

# ¿Qué ofrece Autentia Real Business Solutions S.L?

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

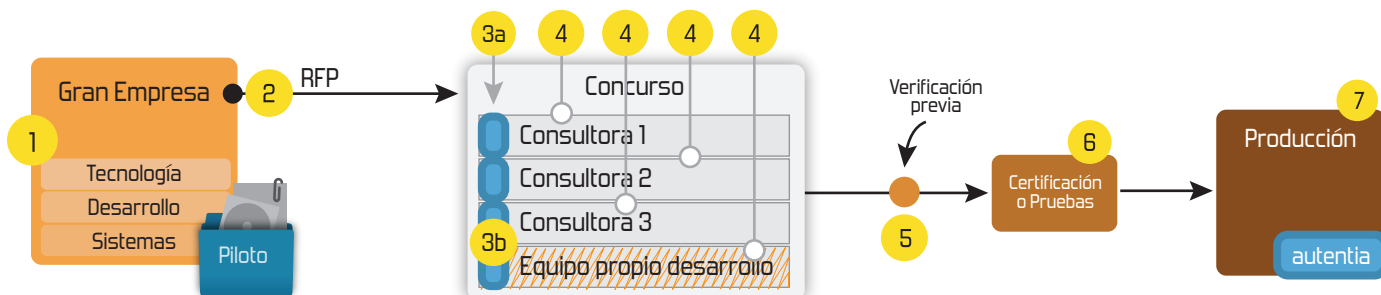
## 1. Desarrollo de componentes y proyectos a medida



## 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](#)

**Autor del tutorial:** *Cristhian Kirs Herrera Basurto*



▪ **Lugar de residencia:** Quito - Ecuador

*Cuento con experiencia en el área de desarrollo de software y en la docencia académica. Dentro de la construcción de software he manejado las etapas de: análisis, diseño, personalización e implementación de aplicaciones bajo ambientes Cliente / Servidor e Internet.* [Cristhian.Herrera@gmail.com](mailto:Cristhian.Herrera@gmail.com) / [cherrera@kruger.com.ec](mailto:cherrera@kruger.com.ec)

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

#### **Empleo malaga**

15 Hoteles en Málaga online. Con ofertas especiales!

#### **Trabajo en Londres U.K.**

Con o sin experiencia Alojamiento disponible

#### **Recibe ofertas de empleo**

Apúntate y recibe ofertas por email Consigue empleo por Internet

#### **InfoJobs.net**

La Bolsa de Empleo Líder en España Tenemos el profesional que buscas

Anuncios Goooooogle

### **Introducción a iReport**

La herramienta iReport es un constructor / diseñador de informes visual, poderoso, intuitivo y fácil de usar para **JasperReports** escrito en Java. Este instrumento permite que los usuarios corrijan visualmente informes complejos con cartas, imágenes, subinformes, etc. iReport está además integrado con JFreeChart, una de la biblioteca gráficas OpenSource más difundida para Java. Los datos para imprimir pueden ser recuperados por varios caminos incluso múltiples uniones JDBC, TableModels, JavaBeans, XML, etc.

### **Características de iReport**

La lista siguiente describe algunas de las características importantes de iReport:

- 100% escrito en JAVA y además OPENSOURCE y gratuito.
- Maneja el 98% de las etiquetas de JasperReports
- Permite diseñar con sus propias herramientas: rectángulos, líneas, elipses, campos de los textfields, cartas, subreports (subreportes).
- Soporta internacionalización nativamente.
- Browser de la estructura del documento.
- Recopilador y exportador integrados .
- Soporta JDBC.
- Soporta JavaBeans como orígenes de datos (éstos deben implementar la interface JRDataSource).
- Incluye Wizard's (asistentes) para crear automáticamente informes .
- Tiene asistentes para generar los subreportes
- Tiene asistentes para las plantillas.
- Facilidad de instalación.

### **Instalación de iReport**

El iReport viene como un archivo comprimido. Esto contiene los archivos de distribución principales (clases y fuente), algunas plantillas para el ayudante (wizard), todos los .jar requeridos adicionales. Esta herramienta trabaja de manera integrada con el jasperreport, por lo tanto no se requiere una instalación adicional de jasper.

El archivo puede descargarse desde <http://sourceforge.net/projects/ireport/>

### **Prerrequisitos**

Necesitará disponer de lo siguiente:

- Apache Ant
- IReport
- Jassperreport-0.x.x (usar la versión que viene incluida con iReport)

### **Instalación y configuración del ambiente de trabajo**

#### **Instalación de ANT**

- Descargar la versión **apache-ant-1.6.2-bin.zip** o superior.
- Descomprimir el archivo zip.

- Crear un directorio con el nombre ANT\_HOME o ANT. En el directorio raíz. C:\.
- Copiar el contenido de Ant en la carpeta creada.
- Configurar las variables de entorno:
  1. ANT\_HOME = C:\ANT\_HOME o ANT\_HOME = C:\ANT
  2. JAVA\_HOME = C:\jdk1.5.0
  3. PATH = C:\ANT\bin

### Instalación de iReport.

- Descargar la version iReport-0.3.2.zip o superior.
- Descomprimir el archivo zip
- Crear un directorio con el nombre iReport en el directorio raíz C:\.
- Copiar en contenido del iReport en la carpeta creada.
- Copiar el archivo **tools.jar** que se encuentra en "C:\jdk1.5.0\lib" del jdk al directorio "C:\iReport\lib"
- Colocar el Driver "**classes12.jar**" de oracle en el directorio "C:\iReport-0.2.2\lib" (Aplica para cualquier driver JDBC que necesitemos)
- Configurar las variables de entorno:
  1. CLASSPATH: C:\iReport\lib\jasperreports-0.6.0.jar.
  2. CLASSPATH: C:\iReport\lib\itext-1.02b.jar (este archivo es para el uso de archivos pdf)
- Finalmente deberá compilar los fuentes. Para esto ubíquese en C:\iReport y ejecute el comando **ant**

### Levantando el iReport

Para levantar la aplicación del iReport, debe dirigirse hasta el lugar donde descomprimió los archivos, ahí ejecutar el comando **iReport.bat** como se muestra en la siguiente figura:

```

C:\WINNT\system32\cmd.exe
25/04/2005  14:31                12.893 Prueba5.jrxml
25/04/2005  14:31                3.002 Prueba5Scriptlet.java
25/04/2005  15:50               10.261 Prueba7.jasper
25/04/2005  15:50                1.231 Prueba7.pdf
21/04/2005  11:01                4.932 NoReport.jrprint
03/04/2004  10:49                1.342 jasperreports.gif
21/04/2005  11:01                2.451 NoReport.jrxml
26/04/2005   9:27                4.047 NoReport2.jrpxml

Directorio de D:\Kruger\Kirs\Java\XML\iReport-0.4.1-src
36 archivos      229.446 bytes
11 dirs  19.124.633.600 bytes libres

D:\Kruger\Kirs\Java\XML\iReport-0.4.1-src>dir *.bat
El volumen de la unidad D es NUEVO VOL
El número de serie del volumen es: E437-EAF4

Directorio de D:\Kruger\Kirs\Java\XML\iReport-0.4.1-src
12/05/2004  00:13                191 iReport.bat
1 archivos      191 bytes
0 dirs  19.124.633.600 bytes libres

D:\Kruger\Kirs\Java\XML\iReport-0.4.1-src>iReport
  
```

Otra forma de levantar la aplicación es a través de ANT, para ello se debe abrir una ventana de cmd, ubicarse en la ruta donde se encuentra instalado el iReport, ej : C:\iReport y colocar el comando **ant iReport**, como se muestra en la siguiente figura:

```

C:\WINNT\system32\cmd.exe - ant iReport
Microsoft Windows [Versión 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

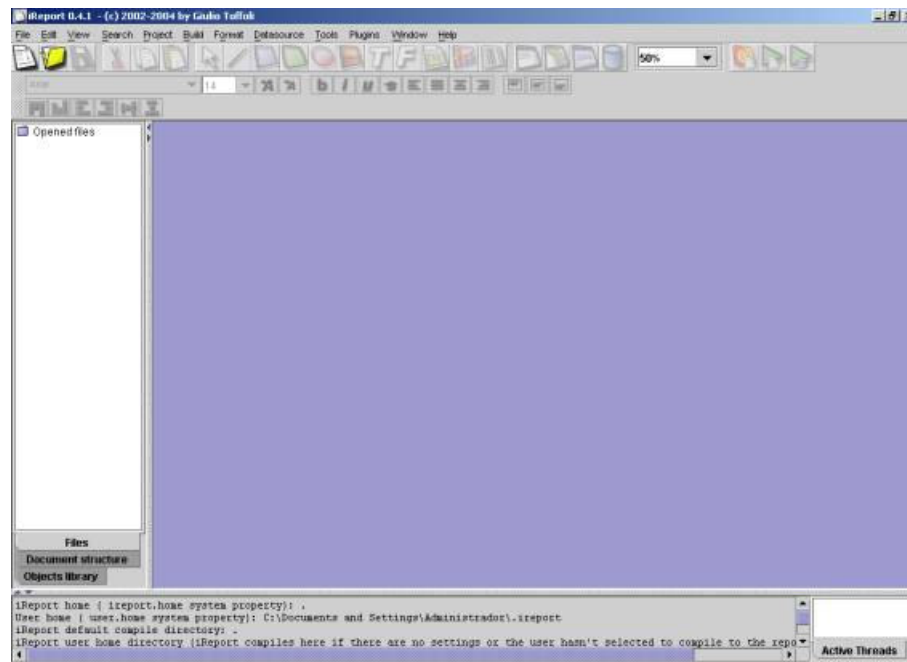
C:\>ant
Buildfile: build.xml does not exist!
Build failed
C:\>cd iReport-0.2.2

C:\iReport-0.2.2>ant
Buildfile: build.xml

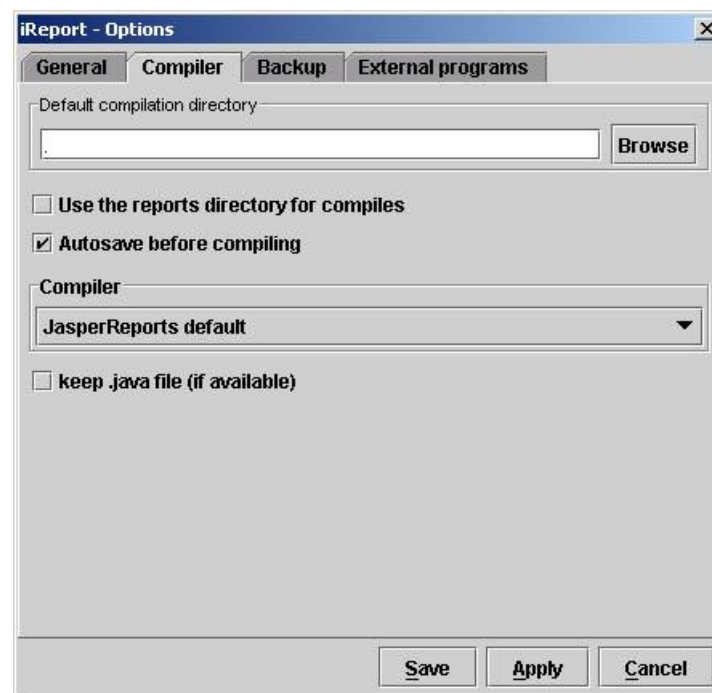
prepare:
compile:
BUILD SUCCESSFUL
Total time: 5 seconds
C:\iReport-0.2.2>ant iReport
Buildfile: build.xml

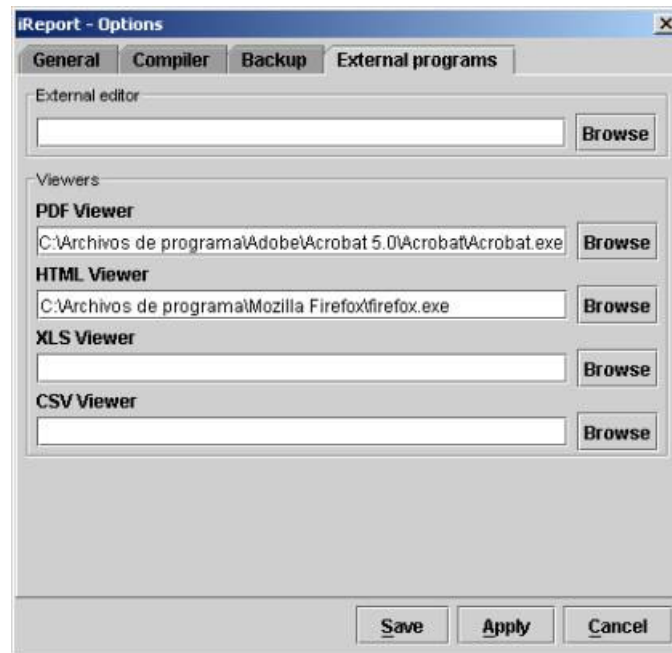
prepare:
compile:
iReport:
  
```

La interfaz gráfica de iReport comienza a cargarse:

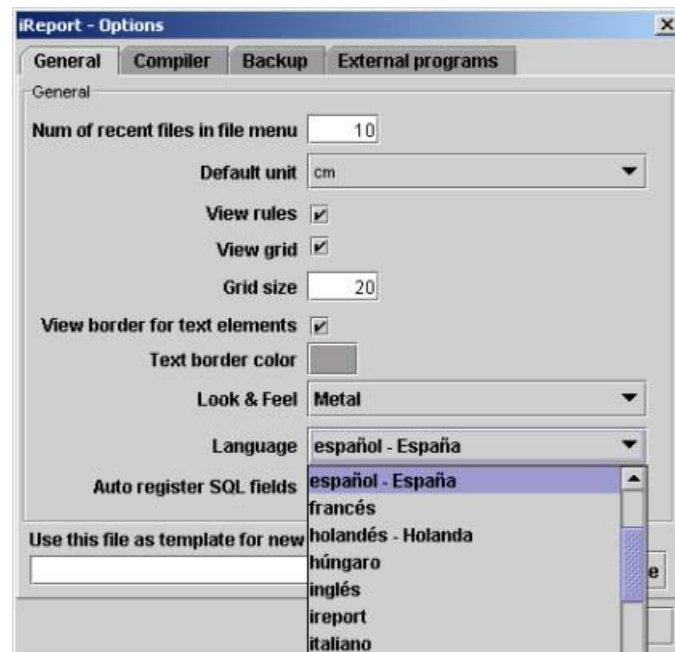


Configurar las opciones: Ir a Tools → Options





Para configurar el idioma: ir a Tools → Options

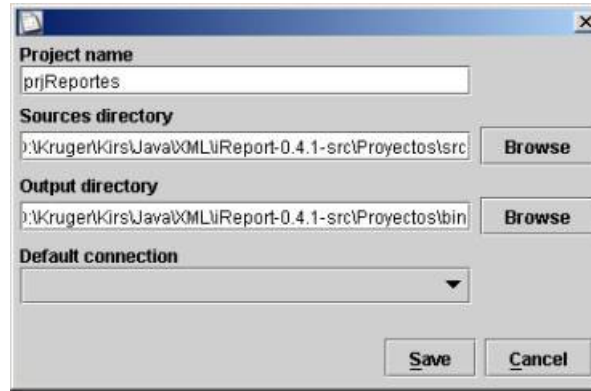


Luego presionar "Apply" y por último "Save". Notaremos como el idioma de la barra de herramientas ha cambiado en función de nuestra elección.

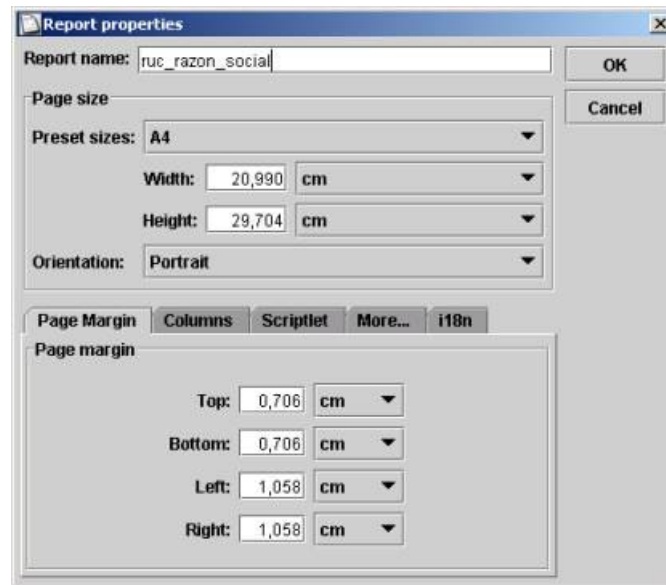


#### **Para empezar con iReports:**

Seleccionar "Project → New"



Seleccionar "File → New Document"



Aquí deberá definir el nombre del reporte, tamaño de la pantalla, orientación, márgenes, etc.

La apariencia del layout (plantilla) de la página del reporte es la siguiente:

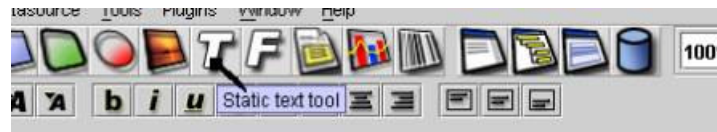
title
pageHeader
columnHeader
detail
columnFooter
pageFooter
summary

Donde:

- **title** Aparece sólo al inicio del reporte. El título se escribe en esta sección. Ejemplo: "Reporte de Desempeño de los empleados"
- **pageHeader** Aparece en la parte superior de cada página. Puede contener información como la fecha y hora, nombre de la organización, etc.
- **columnHeader** Sirve para listar los nombres de los campos que se van a presentar (desplegar). Por ejemplo: "Nombre del Empleado", "Hora de Entrada", "Hora de Salida", "Horas trabajadas", "Fecha", etc.
- **detail** En esta sección se despliegan los valores correspondientes a las entradas de campos definidos en la sección anterior. Por ejemplo "Juan Perez", "09:00", "18:00", "9", "2005-04-27"
- **columnFooter** Puede presentar información resumida para cada uno de los campos. Por ejemplo "Total de Horas Trabajadas: 180"
- **pageFooter** Aparece en la parte inferior de cada página. Esta parte puede presentar, el contador de páginas como "Página 1/7"
- **summary** Esta sección se usa para proporcionar información resumida de los campos presentes en la sección "detail" por ejemplo para el caso de las horas trabajadas de cada empleado se puede definir un objeto gráfico tipo "pie" para tener una mejor comparación y comprensión visual de los datos.

Los objetos de texto, imágenes, etc pueden ser colocados en la plantilla del documento usando el método de arrastrar y pegar.

Para insertar texto puro usar:



Para insertar campos (Fields) usar:



Para insertar imágenes usar:

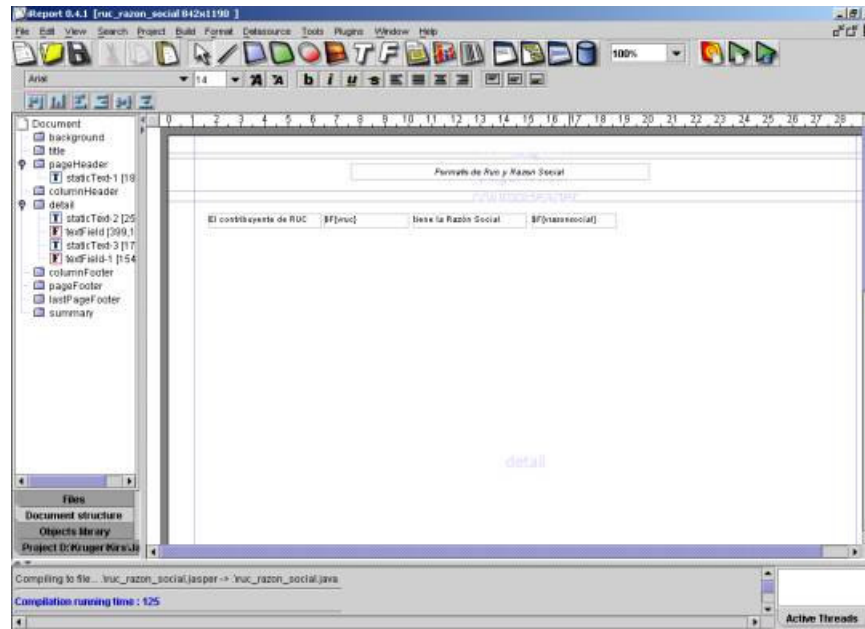


Cuando insertamos un objeto de texto lo haremos arrastrando el elemento hasta el lugar del documento donde queramos visualizarlo. Seguidamente se presionará clic derecho para editar las propiedades del objeto y así poder definir los valores de fuente, bordes y el contenido mismo del texto.



A medida que insertemos objetos de texto (estáticos y campos dinámicos), nuestro reporte empezará a definir la apariencia visual que buscamos.





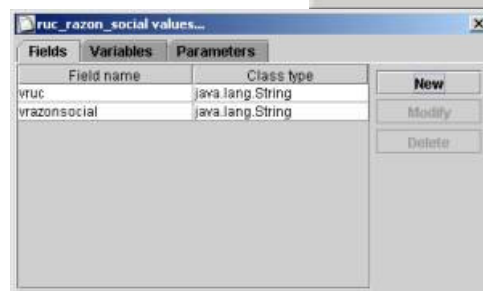
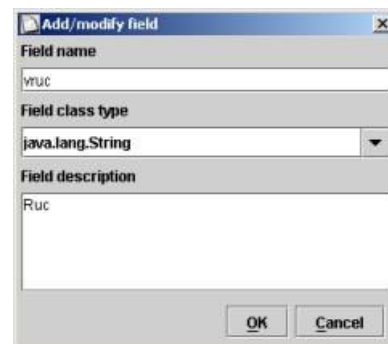
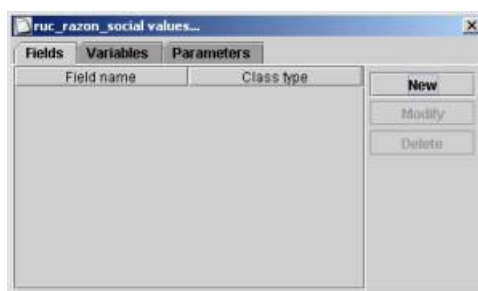
Los documentos que se generan con iReports + jasperreports tienen varios valores que son seteables o configurables, estos son parámetros, campos (Fields) y variables:

**a) Parámetros:** son valores generales como el titulo del reporte / subreporte, ubicación, autor, etc.

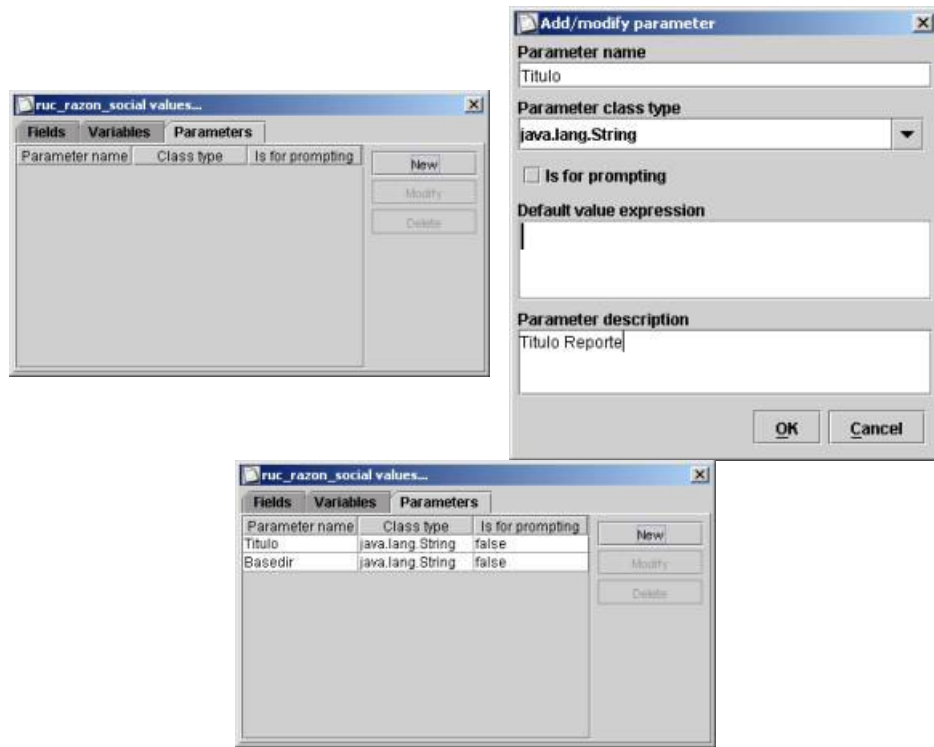
**b) Campos (Fields):** son los campos tipo texto que llenaremos desde las consultas lanzadas a la aplicación durante la ejecución del reporte.

**c) Variables:** una variable es un valor calculado como una sumatoria.

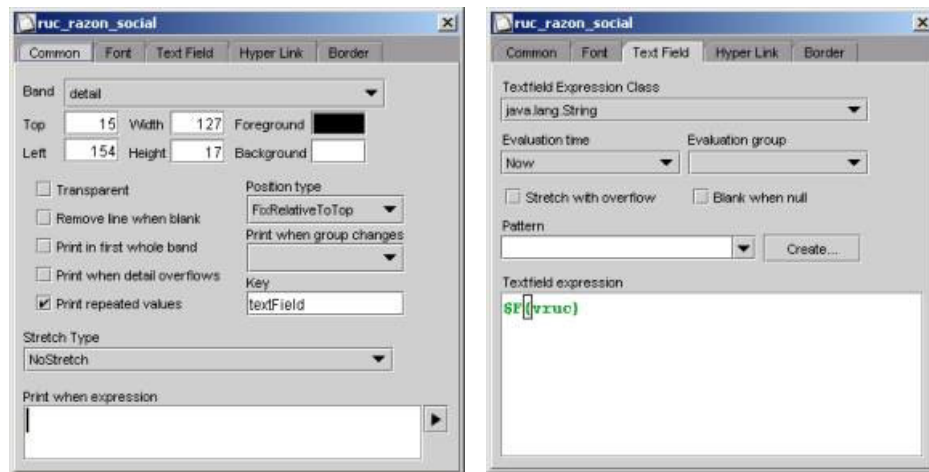
Para definir los parámetros se tiene que pulsar el botón "Parameters" desde la barra de herramientas



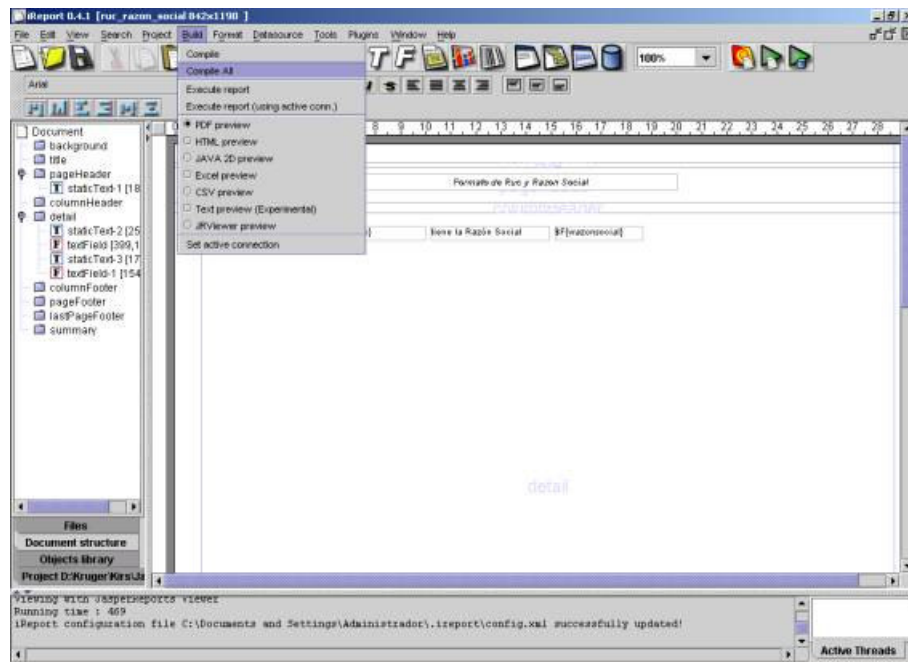




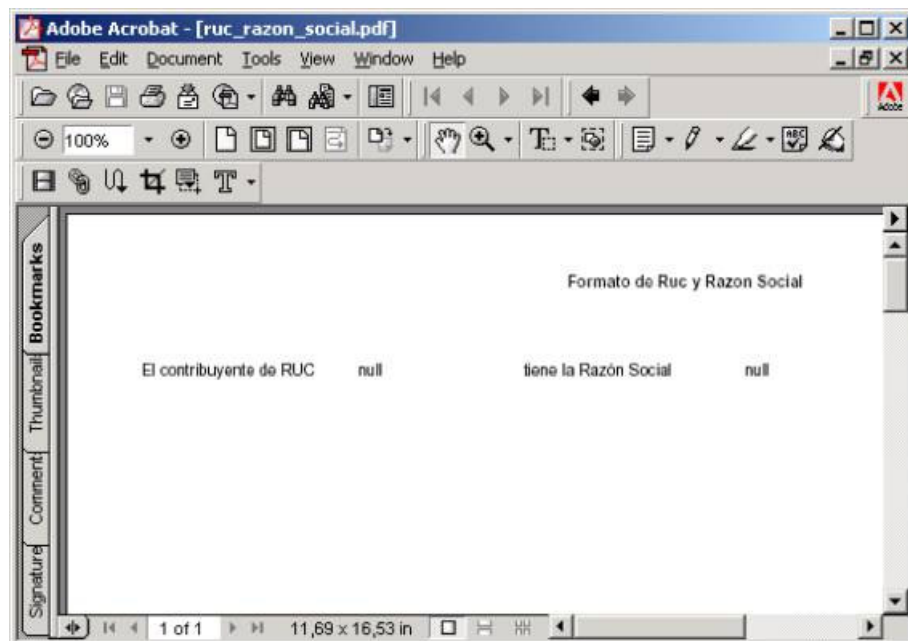
Los objetos de tipo TextField, deben definir un campo field en su viñeta de propiedades y deben usar el key "textField". En general debe configurarse como se indica en las siguientes pantallas.



Una vez que hemos terminado de diagramar nuestro reporte, debemos compilar y luego ejecutarlo. La herramienta nos permite tener una vista previa en varios formatos de la apariencia de nuestro documento.



Una vez compilado, se tiene que ejecutar el reporte. Para esto iremos a Build → Execute Report

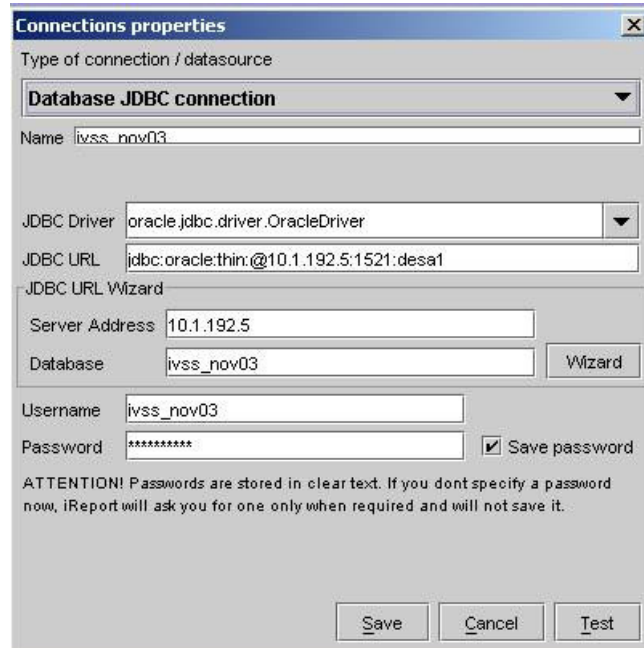


Observamos que cuando no definimos un datasource los campos aparecen como null.

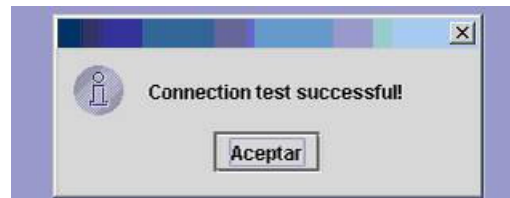
### **Conexión con la Base de Datos.**

Seleccionar DataSource--> Connection/DataSource-->New

- Name: Escribir el nombre de la base de datos.
- JDBC Driver: Seleccionar el Driver de la base de datos con la cual se realizará la conexión.
- JDBC URL: escribir la dirección URL de donde se encuentra alojado el ambiente de trabajo de la base de datos.
- Especificar la cadena de conexión con la base de datos, colocar usuario y password.
- Salvar la configuración de la conexión.



Presionar el botón Test, para verificar que la conexión este correcta, para la cual deberá aparecer la siguiente ventana:



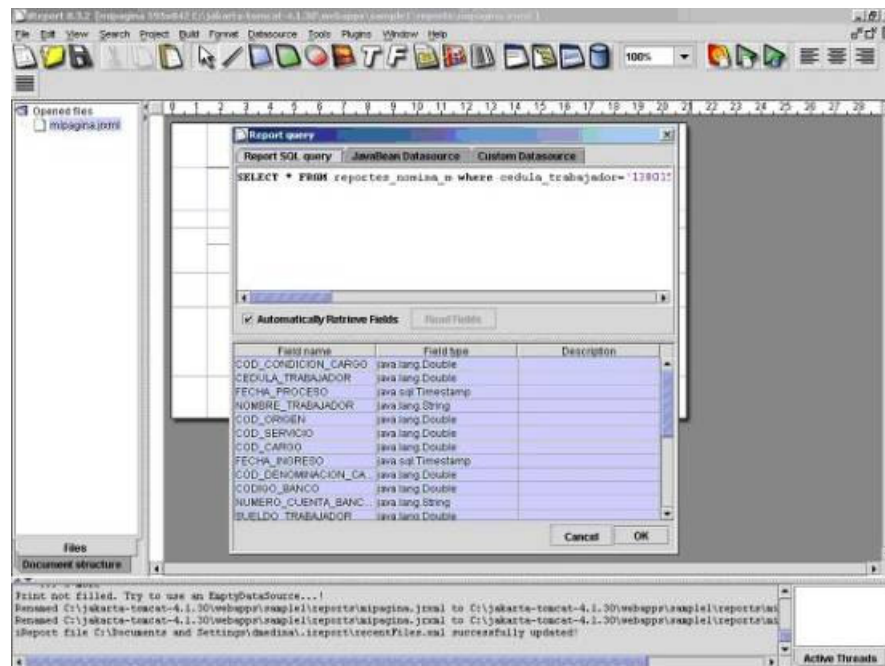
Para seleccionar la conexión configurada: Ir al Menú y seleccionar " Build → Set Active Connection". Aparecerá una pantalla como la que se muestra a continuación:



### Incluir clases en iReport

Pasos para incluir un clase java en el iReport:

- Agregamos nuestros campos de la Base de Datos:



- Creamos la clase que vamos a utilizar:

CalculosFecha.java

```
import java.sql.*;

import java.util.*;

import java.text.*;

import net.sf.jasperreports.engine.*;

public class CalculosFecha extends JRDefaultScriptlet
{

    public static String user = "xxxxxx";

    public static String pasw = "xxxx";

    public static String driver = "oracle.jdbc.driver.OracleDriver";

    public static String url = "xxxxxx";

    public static ResultSet rs = null;

    public static Statement st = null;

    public static Connection cn = null;

    public static PreparedStatement pst = null;

    public static int per=0;

    /*

    Datos del Reporte

    */

    public String hello() throws JRScriptletException
    {

        return "Mi Primera clase";
    }
}
```

```

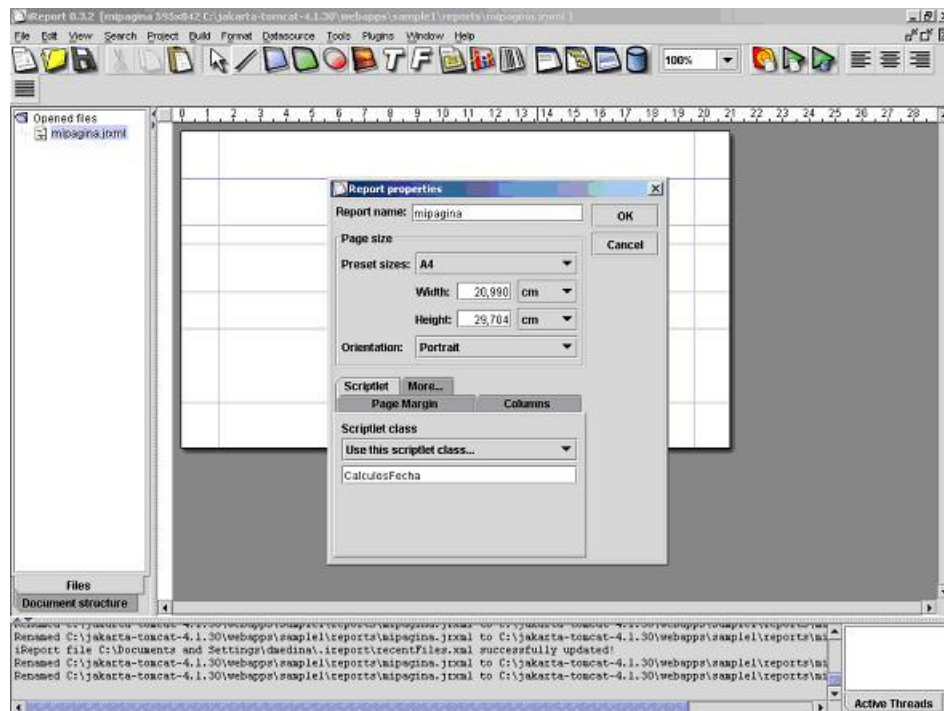
    }
}

```

- Le indicamos que vamos a usar una clase:

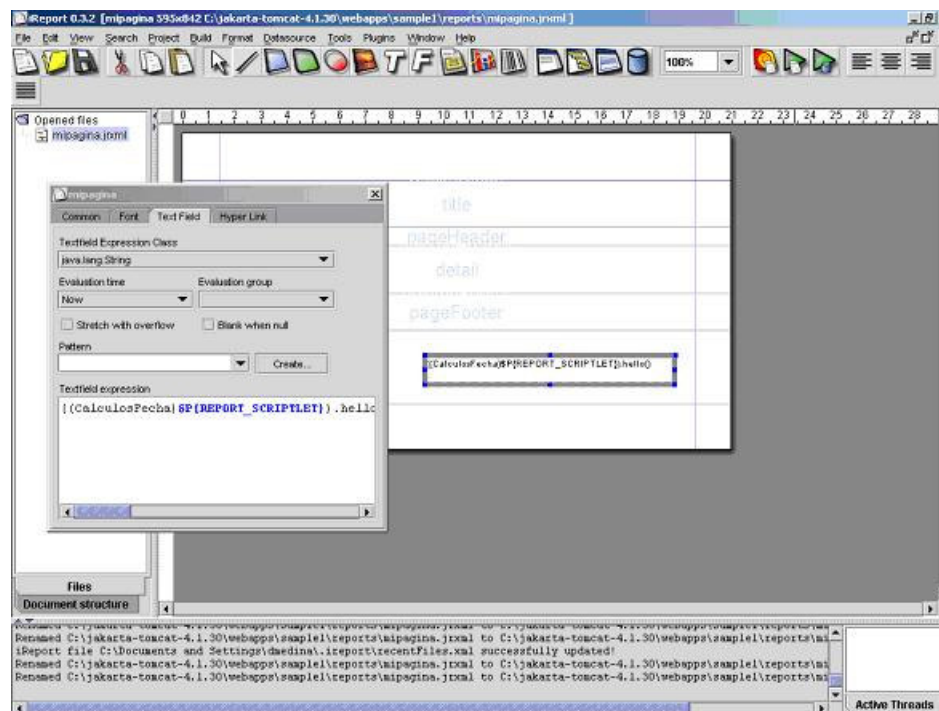
Uses this Scriptlet class..

Le colocamos el nombre de la clase: CalculosFecha (en este caso)



- Agregamos la descripción de la clase en el iReport:

TextField Expresión:((CalculosFecha)\$P{REPORT\_SCRIPTLET}).hello()



- Compilamos desde mi servidor web, porque si lo hago directamente del iReport me dara error. Entonces lo compilo desde el WEB-SERVER.

```

<!--/*
*****

*      Formulario: scriplet.jsp.      *
*      Autor: Delia M Medina C.      *
*      Fecha de Creacion: 28-09-2004      *
*****

*      Funcion:      ****
*****

*/-->

<%@ page import="net.sf.jasperreports.engine.*" %>

<%@ page import="net.sf.jasperreports.engine.design.*" %>

<%@ page import="net.sf.jasperreports.engine.data.*"%>

<%@ page import="net.sf.jasperreports.engine.export.*"%>

<%@ page import="net.sf.jasperreports.engine.util.*"%>

<%@ page import="net.sf.jasperreports.view.*"%>

<%@ page import="net.sf.jasperreports.view.save.*"%>

<%@ page import="java.sql.*"%>

<%@ page import="java.util.*" %>

<%@ page import="java.io.*" %>

<html>

<head><title>Generando Reporte Nomina.....</title>

<%

        System.out.println("Comienza el Reporte-----");

%>

<script language="JavaScript">

function regresar(){

        history.back();

}

function error(){

        alert("Error encontrando procesando el periodo: " + periodo);

        history.back();

}

</script>

<head>

<%

try{

        ResultSet rs = null;

```

```

Statement st = null;

String sql = null;

String driver = "oracle.jdbc.driver.OracleDriver";

String connectString = "*****";

String user = "***** ";

String password = "*****";

Class.forName(driver);

Connection conn = DriverManager.getConnection(connectString, user, password);

////////////////////////////////////

System.out.println("*****Compilamos pago_nomina.jrxml OK*****");

System.setProperty(

    "jasper.reports.compile.class.path",

    application.getRealPath("/WEB-INF/lib/jasperreports-0.6.0.jar") +

    System.getProperty("path.separator") +

    application.getRealPath("/WEB-INF/classes/")

);

System.out.println("*****Cargamos el jasperreports-0.6.0.jar OK*****");

System.setProperty(

    "jasper.reports.compile.temp",

    application.getRealPath("/reports/")

);

JasperCompileManager.compileReportToFile(application.getRealPath("/reports/mipagina.jrxml"));

System.out.println("*****Fin de la Compilamos el archivos*****");

////////////////////////////////////

File reportFile = new File(application.getRealPath("/reports/mipagina.jasper"));

Map parameters = new HashMap();

byte[] bytes =

    JasperRunManager.runReportToPdf(

        reportFile.getPath(),

        parameters,

        conn

    );

response.setContentType("application/pdf");

```



```

        response.setContentLength(bytes.length);

        ServletOutputStream ouputStream = response.getOutputStream();

        ouputStream.write(bytes, 0, bytes.length);

        ouputStream.flush();

        ouputStream.close();

        System.out.println("jasperPDF OK.....");

        //////////////////////////////////////

        System.out.println("Fin del reporte pago_reporte_nomina.....");

    } catch (JRException e)

    {System.out.println("Error:" + e.getMessage());}

    catch (Exception e)

    {

        e.printStackTrace();

        System.out.println("Error2:" + e.getMessage());

    }

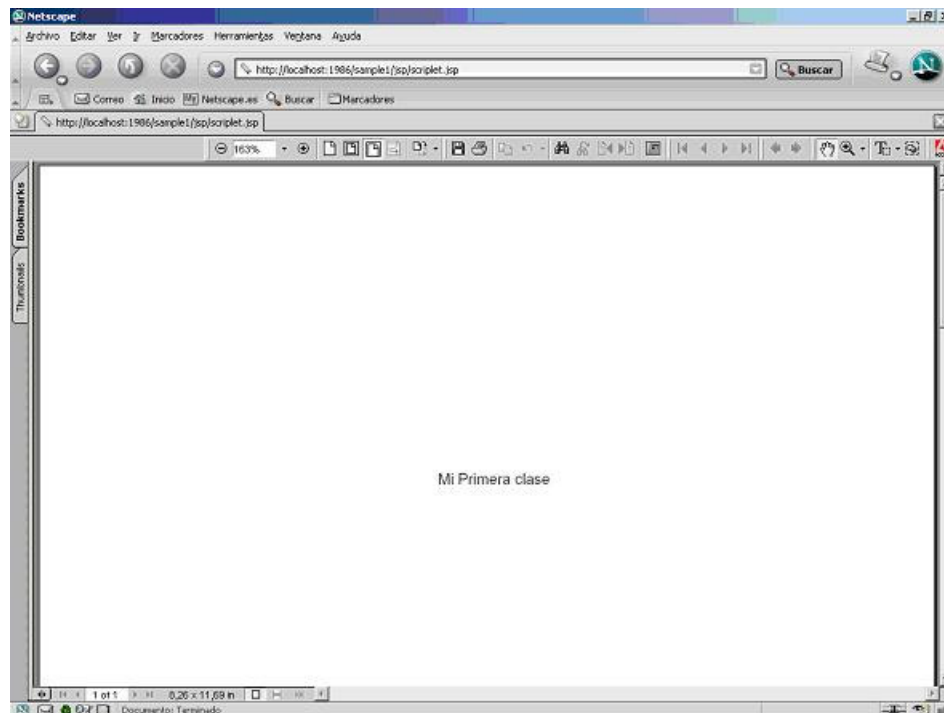
%>

<body >

</body>

</html>

```



#### Fuentes:

- Documentación de iReports
- Documento de Delia Medina, Manual de iReports

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

**J2EE, EJBs, Struts...**

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.

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

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

Nombre Corto	Descripción
<a href="#">XML básico</a>	Si quieres ver de un modo visual como crear un documento XML, este es tu tutorial. Este es el primero de un conjunto de tutoriales que iremos publicando sobre esta fascinante y amplia tecnología
<a href="#">Generación de Informes pdf con DataVision</a>	Os mostramos como simplificar la creación de informes un múltiples formatos, incluyendo pdf, con una fantástica herramienta visual (DataVision). Os mostramos como integrarla con Servlets
<a href="#">Schemas XML. Introducción esquemas XML</a>	Los esquemas XML (schemas XML) son una evolución natural de las DTDs. Os mostramos como empezar con esta tecnología.
<a href="#">XSLT:FO y FOP</a>	Os mostramos como generar documentos PDF desde XML gracias a FOP y XSLT:FO
<a href="#">Crear pdfs a partir de páginas HTML</a>	Os mostramos como agrupar y convertir páginas HTML que os interesen en documentos PDF a través de herramientas gratuitas
<a href="#">Modelado UML con Visual Paradigm</a>	Os mostramos como instalar y utilizar la versión gratuita de Visual Paradigm for UML. Cabe destacar que permite extraer elementos de diseño desde textos de análisis.
<a href="#">Gráficas en Java con JFreeChart</a>	Os mostramos como generar gráficas profesionales, en aplicaciones y servlets, en Java con la librería gratuita JFreeChart
<a href="#">Introducción a ANT</a>	En el mundo Java, la compilación, verificación e instalación de aplicaciones se ha normalizado con este potente paquete llamado ANT.
<a href="#">Construcción ficheros HLP usando HTML</a>	Os mostramos como montar un fichero HLP a partir de un árbol estático de HTML con la herramienta HTMLtoHLP
<a href="#">Introducción a log4j</a>	En un desarrollo Java es vital normalizar los logs para posteriormente poder depurar el funcionamiento de nuestra aplicación. Os mostramos como usar Log4j.

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](mailto:rcanales@adictosaltrabajo.com) para su resolución.

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

