

¿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

Control de autenticación y
 acceso (Spring Security)
 UDDI

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

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

Web Services
 Rest Services
 Social SSO
 SSO (Cas)

Dirección de Proyectos Informáticos.
 Metodologías ágiles
 Patrones de diseño
 TDD

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

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

AdictosAlTrabajo

Terrakas 1x05
¡¡Ya está en la web!! :-)
terrakas.com



autentia
Soporte a desarrollo informático
Hosting patrocinado por
ENREDADOS

Entra en Adictos a través de  

E-mail

Contraseña

Entrar [Deseo registrarme](#)
[Olvidé mi contraseña](#)



[Inicio](#) [Quiénes somos](#) [Formación](#) [Comparador de salarios](#) [Nuestros libros](#) [Más](#)

» Estás en: [Inicio](#) [Tutoriales](#) [Trabajar con tablas en JasperReport](#)



Rubén Aguilera Díaz-Heredero

Consultor tecnológico de desarrollo de proyectos informáticos.

Ingeniero en Informática, especialidad en Ingeniería del Software

Puedes encontrarme en [Autentia](#): Ofrecemos servicios de soporte a desarrollo, factoría y formación

Somos expertos en Java/J2EE



[Ver todos los tutoriales del autor](#)



Fecha de publicación del tutorial: 2013-02-12

Tutorial visitado 1 veces [Descargar en PDF](#)

Trabajar con tablas en JasperReport

0. Índice de contenidos.

- 1. Entorno
- 2. Introducción
- 3. Creación del informe con una tabla
- 4. Paso de parámetros a la tabla desde el informe principal

1. Entorno

Este tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil Mac Book Pro 17" (2,6 Ghz Intel Core i7, 8 GB DDR3)
- Sistema Operativo: Mac OS X Snow Leopard 10.6.4
- Jaspersoft Studio 2.0

2. Introducción

La mayoría de nosotros ya conocemos y hemos hecho algún informe con JasperReport de hecho se ha convertido en la herramienta de referencia en cuanto una historia habla de realizar un informe del tipo que sea. En el 90% de los casos solo necesitaremos un único dataset que nos proporcione los datos. Para lo cual las columnas las ponemos en la banda "Column Header" y los datos en la banda "Detail".

En este tutorial vamos a ver como podemos mostrar datos de distintos dataset dentro de un mismo informe gracias al componente tabla que podemos encontrar en la paleta de componentes de la herramienta y que permite definir un dataset independiente. También veremos como podemos hacer que los valores de la tabla dependan de una variable de entrada a nivel de informe.

3. Creación del informe con una tabla

Para ponerlo en práctica vamos a partir del siguiente esquema de base de datos.

```
CREATE TABLE TEST
(
  CODIGO    INTEGER NOT NULL,
  NOMBRE    VARCHAR2(255) NOT NULL,
  TIPO      INTEGER NOT NULL,
  CONSTRAINT PK_TEST PRIMARY KEY (CODIGO)
);
```

Cargamos la tabla TEST con los siguientes datos:

```
insert into TEST values (1, 'A', 1);
insert into TEST values (2, 'B', 1);
insert into TEST values (3, 'C', 2);
insert into TEST values (4, 'D', 3);
insert into TEST values (5, 'E', 2);
insert into TEST values (6, 'F', 2);
insert into TEST values (7, 'G', 3);
insert into TEST values (8, 'H', 1);
insert into TEST values (9, 'I', 1);
insert into TEST values (10, 'J', 4);
insert into TEST values (11, 'K', 2);
insert into TEST values (12, 'L', 3);
insert into TEST values (13, 'M', 1);
insert into TEST values (14, 'N', 3);
```

Catálogo de servicios Autentia



Síguenos a través de:



Últimas Noticias

» Comentando el libro: La inteligencia reformada, las inteligencias múltiples en el siglo XXI de Howard Gardner

» Hangout "El precio de un servicio"

» Comentando el Libro HACER: ¡NADA! DE J. Keith Murnighan

» ¿Qué es aportar valor como técnico/programador?

» Como arrancar un nuevo proyecto e integrar el agilidad en una organización

[Histórico de noticias](#)

Últimos Tutoriales

» Gestión de expedientes en el ámbito de las Administraciones Públicas (I): ámbito funcional.

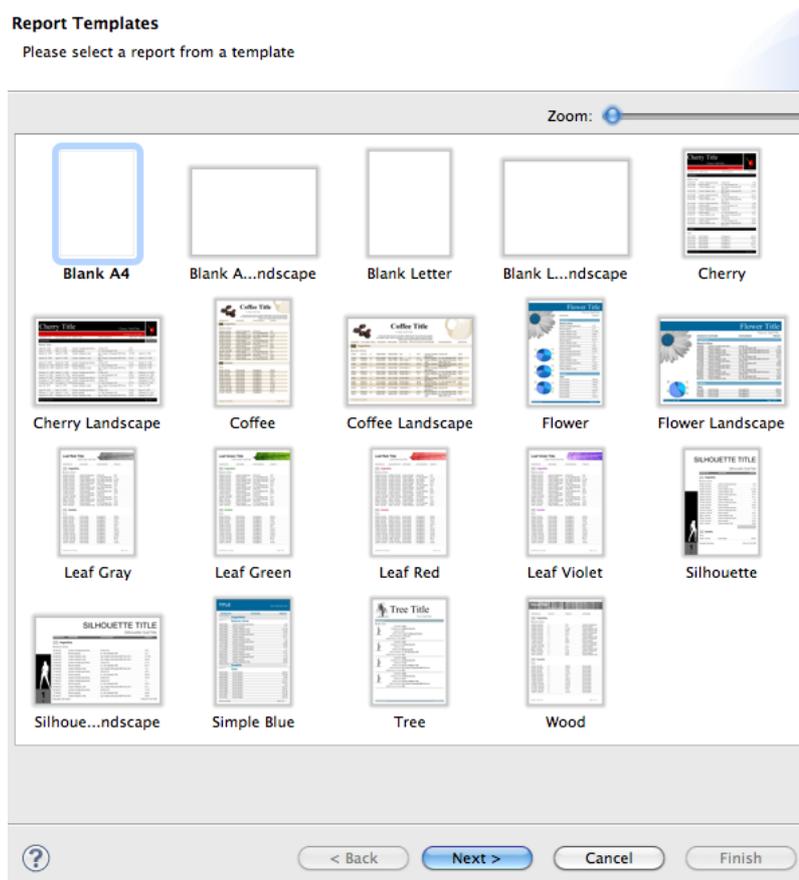
» CDI: Inyección de dependencias en JEE y ejecución de test de integración con el soporte de Arquillian.

» PrimeUI: set de componentes visuales en javascript - un spin-off de Primefaces.

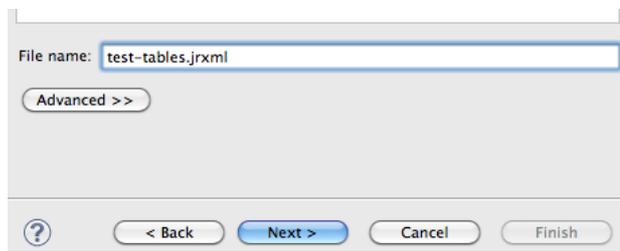
» ¡Reactiva tu dinosaurio!

Lo primero que tenemos que hacer es abrir una instancia de Jaspersoft Studio, asegúrate de que es al menos la versión 2.0 porque las anteriores dan muchos problemas de estabilidad y podrías llegar a perder todos los cambios.

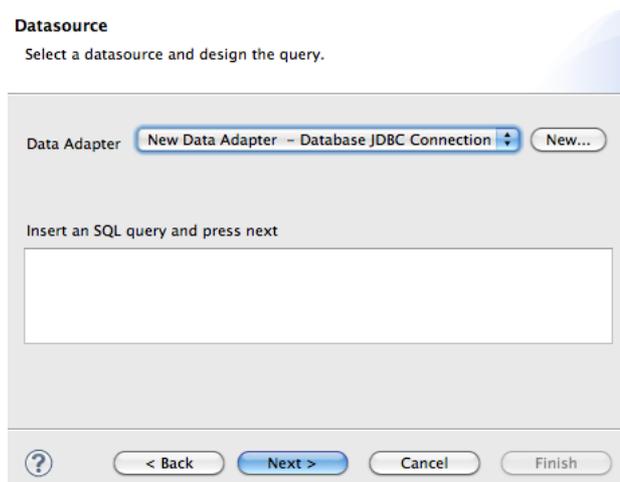
Ahora creamos un informe en blanco. Para ello seleccionamos File --> New --> Jasper Report con lo que se muestra el primer paso del asistente. En este paso seleccionamos que queremos un informe en blanco (Blank A4) y pulsamos en "Next".



En el siguiente paso establecemos el nombre del fichero del informe, la localización dentro de la estructura de proyectos y pulsamos "Next".



Ahora tenemos que definir la fuente de datos que vamos a utilizar en los informes.



Al pulsar en "New" se muestra un segundo asistente que nos permite definir la fuente de datos.

» [Instalación de Oracle WebLogic 12c en Mac OS.](#)

Últimos Tutoriales del Autor

» [Aspectos avanzados de Sitemesh](#)

» [Introducción a Sitemesh](#)

» [Robolectric: aplicando TDD en Android](#)

» [Trabajando en Android con Maven](#)

» [Sonar Runner: Analizar proyectos sin Maven en cualquier lenguaje](#)

Últimas ofertas de empleo

2011-09-08
[Comercial - Ventas - MADRID.](#)

2011-09-03
[Comercial - Ventas - VALENCIA.](#)

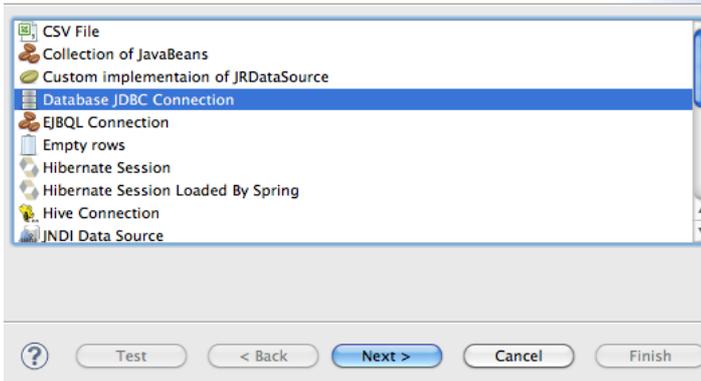
2011-08-19
[Comercial - Compras - ALICANTE.](#)

2011-07-12
[Otras Sin catalogar - MADRID.](#)

2011-07-06
[Otras Sin catalogar - LUGO.](#)

Data Adapters

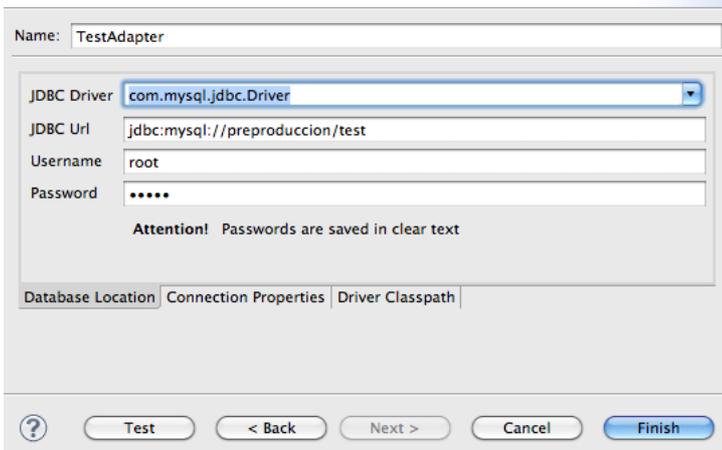
Use SQL queries to get data from a database



Seleccionamos la opción de fuente de datos a partir de conexión JDBC (Database JDBC Connection) y pulsamos en "Next". En esta pantalla establecemos el nombre del adaptador y los datos de conexión con la base de datos.

Data Adapter

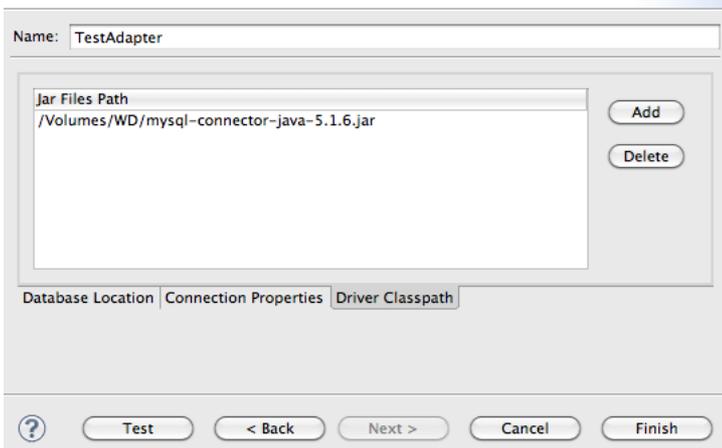
Database JDBC Connection



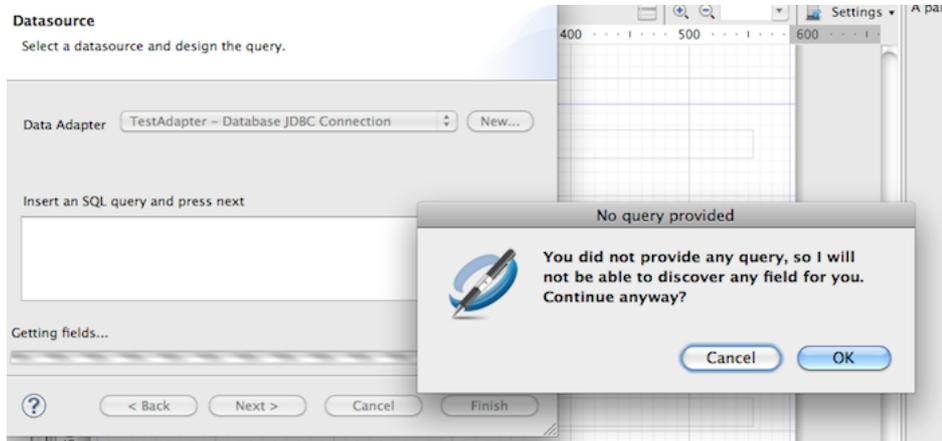
No hay que olvidar incluir el Driver específico para la base de datos deseada. Para ello pulsamos en la pestaña "Driver Classpath" y seleccionamos mediante "Load" el driver requerido.

Data Adapter

Database JDBC Connection



Para comprobar que la conexión es correcta, pulsamos en "Test" y debemos ver el mensaje "Successful". Pulsamos "Finish" para volver al asistente principal donde al pulsar en "Next" nos muestra una advertencia diciendo que no puede descubrir campos a nivel de informe, le damos a "OK", ya que no nos interesa este descubrimiento.

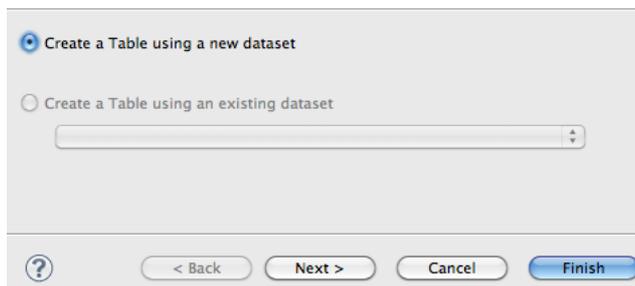


Por último en la pantalla de felicitaciones pulsamos en "Finish".

El objetivo de este tutorial es crear un informe con una tabla de datos. Para ello lo que hacemos es arrastrar a la banda "Detail" un componente de tipo tabla desde la paleta de componentes, lo que hace que se muestre un asistente donde

Dataset

Please select or build a dataset that will be used for this component.

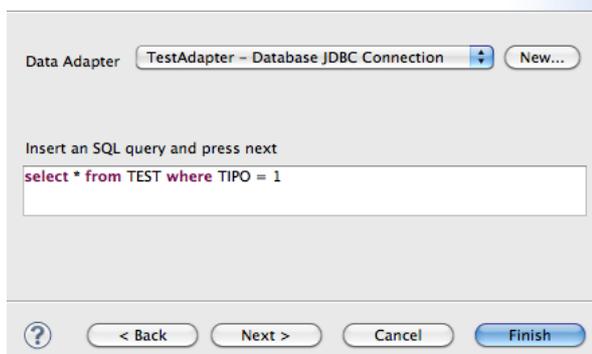


nos indica la creación de un nuevo Dataset.

En el siguiente paso vamos a establecer el nombre al nuevo dataset (TestDatasetTipo1) e indicamos que lo queremos crear a partir de una fuente de datos (Create new dataset from a connection or datasource). En la siguiente pantalla seleccionamos el adaptador creado anteriormente e introducimos la sentencia SQL que nos devuelva los datos que queramos mostrar en forma de tabla. En nuestro caso queremos que esta tabla muestre los elementos que sean de tipo 1.

Datasource

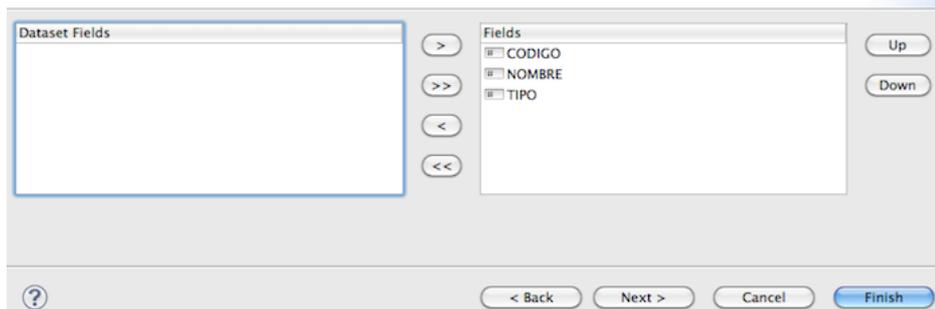
Select a datasource and design the query.



Pulsando en "Next" el asistente nos descubre los campos del dataset en función de la consulta introducida. Seleccionamos los campos que vamos a querer que estén disponibles para creación de la tabla, en mi caso, selecciono todos.

Fields

Please select dataset fields



Después podemos seleccionar de que campos queremos hacer "group by", en mi caso, no selecciono nada.

Group By

Please select fields to Group By

Ahora seleccionamos el tipo de conexión, para nuestro caso, seleccionamos la misma que la del informe.

Connection

Please select the connection

Ahora seleccionamos las columnas que queremos mostrar en la tabla, en mi caso, selecciono todos.

Table columns

Select the fields of the dataset that will be used to produce table columns. If no fields are selected, the table will be created without any column.

Es momento de seleccionar el layout de la tabla, en este caso basta con mostrar el header de las columnas.

Layout

Please select layout

Pulsando en "Finish" ya podemos ver la tabla en el informe. Para probar el resultado podemos pulsar en la pestaña "Preview" donde nos vamos a llevar la sorpresa de que solo aparece el mensaje "Document is Empty". Esto es así porque si el dataset del informe no tiene datos no se muestra ninguna sección del informe. La forma más rápida de resolver esto es ir a la pestaña "Source" e introducir a mano una sentencia SQL que devuelva datos dentro de la etiqueta `<queryString>` que esta fuera de la etiqueta `<subDataset>`, como se muestra en la imagen:

```

<subDataset name="TestDatasetTipo1" uuid="ff20eaf0-4e05-4007
<property name="com.jaspersoft.studio.data.defaultdataac
<queryString>
<![CDATA[select * from TEST where TIPO = 1]]>
</queryString>
<field name="CODIGO" class="java.lang.Integer">
<fieldDescription><![CDATA[]]></fieldDescription>
</field>
<field name="NOMBRE" class="java.lang.String">
<fieldDescription><![CDATA[]]></fieldDescription>
</field>
<field name="TIPO" class="java.lang.Integer">
<fieldDescription><![CDATA[]]></fieldDescription>
</field>
</subDataset>
<queryString>
<![CDATA[select * from TEST]]>
</queryString>
<background>

```

Hecho esto si volvemos a ir a la pestaña "Preview" debemos ver un resultado parecido a este.

CODIGO	NOMBRE	TIPO
1	A	1
2	B	1
8	H	1
9	I	1
13	M	1

4. Paso de parámetros a la tabla desde el informe principal

Ahora lo que queremos conseguir es que el parámetro tipo de la sentencia SQL dependa de un parámetro externo que se defina a nivel del informe principal. Para hacer esto nos podemos apoyar en la herramienta aunque lo más rápido es editar directamente el código dentro de la pestaña "Source". No tengáis miedo porque el compilador de Jasper Reports nos indicará si lo estamos editando correctamente.

Primero definimos un parámetro de entrada a nivel de informe de tipo Integer con el nombre tipo.

```

</field>
<field name="TIPO" class="java.lang.Integer">
<fieldDescription><![CDATA[]]></fieldDescription>
</field>
</subDataset>
<parameter name="tipo" class="java.lang.Integer"/>
<queryString>
<![CDATA[select * from TEST]]>
</queryString>
<background>
<band splitType="Stretch"/>
</background>
<title>
<band height="235" splitType="Stretch">
<componentElement>

```

Luego dentro del subDataset vamos a definir un parámetro también de tipo Integer con el nombre tipo_tabla el cual vamos a referenciar en la propia consulta SQL con la sintaxis \$P{variable}

```

<style name="table_in" mode="Opaque" bgcolor="#F0F0F0"/>
<style name="Table_CH" mode="Opaque" bgcolor="#BFF3FF"/>
<style name="Table_TD" mode="Opaque" bgcolor="FFFFFF"/>
<subDataset name="TestDatasetTipo1" uuid="ff20eaf0-4e05-4007-a6db-037baacc5a39">
<property name="com.jaspersoft.studio.data.defaultdataadapter" value="TestAdapter"/>
<parameter name="tipo_tabla" class="java.lang.Integer"/>
<queryString>
<![CDATA[select * from TEST where TIPO = $P{tipo_tabla}]]>
</queryString>
<field name="CODIGO" class="java.lang.Integer">
<fieldDescription><![CDATA[]]></fieldDescription>
</field>
<field name="NOMBRE" class="java.lang.String">
<fieldDescription><![CDATA[]]></fieldDescription>
</field>
<field name="TIPO" class="java.lang.Integer">
<fieldDescription><![CDATA[]]></fieldDescription>
</field>
</subDataset>
<parameter name="tipo" class="java.lang.Integer"/>
<queryString>
<![CDATA[select * from TEST]]>
</queryString>
<background>
<band splitType="Stretch"/>

```

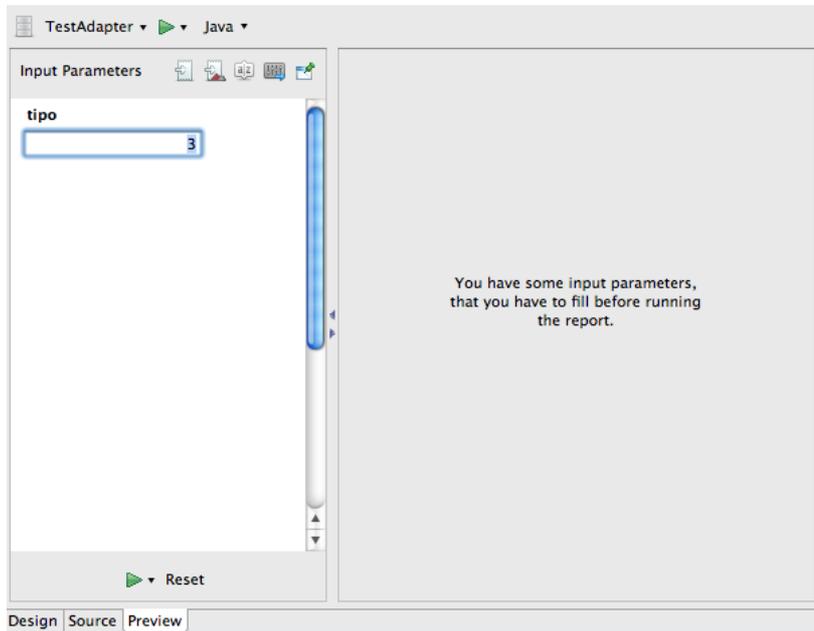
Por último para que la tabla pueda tener acceso al parámetro del informe necesitamos realizar el mapeo a nivel de tabla.

```

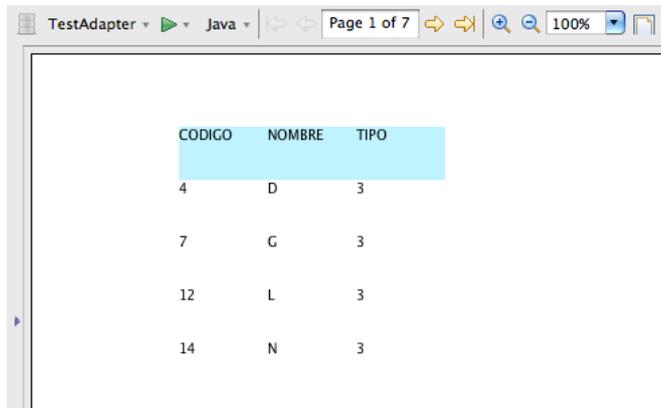
<reportelement uuid="e19b9d8c-2b3e-48e2-ba7e-2b14dct/40/e" x="90" y="35" width="200" height="200"/>
<jr:table xmlns:jr="http://jasperreports.sourceforge.net/jasperreports/components" xsi:schemaLocation="
  http://jasperreports.sourceforge.net/jasperreports/components http://jasperreports.sourceforge.net/jasperreports/components-2.0.0-rc10.xsd">
  <datasetRun subDataset="TestDatasetTipo1" uuid="29658364-c8fa-4fb3-a0a2-9bce6e76f738">
    <datasetParameter name="tipo_tabla">
      <datasetParameterExpression><![CDATA[{$P{tipo}}]></datasetParameterExpression>
    </datasetParameter>
    <connectionExpression><![CDATA[{$P{REPORT_CONNECTION}}]></connectionExpression>
  </datasetRun>
  <jr:column width="66" uuid="d8217578-74db-4d37-a51e-97ec1ca93ff5">
    <jr:columnHeader style="Table_CH" height="40">

```

De esta forma cuando volvamos a pulsar en "Preview" la herramienta nos solicita que introduzcamos un valor para la variable "tipo".



Ahora el resultado de la tabla tendrá en cuenta el valor de la variable "tipo" de entrada.



Cualquier duda o sugerencia en la zona de comentarios.

Saludos.

A continuación puedes evaluarlo:

[Regístrate para evaluarlo](#)

Por favor, vota +1 o compártelo si te pareció interesante

Share |

Animáte y coméntanos lo que pienses sobre este **TUTORIAL**:

» [Regístrate](#) y accede a esta y otras ventajas «



Esta obra está licenciada bajo licencia [Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

IMPULSA

Impulsores

Comunidad

[¿Ayuda?](#)

sin clicks

0 personas han traído clicks a esta página

+ + + + + + + +

powered by [karmacacy](#)

Copyright 2003-2013 © All Rights Reserved | [Texto legal y condiciones de uso](#) | [Banners](#) | [Powered by Autentia](#) | [Contacto](#)

