siempre quiso tener...

1. Desarrollo de componentes y proyectos a medida



2. Auditoría de código y recomendaciones de mejora

3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)
 Gestor de contenidos (Alfresco)
 Aplicaciones híbridas

Tareas programadas (Quartz)
 Gestor documental (Alfresco)
 Inversión de control (Spring)

Control de autenticación y
 acceso (Spring Security)
 UDDI
 Web Services
 Rest Services
 Social SSO
 SSO (Cas)

JPA-Hibernate, MyBatis
 Motor de búsqueda empresarial (Solr)
 ETL (Talend)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

BPM (jBPM o Bonita)
 Generación de informes (JasperReport)
 ESB (Open ESB)



[Home](#) | [Quienes Somos](#) | [Empleo](#) | [Foros](#) | [Tutoriales](#) | [Servicios Gratuitos](#) | [Contacte](#)

	<p>Tutorial desarrollado por: Roberto Canales Mora 2003-2005 Creador de AdictosAlTrabajo.com y</p> <p>Director General de Autentia S.L.</p> <p>Recuerda que me puedes contratar para echarte una mano:</p> <p>Desarrollo y arquitectura Java/J2EE Asesoramiento tecnológico Web Formación / consultoría integrados en tu proyecto</p> <p>No te cortes y contacta: 655 99 11 72 rcanales@autentia.com.</p>	
---	--	---

Descargar este documento en formato PDF [asserts.pdf](#)

Brindys Gedex

Gestión de Expedientes Jurídicos
Su PC puede ayudarle.
Descúbralo.

Curso Web J2EE

Curso Avanzado en Desarrollo
Web con J2EE

Inmopc, Programa + Web

Software de gestión inmobiliaria
Publique en su propio portal web

Mn Program

El único software de Gestión
Totalmente Gratis

Anuncios Goooooogle

Anunciarse en este sitio

Asserts en Java

En los lenguajes orientados a objeto, la gestión de los errores se realiza a través de excepciones.

Existen dos familias de expciones:

- Lo errores: Problemas graves que se pueden dar y que no merece la pena controlar (falta de memoria, error de comunicaciones, etc)
- Las excepciones en sí mismas que son predecibles y que de un modo razonable podemos (y debemos controlar).

Java esta en continua evolución y a partir de la versión 1.4 se ha introducido el concepto de asserts.

Un assert es una condición que se debe dar en un programa para que no se produzca un error.

La gracia que tiene es que son una mezcla de los dos tipos vistos anteriormente: Pueden suspender la ejecución de un programa pero no estamos obligados a controlarlas (declararlas y capturar su posible ejecución).

Además, en tiempo de ejecución podemos definir si queremos evaluarlas o no....

Vamos a introducir las en un programa y verificar su uso.

Creemos una clase normal

```
class misassert
{
    public static void main(String [] main)
    {
        int i = 10;

        assert i > 100;

        System.out.println("Este es mi ejemplo");
    }
}
```

Si compilamos, nos da un error:

```

C:\WINNT\system32\cmd.exe
C:\java>javac *.java
misassert.java:7: warning: as of release 1.4, assert is a keyword, and may not be
e used as an identifier
    assert i > 100;
    ^
misassert.java:7: ';' expected
    assert i > 100;
           ^
misassert.java:7: cannot resolve symbol
symbol  : class assert
location: class misassert
    assert i > 100;
    ^
misassert.java:7: i is already defined in main<java.lang.String[]>
    assert i > 100;
    ^
3 errors
1 warning
C:\java>_

```

Para poder compilar necesitamos especificar que trabajamos con la versión 1.4

```
javac -source 1.4 *.java
```

Ejecución

Si ahora ejecutamos nuestra clase, veremos que no tiene en cuenta el assert

Para que lo considere hay que utilizar un parámetro nuevo: **-ea**

```

C:\WINNT\system32\cmd.exe
C:\java>java misassert
Este es mi ejemplo

C:\java>java -ea misassert
Exception in thread "main" java.lang.AssertionError
    at misassert.main(misassert.java:7)

C:\java>_

```

Mensajes adicionales

Podemos poner un mensaje:

```
assert i > 100 : La variable i esta fuera de rango ;
```

```

C:\java>java -ea misassert
Exception in thread "main" java.lang.AssertionError: La variable i esta fuera de rango
    at misassert.main(misassert.java:7)

C:\java>

```

En mensaje es una expresión por lo que podría llamarse a una función... en nuestro caso podría mostrar los mensajes en función del idioma seleccionado

```

class misassert
{
    public static void main(String [] main)
    {
        int i = 10;

        assert i > 100 : mensajeError();

        System.out.println("Este es mi ejemplo");
    }

    public static String mensajeError()
    {
        // montaje de error internacionalizado
        return "La variable está fuera de rango";
    }
}

```

Si ahora decompilamos el código vamos que el compilador ha echo cosas por nosotros.... y nuestro código es más pesado

```
// Decompiled by DJ v3.5.5.77 Copyright 2003 Atanas Neshkov Date: 30/01/2004 19:30:08
// Home Page : http://members.fortunecity.com/neshkov/dj.html - Check often for new version!
// Decompiler options: packimports(3)
// Source File Name: misassert.java

import java.io.PrintStream;

class misassert
{
    misassert()
    {
    }

    public static void main(String args[])
    {
        byte byte0 = 10;
        if(!$assertionsDisabled && byte0 <= 100)
        {
            throw new AssertionError(mensajeError());
        } else
        {
            System.out.println("Este es mi ejemplo");
            return;
        }
    }

    public static String mensajeError()
    {
        return "La variable esta fuera de rango";
    }

    static Class _mthclass$(String s)
    {
        return Class.forName(s);
        ClassNotFoundException classnotfoundexception;
        classnotfoundexception;
        throw new NoClassDefFoundError(classnotfoundexception.getMessage());
    }

    static final boolean $assertionsDisabled; /* synthetic field */

    static
    {
        $assertionsDisabled = !(misassert.class).desiredAssertionStatus();
    }
}
```

Podemos utilizar en nuestro programa la misma sintaxis directamente::

```
throw new AssertionError(mensajeError());
```

Los asserts son una buena técnica para programas pre y post condiciones.... la programación por contrato....

[Sobre el Autor ..](#)

Si desea contratar formación, consultoria o desarrollo de piezas a medida puede contactar con

Formación en nuevas tecnologías

[Autentia S.L.](#) Somos expertos en:
J2EE, C++ , OOP, UML, Vignette, Creatividad ..
 y muchas otras cosas

Nuevo servicio de notificaciones

Si deseas que te enviemos un correo electrónico cuando introduzcamos nuevos tutoriales, inserta tu dirección de correo en el siguiente formulario.

Subscribirse a Novedades	
e-mail	<input type="text"/>
	<input type="button" value="Enviar"/>

Otros Tutoriales Recomendados ([También ver todos](#))

Nombre Corto	Descripción
Medida del Rendimiento en aplicaciones J2EE	Os mostramos como medir el rendimiento de vuestras aplicaciones Java J2E
Novedades en Java 1.5	Ya está disponible la versión Beta del J2SDK 1.5. Os mostramos algunas de las nuevas características introducidas en el lenguaje Java: Clases genéricas, enumeraciones, bucles simplificados, etc.
Mensajes multi-idioma en Java	Os mostramos como aprovechar las características multilenguaje de Java, usando las clases: Locate, ResourceBundle, MessageFormat, etc. Fundamental para un correcto diseño ...
Gestión errores en JSPs	Os mostramos como realizar ciertas labores intermedias en JSPs: Comentarios, gestión de errores, formateo de fechas y precompilación de ficheros
Escritura log con Fichero UDP y JMS	Os mostramos ejemplos para cuantificar el coste de escritura de Logs por pantalla, fichero, UDP y JMS (describiendo como configurar el entorno)
AspectJ, Programación con Aspectos	Os mostramos como configurar AspectJ (extensión Java para la programación basada en aspectos) y un pequeño ejemplo para medir la velocidad de una función sin alterar su código.

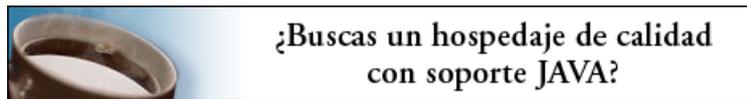
Nota: Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento.

Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores.

En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo.

Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador rcanales@adictosaltrabajo.com para su resolución.

[Patrocinados por enredados.com Hosting en Castellano con soporte Java/J2EE](#)



www.AdictosAlTrabajo.com Optimizado 800X600