

Confección de informes

Informes	1
Ferramentas para a xeración de informes	1
Creación da base de datos	1
Creación dun proxecto JasperReports	4
Conexión coa base de datos	5
Crear un informe	8
Interface Jaspersoft Studio	11
Elementos estruturais dun informe	12
Creación dun informe sinxelo	12
Paso de parámetros a un informe	14
Crear unha agrupación nun informe	17
Integración dun informe nunha aplicación Java	20
Subinformes	23
Gráficos	31
Variables	33
Referencias	37

Informes

Un informe é un documento que permite mostrar o contido dunha **orixe de datos** aplicando un formato que permita aos usuarios ler a información de maneira atractiva.

Son un modo eficaz de presentar a información exercendo un control bastante preciso sobre como se presenta, xa que permite: ver, aplicar formato e resumir datos relacionados.

Normalmente un informe xérase a partir dun deseño, no que se determina a distribución e a configuración dos elementos a mostrar. Pode crearse un informe utilizando algunha ferramenta gráfica e despois combinar o este deseño cos datos almacenados nunha orixe de datos. O resultado mostrará os datos da orixe de datos usando o deseño configurado no informe predefinido, facilitando a lectura e interpretación ás persoas usuarias e, ao mesmo tempo, posibilitando a súa impresión e almacenamento. No caso de que haxa modificacións nos datos, os informes poden volver a xerarse para que incorporen os cambios que se produciron.

Un exemplo de informe que resumise os resultados dunha base de datos de poxas non só mostraría os rexistros individuais, senón que tamén podería agrupar os datos e mostralos, por exemplo, agrupados polo mellor ofertante. Tamén podería mostrar o subtotal, a prima do comprador e o total da suma para cada gañador. Este é un bo exemplo de como un informe pode mostrar información que simplemente non aparece por si soa nunha base de datos, xa que as táboas non presentan de forma resumida os datos dunha colección de rexistros.

Ferramentas para a xeración de informes

As ferramentas para a xeración de informes permiten, mediante unha interface gráfica, determinar o aspecto final e configuración dos elementos que aparecen no informe, xerando automaticamente os arquivos do deseño final.

Existen diferentes ferramentas para elaborar informes, tanto libres como propietarias. Neste documento mostrarase a creación de informes utilizando **Jaspersoft Studio**. Existe unha versión [community edition](#) que se pode descargar, despois de rexistrarse na páxina web.


Exercicio: crea unha conta na web de [Jaspersoft Community](#) e descarga o ficheiro **Jaspersoft-Studio-CE_6.21.0_linux_x86_64.tgz** na carpeta de [máquinas virtuais](#). Unha vez descargado o ficheiro descomprimeo e inicia o software **Jaspersoft Studio**.

Jaspersoft Studio permite a creación de informes que serán almacenados en arquivos XML, con extensión jrxml, que conteñen o deseño do informe. Para xerar o informe hai que compilar o arquivo jrxml, de tal forma que se xerará un novo ficheiro con extensión jasper que se completará coa orixe de datos.

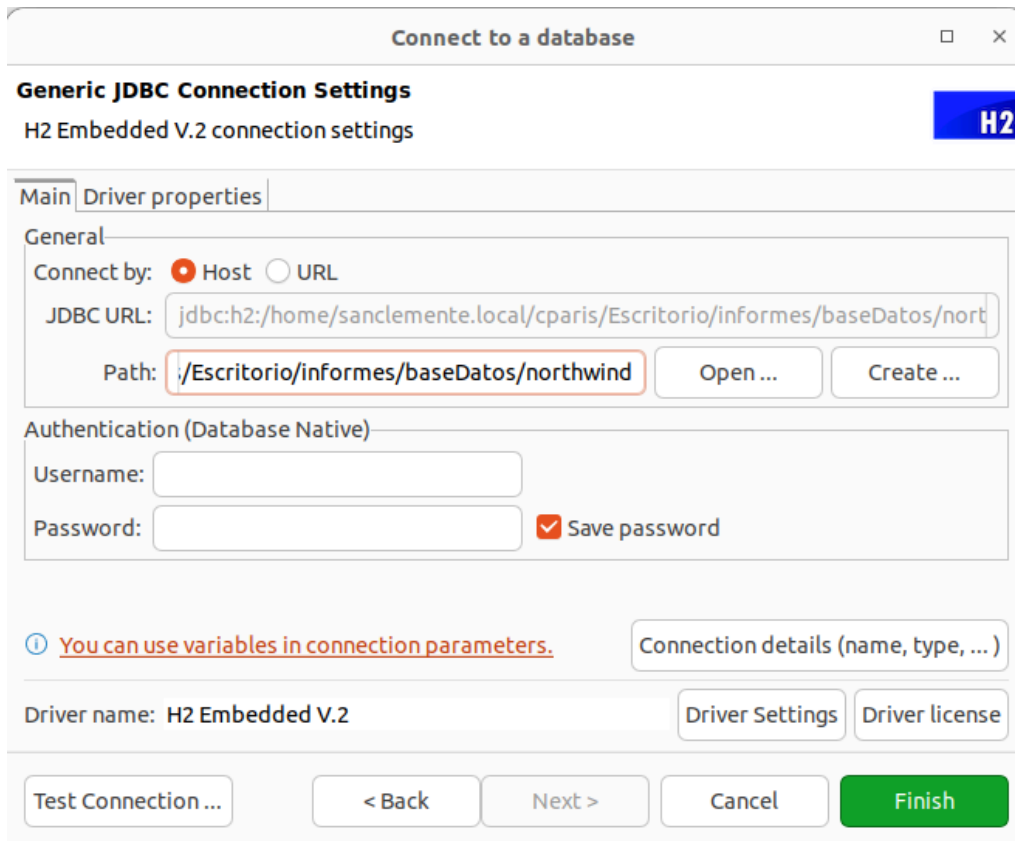
Creación da base de datos

Os informes son elaborados en función dos datos almacenados nunha orixe de datos. Neste documento vaise mostrar como xerar un informe cos datos almacenados nunha base de datos.

Pode utilizarse calquera base de datos, mais que neste caso vaise crear unha base de datos H2 utilizando DBeaver

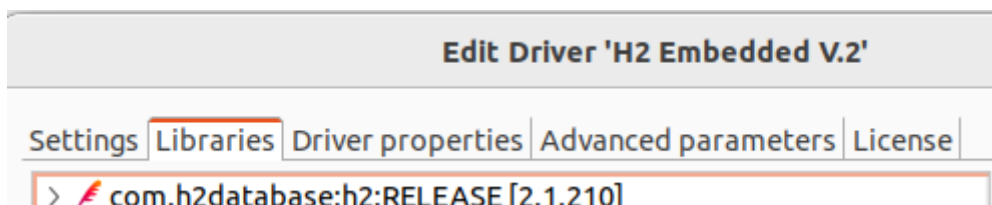
Para crear a base de datos utilizarase a ferramenta DBeaver e pulsarase na icona para crear unha nova conexión  .

Na nova ventá que se abra seleccionárase a base de datos **H2 Embedded V.2** e pulsarase o botón “Create” para crear unha nova base de datos. Na seguinte imaxe pode observarse a configuración para crear unha base de datos chamada northwind na carpeta informes do repositorio do módulo.

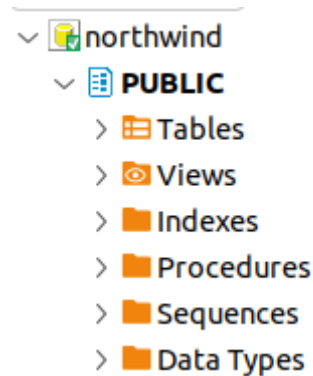


NOTA: pode que sexa necesario descargar algún driver para a creación da base de datos.

Usar a versión 2.1.210 do driver:



Unha vez creada a base de datos e establecida a conexión, poderase administrar dende DBeaver:

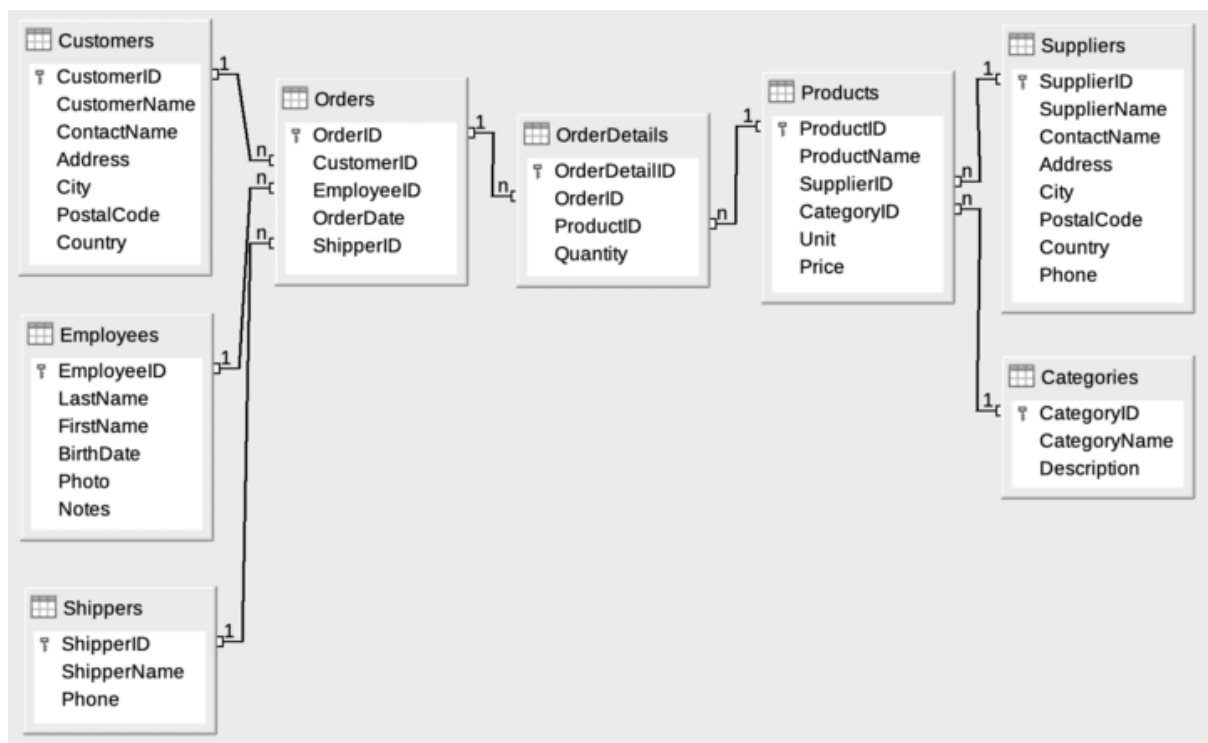


A continuación crearase a estrutura da base de datos e insertaranse os datos utilizando o script “northwind.sql”. Descárgao da aula virtual e execútao na base de datos recién creada.

Para xerar os informes usarase como exemplo a base de datos **Northwind**, que é unha base de datos usada en multitude de tutoriais durante anos.

A base de datos [Northwind](#) contén as vendas dunha compañía ficticia chamada “Northwind Trades”, que importa e exporta comida arredor do mundo. A base de datos contén clientes, pedidos, compras, provedores, empregados, etc.

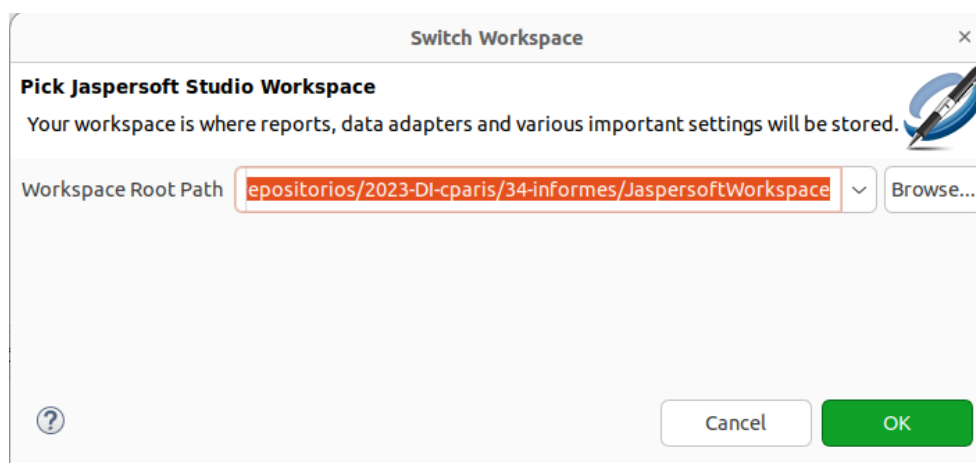
O diagrama da base de datos Northwind pode observarse na seguinte imaxe:



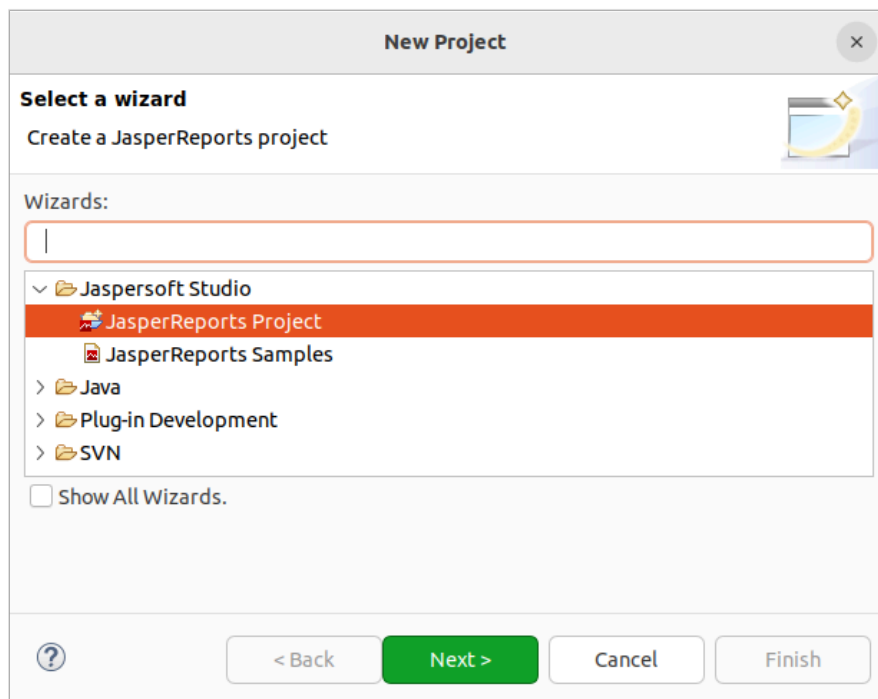
Creación dun proxecto JasperReports

Dende a aplicación Jaspersoft Studio vaise crear un novo proxecto para os informes. Mais antes de crealo, configurarase a carpeta onde se van gardar os informes. Para iso seleccionárase o menú File -> Switch Workspace.

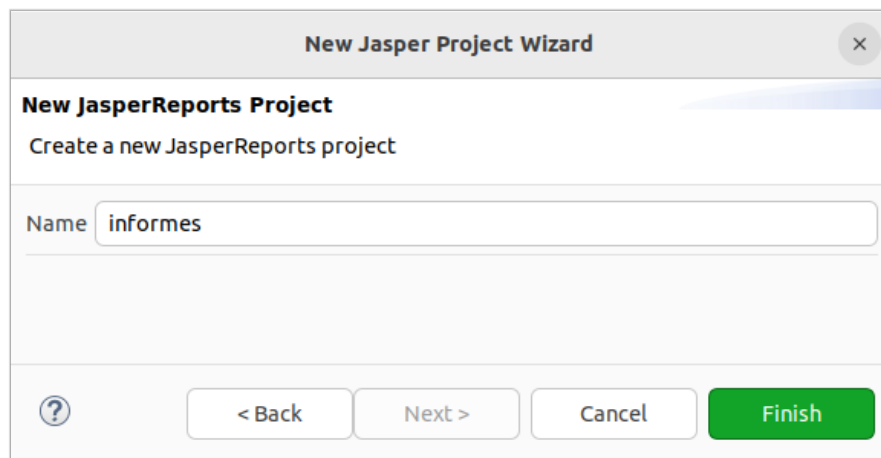
Crearase unha carpeta denominada JaspersoftWorkspace no repositorio do módulo e seleccionárase esta carpeta como workspace de Jaspersoft Studio.



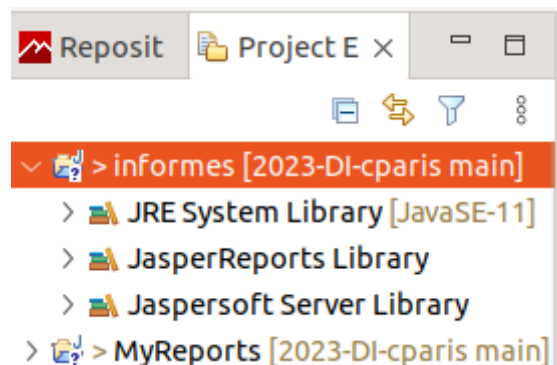
A continuación, para crear un novo proxecto seleccionárase o menú File -> New -> Project... Abrirase unha nova ventá onde se pode escoller o tipo de proxecto. Neste caso seleccionárase “JasperReports Project”.



Na imaxe anterior pulsa “Next”, para a continuación asignarlle un nome ao proxecto e pulsar “Finish”.



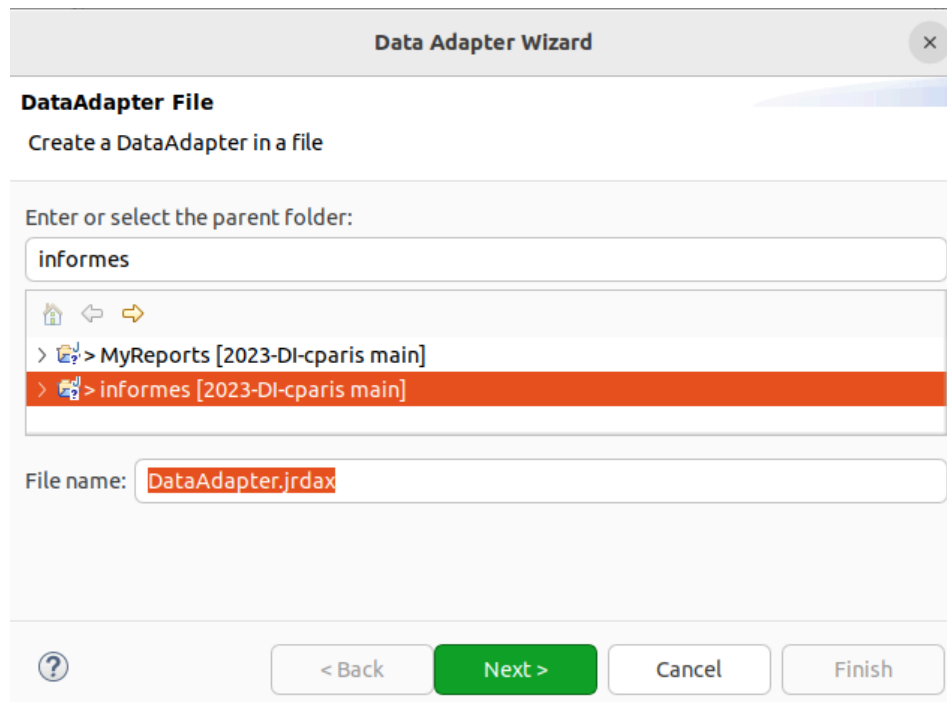
Unha vez creado o proxecto debería verse na pestana “Project Explorer”.



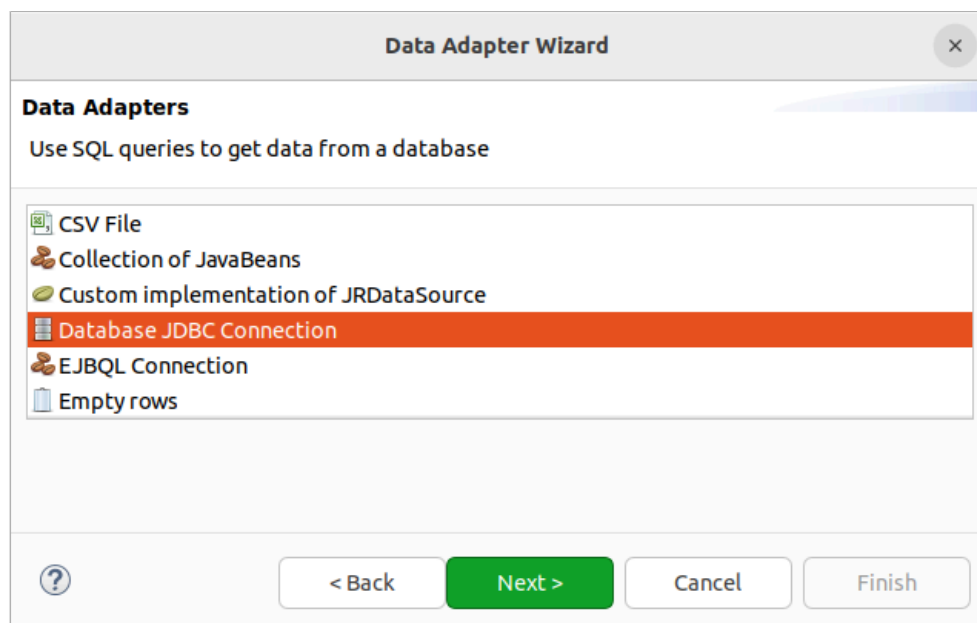
Conexión coa base de datos

Jasper Studio pode conectarse con distintas fontes de datos: base de datos, ficheiros csv, excel ou json, base de datos MongoDB, etc. Neste exemplo vaise utilizar a base de datos H2 creada en apartados anteriores. Esta conexión é necesaria para facilitar a creación do infome e para probalo directamente desde Jaspersoft Studio.

Pulsa "File" -> "new" -> "Data Adapter". Sairá unha ventá donde debes escoller de que proxecto é esta fonte de datos e un nome para o ficheiro onde se gardarán os parámetros de conexión. Selecciona o teu proxecto "informes" e podes deixar o nome do ficheiro por defecto "DataAdapter.jrdax".



Pulsa "Next >" e sairá unha ventá para escoller o tipo de fonte de datos, onde se escollerá "Database JDBC Connection" para, a continuación pulsar "Next >".



Na seguinte imaxe que aparecerá haberá que configurar a conexión á base de datos. Observar na imaxe a configuración seleccionada:

The screenshot shows the 'Data Adapter Wizard' dialog box with the 'Database JDBC Connection' tab selected. The configuration is as follows:

- Name:** informes
- JDBC Driver:** H2 (file) (org.h2.Driver)
- JDBC Url:** >me/sanclemente.local/cparis/Escritorio/informes/baseDatos/northwind
- Username:** (empty)
- Password:** (empty)

Below the fields, there is a warning: **Attention!** Passwords are saved in clear text.

At the bottom, there are three tabs: Database Location, Connection Properties, and Driver Classpath. The 'Database Location' tab is currently active.

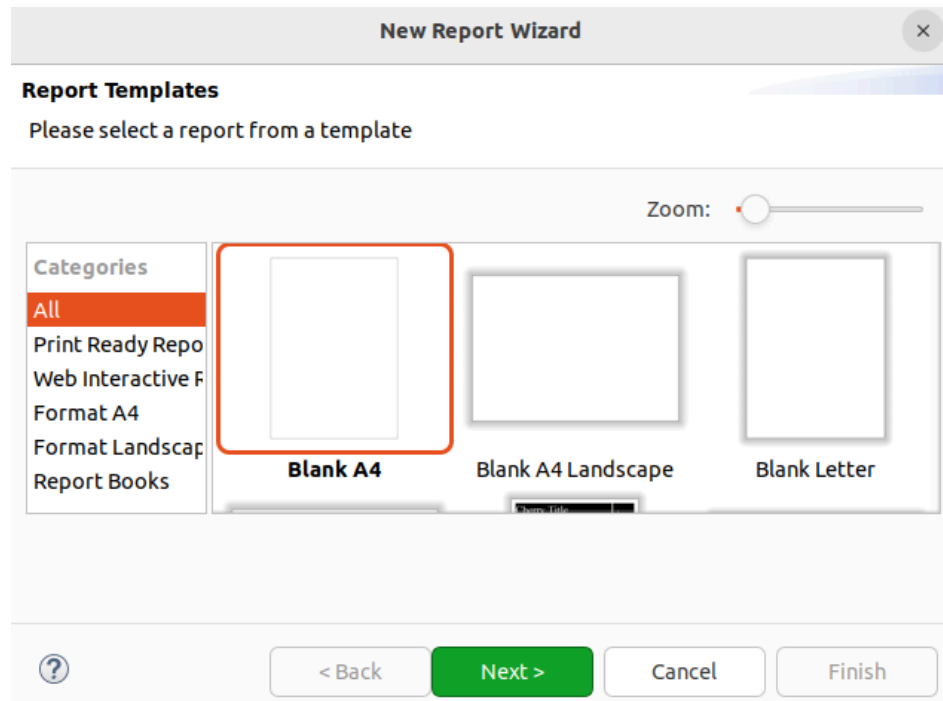
At the very bottom, there are five buttons: a help icon (?), Test, < Back, Next >, Cancel, and a green Finish button.

Pode pulsarse o botón “Test” para comprobar a conexión á base de datos. Se todo é correcto, pulsarase o botón “Finish”.

Observar que se engadiu un novo ficheiro ao proxecto de Jaspersoft Studio chamado “DataAdapter.jrdax”

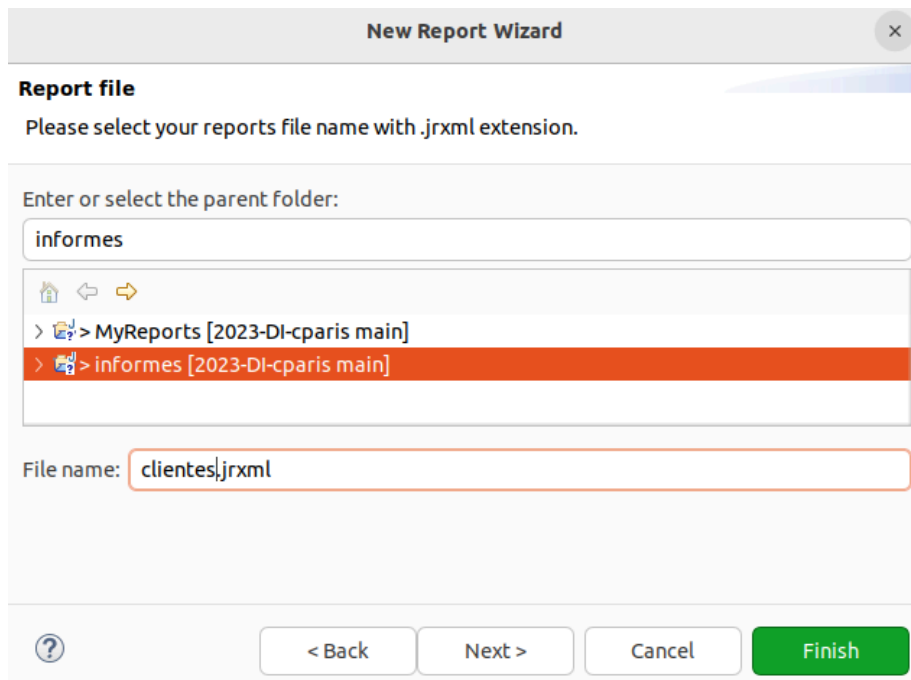
Crear un informe

Para crear un novo informe seleccionárase o menú "File" -> "new" -> "Jasper Report" e aparecerá unha ventá cos modelos de informes dispoñibles:



Pode escollerse o modelo que mellor se adapte aos requerimentos do informe a realizar. Neste caso, para aprender a crear informes escollerase un modelo en branco para crear todas as partes do informe paso a paso. Seleccionar o modelo "Blank A4" e pulsar Next.

A seguinte imaxe pregunta en que proxecto gardar o informe e o nome do mesmo. Cubir os campos como na seguinte imaxe. O nome do informe será “clientes.jrxml”. A continuación pulsar “**Next**”.



New Report Wizard

Report file

Please select your reports file name with .jrxml extension.

Enter or select the parent folder:

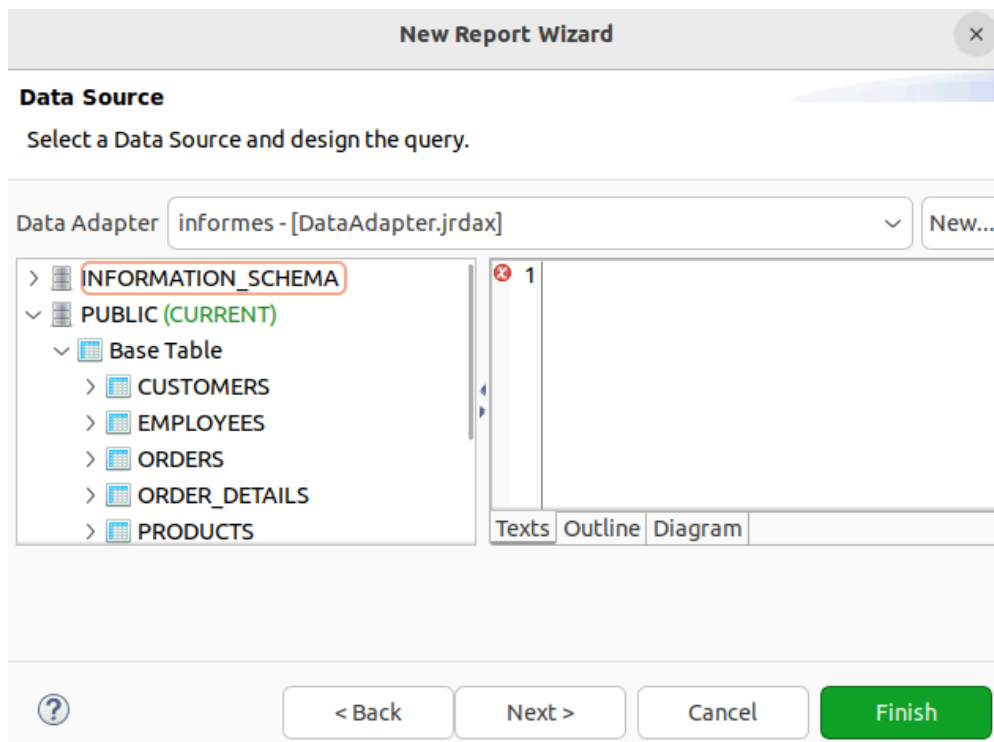
informes

- > MyReports [2023-DI-cparis main]
- > informes [2023-DI-cparis main]

File name: clientes.jrxml

? < Back Next > Cancel Finish

Na seguinte ventá, na parte superior, pode seleccionarse unha das fontes de datos (Data Source) que estean configuradas. Neste caso “informes - [DataAdapter.jrdax]”, que foi a configurada en pasos anteriores. Nada mais escollela, a ventá conéctase coa base de datos e mostra a súa información. Navegando pola árbore, poden verse as táboas:



New Report Wizard

Data Source

Select a Data Source and design the query.

Data Adapter informes - [DataAdapter.jrdax] New...

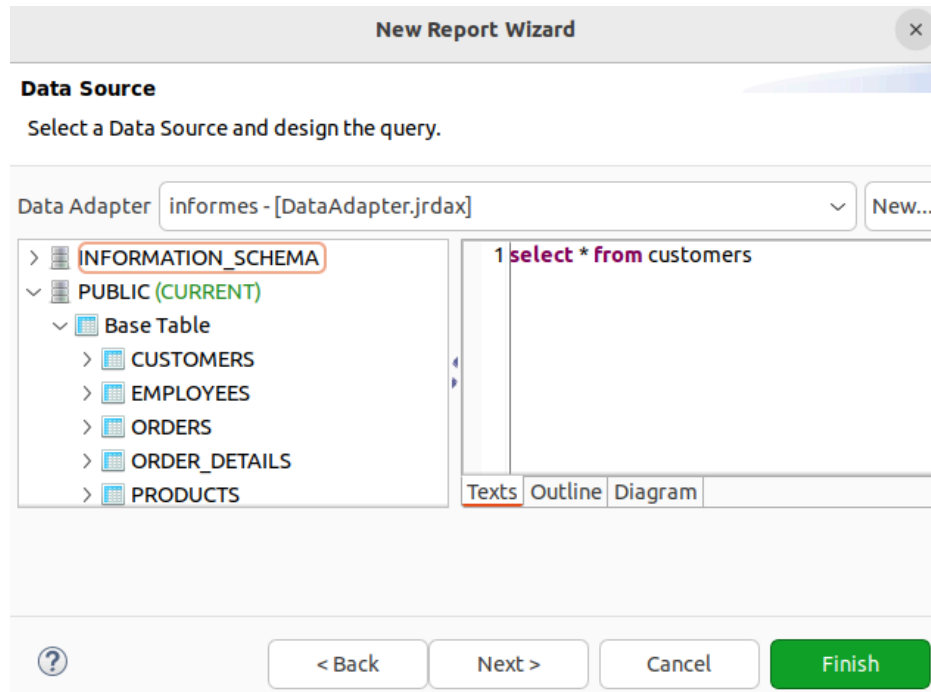
- > INFORMATION_SCHEMA
- > PUBLIC (CURRENT)
 - > Base Table
 - > CUSTOMERS
 - > EMPLOYEES
 - > ORDERS
 - > ORDER_DETAILS
 - > PRODUCTS

1

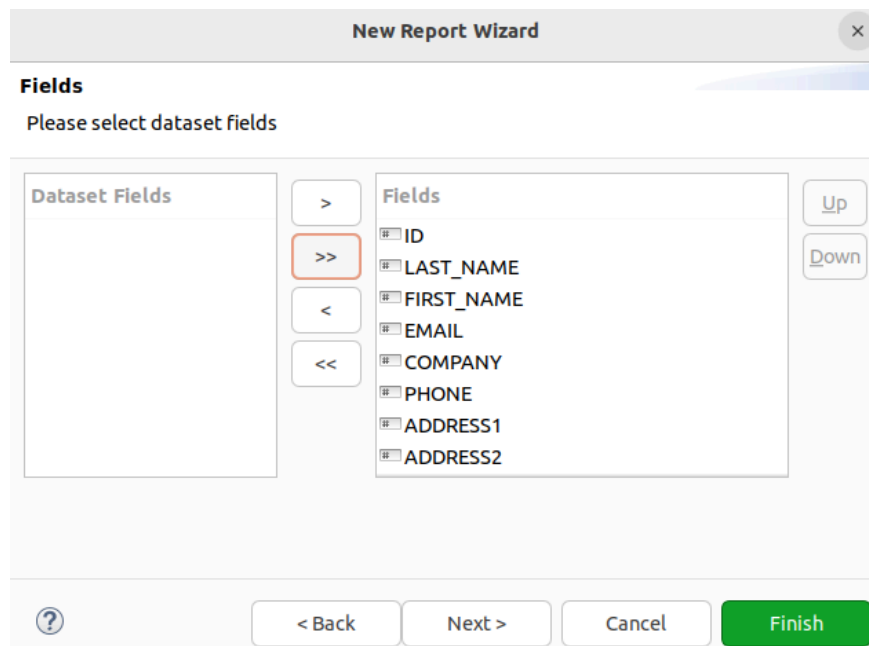
Texts Outline Diagram

? < Back Next > Cancel Finish

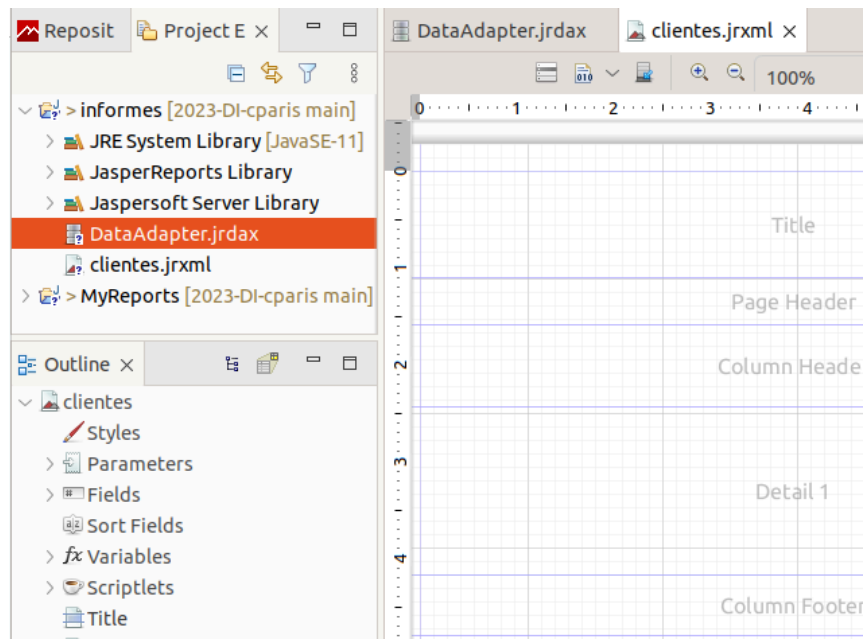
Ademais de seleccionar a base de datos, pode escribirse unha consulta para configurar que campos da base de datos estarán dispoñibles no informe. No exemplo que se está realizando quere facerse un informe dos clientes, polo que se necesitarán mostrar os campos da táboa “Customers”. Para iso escribirase a seguinte sentencia sql e pulsarase o botón Next>:



Para que todos os campos da consulta estean dispoñibles para o informe, haberá que engadilos á parte dereita da seguinte imaxe pulsando o botón “>>”:

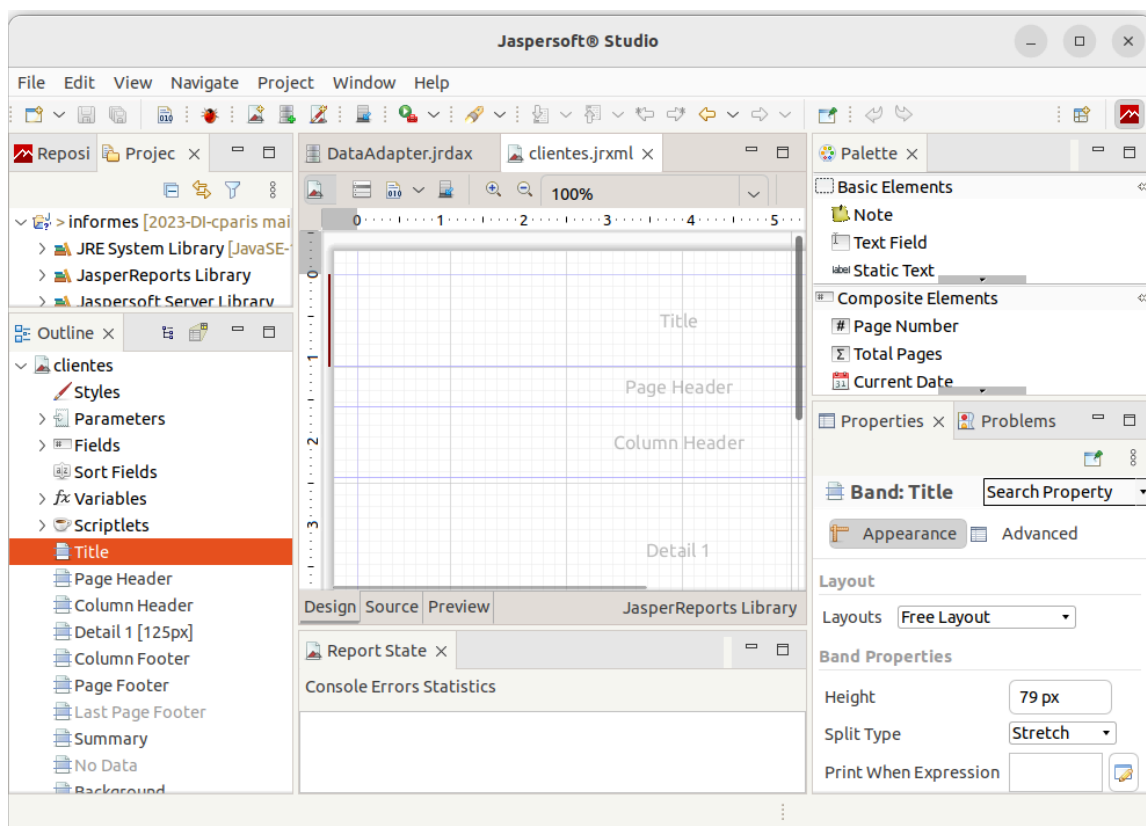


Unha vez pulsado o botón **Finish**, creárase o informe:



Interface Jaspersoft Studio

Cando se crea un novo informe, vese unha interface similar á seguinte:



A continuación descríbense os elementos da interface:

- **Outline:** mostra a estrutura completa do informe, que se compón de obxectos (campos, parámetros e variables), bandas (que son as seccións do documento) e elementos (tales como campos de texto, imaxes ou gráficos).
- **Informe:** a parte central da imaxe contén o informe, que pode mostrarse en modo “Design”, “Source” ou “Preview”. O modo “Design” permite deseñar de forma visual o informe arrastrando, colocando, aliñando e cambiando o tamaño dos elementos do informe. Cando se pulsa en “Preview”, execútase o informe coa conexión activa e móstrase o resultado utilizando un visor propio
- **Paleta:** contén os elementos de deseño que poden ser arrastrados dentro dunha banda para mostrar os datos.
- **Propiedades:** utilízase para establecer as propiedades do compoñente seleccionado no informe.
- **Report State:** contén a listaxe de erros encontrados ao compilar o informe.

Elementos estruturais dun informe

Os informes creados están formados polos seguintes elementos estruturais:

- **Título:** só aparece na primeira páxina do informe.
- **Cabeceira de páxina:** aparece na parte superior de cada páxina. Normalmente aquí colócase información como a data e hora, nome da organización, etc.
- **Cabeceira de columna:** úsase para listar os nomes dos campos que se van presentar.
- **Detalle:** nesta sección irán os valores correspondentes ás entradas de campos definidas no apartado anterior.
- **Pé de páxina:** aparece na parte inferior de cada páxina. Nesta parte pódese incluír, entre outras cousas, un contador de páxinas.
- **Resumo:** esta sección úsase para proporcionar información resumida dos campos presentes na sección “Detalle”. Só aparece na última páxina do informe.

Cando se diseña un informe decídese cales dos elementos estruturais debe incluír. Se non é necesaria unha banda, para eliminala recoméndase acceder ao **Outline** -> pulsar co botón dereito sobre a banda innecesaria -> Eliminar.

Creación dun informe sinxelo

Un informe mostra datos almacenados nunha orixe de datos, no noso caso, unha base de datos. Polo tanto, para crear un informe o primeiro será escribir a consulta SQL para obter eses datos.

Unha vez definida a consulta, e despois de pulsar o botón “OK” aparecen no “Report Inspector” os campos recuperados da base de datos.



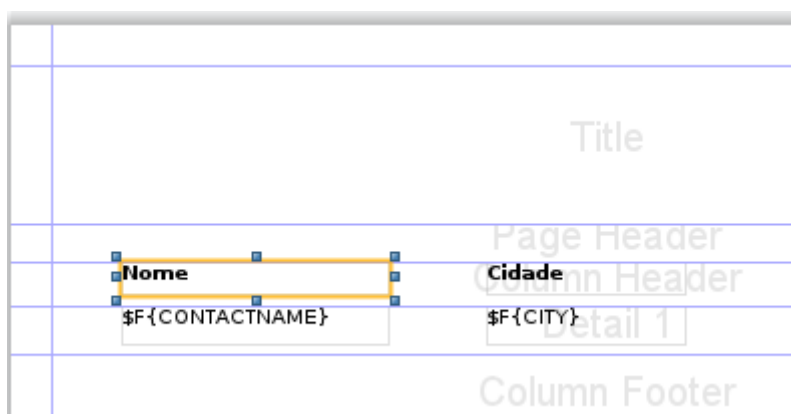
Para que calquera dos campos anteriores aparezan no informe, basta con arrastralos á banda adecuada.

Recordar que o informe está definido en bandas. Dependendo de en que banda se coloque un elemento, terá un comportamento ou outro.

Os elementos sobre os que se queira iterar hai que colocalos na banda de **detalle**.

Crear un informe como o da imaxe cos campos **contactName** e **city**.

Modificar tamén o alto das bandas, traducir a cabeceira a galego e poñela en letra grosa.

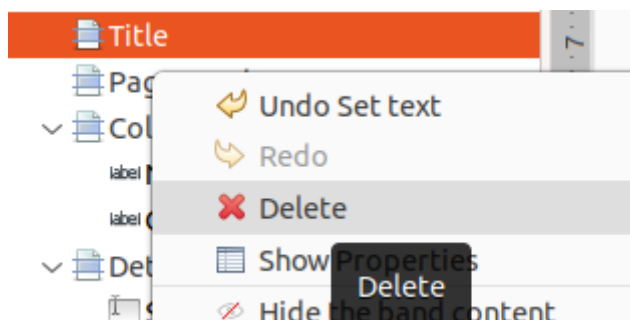


Unha vez deseñado o informe, pode pulsarse o botón “Preview” para velo cos datos da orixe de datos.

Nome	Cidade
Maria Anders	Berlin
Ana Trujillo	México D.F.
Antonio Moreno	México D.F.
Thomas Hardy	London

Fixarse que queda un espazo ao comezo do informe que é o reservado para o título e a cabeceira da páxina.

As bandas que non se necesitan dun informe poden eliminarse. Para facelo recoméndase ir ao inspector do informe, seleccionar unha banda -> pulsar co botón dereito sobre ela -> eliminála.



Exercicio:

1. Crea un informe para mostrar os produtos almacenados na base de datos. O informe debe mostrar o nome do produto, as unidades e o prezo.
2. Crea un informe para mostrar os datos dos subministradores (suppliers). O informe debe mostrar o nome do subministrador, o nome da persoa de contacto, a cidade e o país.

Paso de parámetros a un informe

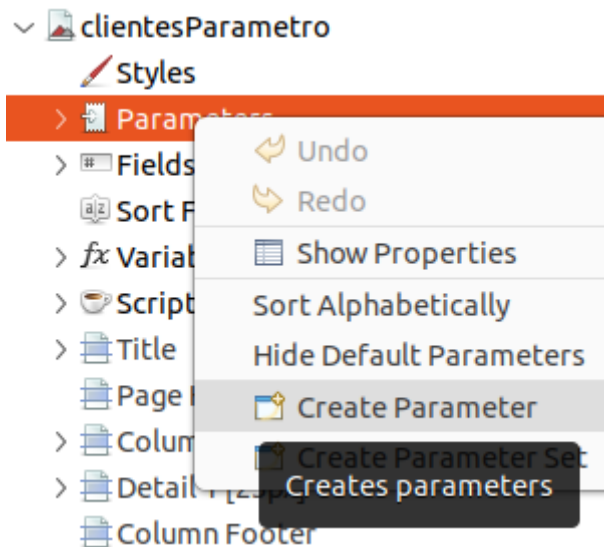
Un informe mostra os datos dunha orixe de datos, no noso caso unha base de datos. Tal e como se deseñaron os informes ata agora, estes sempre amosan os mesmos datos, salvo que haxa modificacións na base de datos. Sería interesante crear un informe, por exemplo, que amose os clientes dunha determinada cidade pasada como parámetro.

Configurar un informe con un parámetro permitiría executalo dende unha aplicación de escritorio. Por exemplo, pode deseñarse un Frame con un ComboBox coa lista de cidades e pasar ao informe a cidade seleccionada, para que se elabore cos datos dos clientes desa cidade.

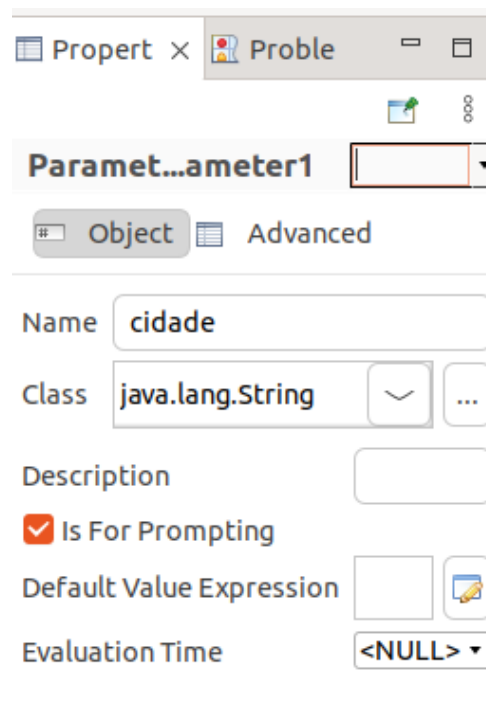
Vaise copiar o informe de clientes elaborado en pasos anteriores co nome clientesParametro. O novo informe creado vaise modificar para engadir un parámetro "cidade" para que amose só os clientes da cidade pasada como parámetro.

No **Outline**, se se desprega o obxecto **Parameters** poden verse os parámetros que están definidos no informe.

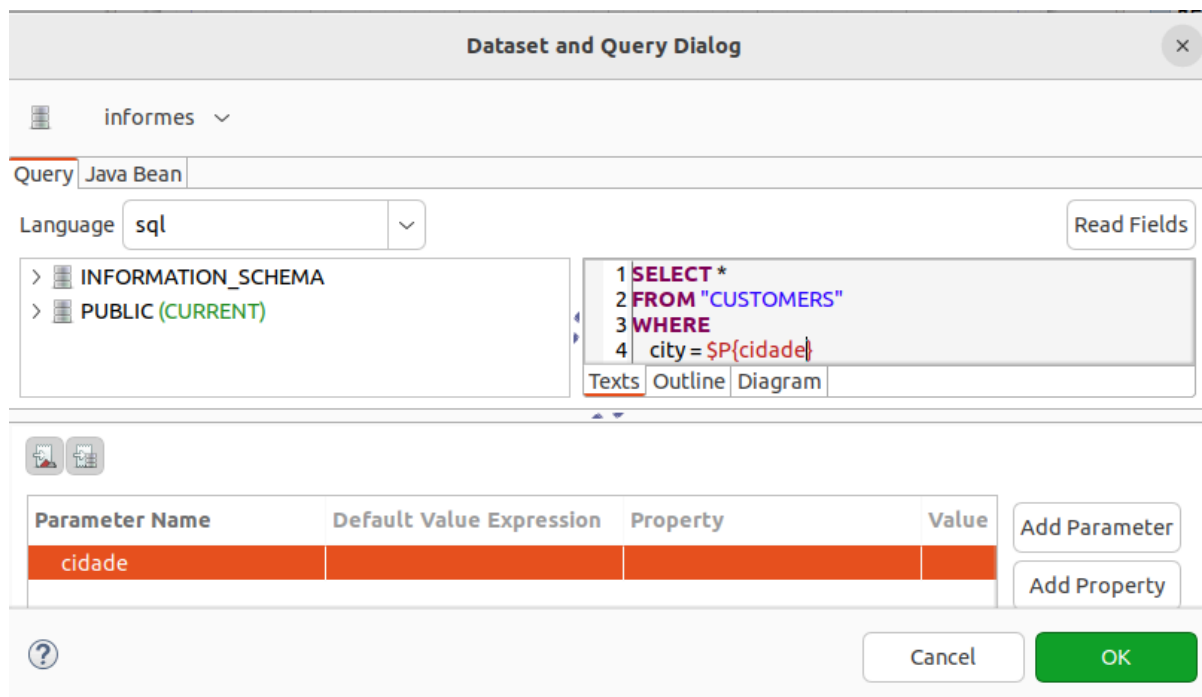
Neste caso vaise crear un novo parámetro pulsando co botón dereito do rato sobre **Parameters** -> **Create Parameter**.



Configurar como nome do parámetro **cidade** e tipo **String**. O campo “Is For Prompting” úsase para cando se pulse en “Preview” pida un valor para o parámetro e mostre só os clientes desa cidade.



Agora hai que modificar a consulta, xa que actualmente amosa todos os clientes. Haberá que incluír a cláusula **where**. Para iso acceder ás propiedades do informe e pulsar “Edit query, filter and sort options” e modificar a consulta como se indica na seguinte imaxe:



Pode modificarse un pouco o informe e engadir na banda de título un **texto estático** que sexa “Clientes da cidade” e ao lado incluír o parámetro. Modificar o tamaño de letra destes dous obxectos establecendo un tamaño 14 e eliminar a banda **Page Header**.

Clientes da cidade		\$P{cidade}	Title
Nome		Cidade	Column Header
\$F{CONTACTNAME}		\$F{CITY}	Detail 1
			Column Footer

Probar a visualizar o informe. Comprobar que solicita o nome da cidade e despois mostra os clientes da cidade introducida como parámetro.

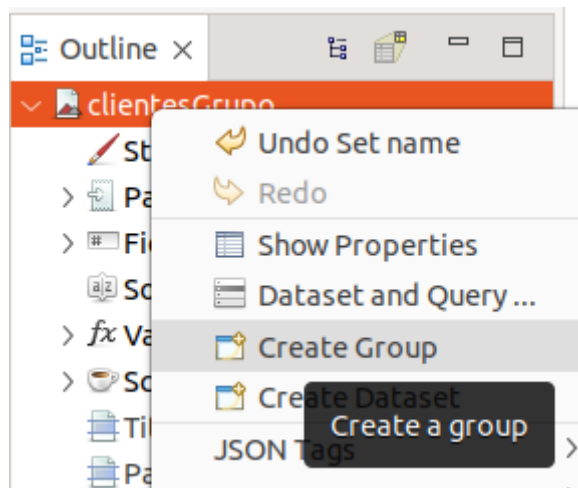
Exercicios:

1. Partindo do informe dos produtos, crea un novo informe para incluír un parámetro que permita sacar un informe cos produtos dos que o precio sexa menor que o parámetro dado. Inclúe na cabeceira información do prezo máximo dos produtos.
2. Modifica o informe dos subministradores (suppliers) para incluír un parámetro que sexa o país, de tal forma que só se mostren aqueles do país seleccionado. Inclúe na cabeceira información do país seleccionado.

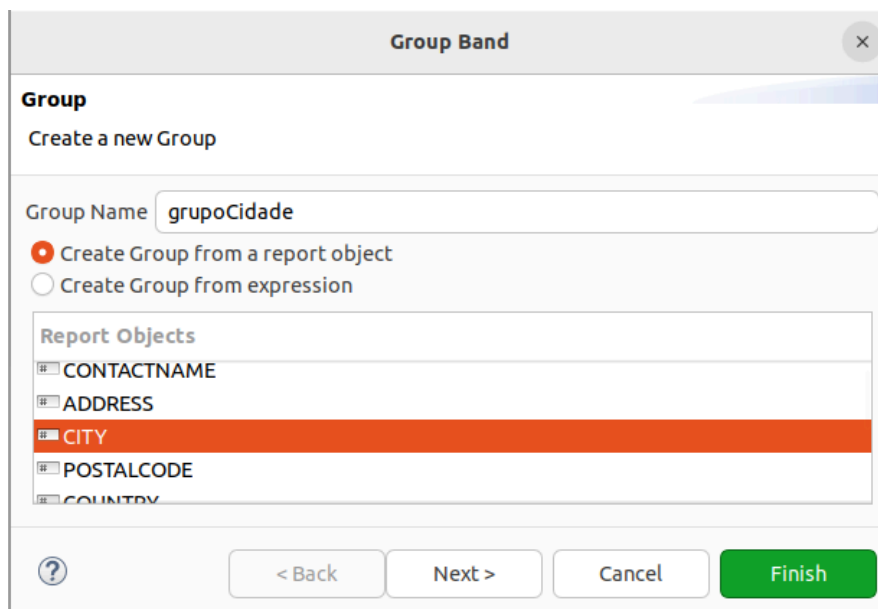
Crear unha agrupación nun informe

Unha agrupación nun informe permite agrupar os datos por un campo. Por exemplo, no caso dos clientes, imaxinar que se queren mostrar os clientes agrupados por cidade, de tal forma que os clientes dunha cidade aparezan xuntos no informe.

Facer unha copia do informe de clientes denominada **clientesGrupo**. Á copia feita vaise engadirlle un grupo. Para iso hai que pulsar co botón dereito do rato sobre **nome do informe** -> **Create Group**.



Asignar un nome ao grupo e indicar que se vai agrupar polo obxecto **City**.



A continuación pregunta se se queren engadir as bandas correspondentes ao grupo no informe. En principio vanse engadir. Estas bandas permitirán engadir información relativa ao grupo.

Group Bands

Group layout

Please select what bands should we create

☒ Add the Group Header

☒ Add the Group Footer

No informe, vaise engadir o campo (**Field**) **City** á banda cabeceira do grupo. Cando se engade sae a seguinte pantalla preguntando cal é o valor que se quere mostrar. Neste caso vaise escoller “No Calculation Function”.

TextField Wizard

TextField Wizard

Please select what type of calculation you want

No Calculation Function

Count

Distinct Count

O paso anterior engadiría o nome da cidade á cabeceira do grupo. Se o que se quere é que mostre a mensaxe “**Cientes de** ” + **#{CITY}**, hai que editar a expresión pulsando co botón dereito sobre o campo que se acaba de engadir -> **Edit expression**. A sintaxe da expresión que escrita é equivalente á concatenación de cadeas en Java.

Expression Editor (java)

Expression Editor

Edit the JasperReports expression that returns the expected type.

"Cientes de " + \$F{CITY}

Parameters

Fields

Variables

Resources

Built-in Functions

type filter text

CUSTOMERID Field Integer

CUSTOMERNAME Field String

CONTACTNAME Field String

type filter text

bitCount(int) int

byteValue() byte

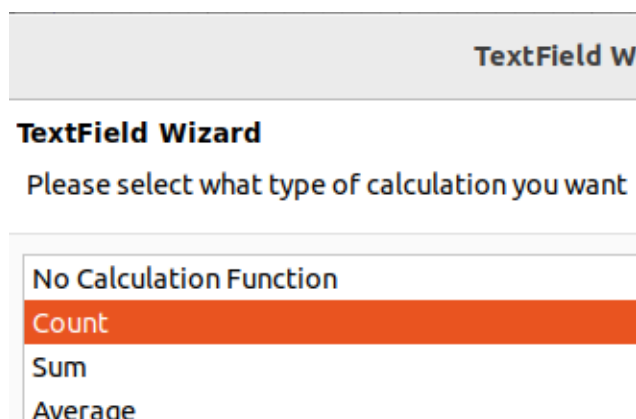
compare(int, int)

18

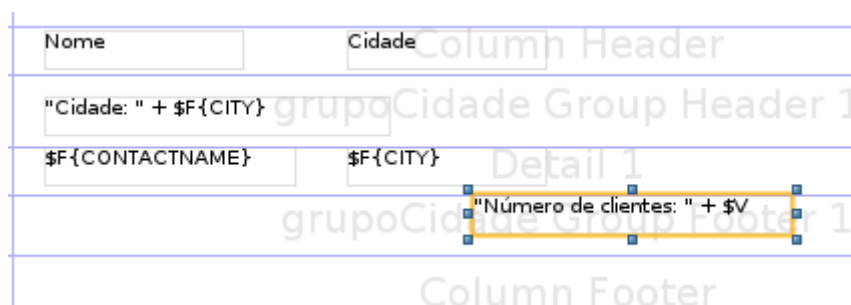
Se se previsualiza o informe comprobarase que non funciona, porque a sentencia sql non trae os datos agrupados ao informe.

Para solucionar este problema haberá que modificar a consulta por esta outra: “**select * from customers order by city**”.

Pode ser interesante tamén incluír unha mensaxe que mostre o número de clientes por cidade. Para facelo, vaise arrastrar o campo (Field) **CustomerId** ao pé do grupo Cidade. Ao arrastralo aparece a seguinte imaxe, onde se seleccionará a función Count para que conte o número de clientes.



Tamén se pode editar a expresión para que informe do valor do dato.



Se se comete algún erro ao crear unha variable, para eliminala non serve con eliminala do deseño do informe. Hai que acceder ao **Outline**, na sección de variables e eliminala.

Exercicio:

1. Modifica o informe dos subministradores (suppliers) para que saian agrupados por país. Inclúe na cabeceira do grupo o nome do país e no pé do grupo o número de subministradores por país.

Realiza a configuración necesaria para que todos os subministradores do mesmo país saian na mesma páxina.

2. Crea un novo informe que mostre o nome e cidade do cliente. Ademais, agrupa a información por país. Inclúe na cabeceira do grupo o nome do país e no pé do grupo o número de clientes do país.

3. Crea un novo informe que mostre os clientes agrupados por país e, para cada país se agrupen por cidade. Engade no pé do grupo o número de clientes totais por país.

Engade tamén no pé de páxina o número de páxina actual e o total de páxinas. Así, debería mostrarse “1 de 6”, “2 de 6”, “3 de 6”, ...

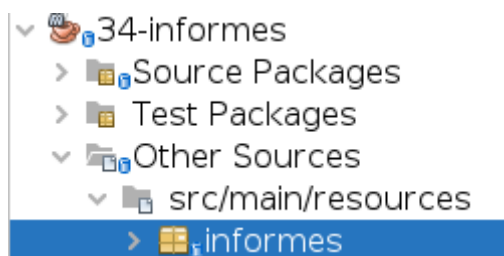
Integración dun informe nunha aplicación Java

Partindo dun informe previamente deseñado onde se mostran os clientes dunha cidade pasada como parámetro, vaise integrar nunha aplicación Java que teña un comboBox coa lista das cidades. Cando a persoa usuaria seleccione unha cidade e pulse o botón aceptar, debe xerarse o informe da cidade seleccionada.

Neste caso vaise partir dun informe similar ao da seguinte imaxe que ten un parámetro chamado cidade:

Clientes da cidade		\$P{cidade}
Nome	Cidade	
\$F{CONTACTNAME}	\$F{CITY}	

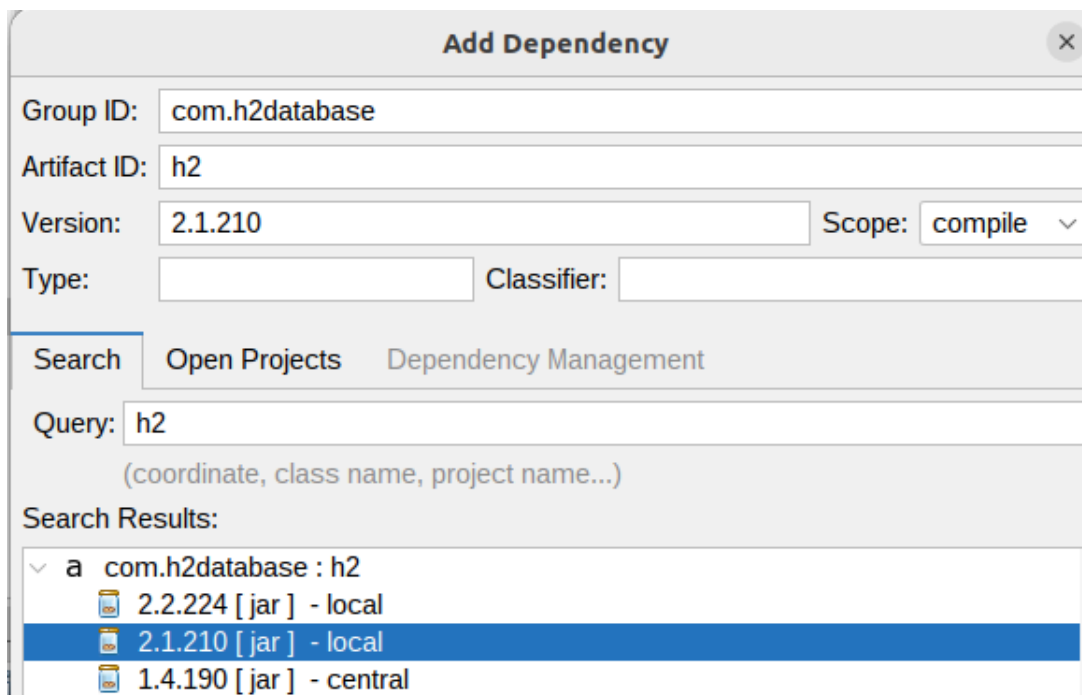
A continuación haberá que crear unha aplicación Java con Maven dende NetBeans. Recordar a estrutura de carpetas dun proxecto Java, que se mostra na seguinte imaxe. Para evitar problemas coas rutas ao acceder aos informes, estes almacenaranse na carpeta informes.



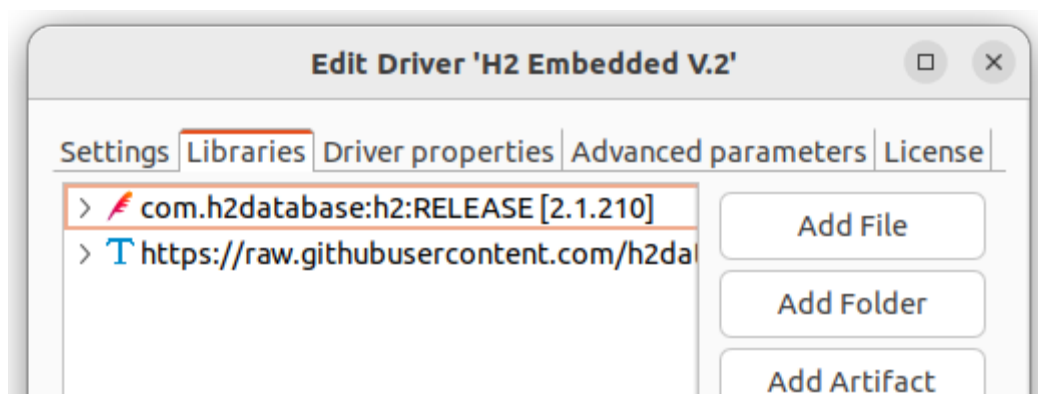
Hai que copiar á carpeta informes do proxecto o informe compilado, o que ten extensión **.jasper**. Recordar que o código fonte dun informe almacénase no ficheiro con extensión **.jrxml**

O seguinte que hai que facer é engadir a dependencia ao proxecto para poder usar a base de datos H2. Se a base de datos fose doutro tipo, habería que engadir a librería adecuada.

Pulsar co botón dereito do rato sobre **Dependencies** -> **Add Dependency**. Na nova ventá que se abra escribir na caixa Search -> query: “h2” e seleccionar a versión que se corresponda coa versión da base de datos creada dende DBeaver. Na imaxe utilizarase a versión 2.1.210.



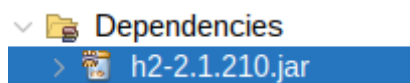
A seguinte imaxe mostra que a versión do driver de H2 en DBeaver é a 2.1.210.



Se en NetBeans non aparece a versión concreta da dependencia, pode engadirse directamente o seguinte código [extraído do repositorio de maven](#) ao ficheiro pom.xml:

```
<!-- https://mvnrepository.com/artifact/com.h2database/h2 -->
<dependency>
  <groupId>com.h2database</groupId>
  <artifactId>h2</artifactId>
  <version>2.1.210</version>
</dependency>
```

Pode ser tamén necesario descargar a dependencia no proxecto. Unha vez descargada, debería verse a seguinte imaxe do proxecto:



A continuación crearase un formulario no proxecto similar ao da seguinte imaxe. Incluír no comboBox algunhas cidades para probar que funciona. O ideal sería acceder á base de datos e completar as cidades cos datos almacenados na base de datos.

 A screenshot of a web form. It features a light gray rectangular container. Inside, on the left, is a dropdown menu with a white background and a gray border, containing the text 'Berlin' and a small downward arrow. To the right of the dropdown is a button with a gray border and the text 'Aceptar'.

O código a executar cando se pulsa o botón **Aceptar** é o seguinte:

```
Connection conexion;
try {

    String url = "jdbc:h2:/media/DATOS/repositorios/2023-DI-cparis/informes/northwind";
    conexion = DriverManager.getConnection(url);

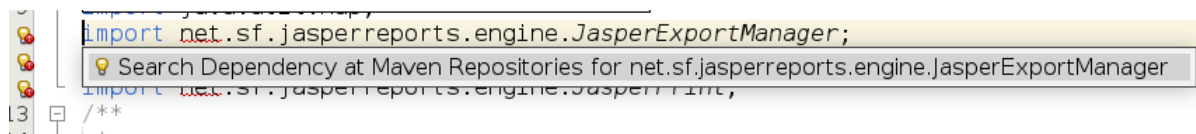
    Map parametros = new HashMap();
    parametros.put("cidade", jComboBoxCidade.getSelectedItem());
    JasperPrint print
        = JasperFillManager.fillReport("src/main/resources/informes/clientesParametro.jasper",
            parametros,
            conexion);

    // exportar a PDF
    JasperExportManager.exportReportToPdfFile(print, "src/main/resources/informes/informe.pdf");
    System.out.println("PDF creado -----");
} catch (Throwable e) {
    e.printStackTrace();
}
```

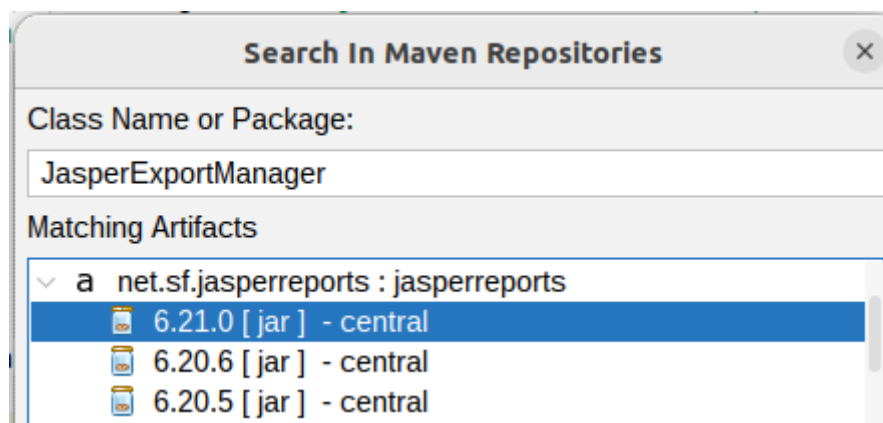
Fixarse que todo o código está nun bloque try-catch para capturar posibles excepcións que se produzan durante a execución.

Fixarse tamén que o parámetro se pasa nun mapa. Os mapas Java permiten almacenar pares chave-valor. Neste caso a chave é o nome do parámetro (**ten que coincidir exactamente co do informe**, incluídas maiúsculas e minúsculas) e o valor é o valor do parámetro.

Pode ser necesario engadir as dependencias e os imports de forma manual ao proxecto. Para iso pulsárase sobre o ícone do lateral da liña para buscar as dependencias no repositorio de Maven.



E incluírase a versión 6.21.0:



Se non aparece, intentar buscala e engadila manualmente (Add dependency).

Xa pode probarse e executar a aplicación. Comprobar que se crea un pdf co informe na carpeta informes.

Exercicio:

1. Crea unha aplicación gráfica Java que pida un número, de tal forma que se cree o informe con aqueles produtos con precio menor que o número indicado como parámetro.
2. Crea unha aplicación gráfica Java que permita elaborar o informe dos subministradores (suppliers) pasando o país como parámetro. A lista de países debe lerse da base de datos e cargalos nun comboBox, para que a persoa usuaria seleccione aquel do que queira elaborar o informe.

Subinformes

Un subinforme permite meter un informe dentro doutro informe. É útil porque permite mostrar os datos de dúas consultas á orixe de datos. Por exemplo, cando se queren mostrar os datos dun pedido e os datos do cliente do pedido é necesario facer dúas consultas á base de datos. Primeiro consultar o pedido e despois consultar o cliente.

Para exemplificar a creación de subinformes vaise facer un informe que permita mostrar as liñas dun pedido pasado como parámetro e o subinforme mostrará os datos do cliente.

Primeiramente crearase un informe chamado **informePrincipal** para mostrar os datos do pedido e pulsarase o botón **Next**.

Report file

Please select your reports file name with .jrxml extension.

Enter or select the parent folder:

informes

> MyReports [2023-DI-cparis main]
 > > informes [2023-DI-cparis main]

File name: informePrincipal.jrxml

? < Back Next > Cancel Finish

Seleccionar o Data Source e pulsar **Finish**. A consulta escribirase máis adiante:

Data Source

Select a Data Source and design the query.

Data Adapter informes - [DataAdapter.jrdax] New...

> INFORMATION_SCHEMA
 > PUBLIC (CURRENT)

1

Texts Outline Diagram

? < Back Next > Cancel Finish

Para sacar os datos dun pedido, hai que facer unha sentencia SQL que involucre as táboas **orders**, **orderdetails** e **products**. Pode revisarse o diagrama da base de datos no apartado [Creación da base de datos](#).

Cada pedido terá unha factura, polo que se vai crear un parámetro **idFactura** (de tipo integer) para pasarlle ao informe o identificador da factura a mostrar..

The screenshot shows the 'Properties' window for a parameter named 'idFactura'. The 'Object' tab is selected. The 'Name' field contains 'idFactura'. The 'Class' field is set to 'java.lang.Integer'. The 'Description' field is empty. The 'Is For Prompting' checkbox is checked. The 'Default Value Expression' field is empty. The 'Evaluation Time' dropdown is set to '<NULL>'. There is a 'Search Property' search bar at the top right.

A continuación xa se pode escribir a sentencia SQL a usar no informe. Nas propiedades do informe, pulsar “Edit query, filter and sort options” e escribir a seguinte consulta:

```
SELECT o.CUSTOMERID as idCliente, o.ORDERID, p.PRICE , d.QUANTITY, p.PRODUCTNAME
FROM orders o, orderdetails d, products p
WHERE o.ORDERID = d.ORDERID and d.PRODUCTID = p.PRODUCTID and o.orderid =
$P{idFactura}
```

The screenshot shows the 'Query' window with the 'Language' set to 'sql'. The query text is:


```
1 SELECT o.CUSTOMERID as idCliente, o.ORDERID, p.PRICE , d.QUANTITY, p.PRODUCTNAME
2 FROM orders o, orderdetails d, products p
3 WHERE o.ORDERID = d.ORDERID and d.PRODUCTID = p.PRODUCTID and o.orderid = $P{idFactura}
```

 Below the query, there is a table with the following columns: Field Name, Class Type, Description, Time zone, Column name, Column label, Column index, and Properties. The table contains five rows of data:

Field Name	Class Type	Description	Time zone	Column name	Column label	Column index	Properties
IDCLIENTE	java.lang.Integ						3 Properties
ORDERID	java.lang.Integ						3 Properties
PRICE	java.math.BigC						3 Properties
QUANTITY	java.lang.Integ						3 Properties
PRODUCTNAM	java.lang.Strin						3 Properties

 At the bottom right of the table are buttons for 'Add', 'Edit', 'Delete', and 'Up'. At the bottom of the window are 'Cancel' and 'OK' buttons.

NOTA: pulsar “Read Fields” para que se carguen os campos da base de datos.

Unha vez definida a instrución SQL, deseña un informe o máis parecido ao da seguinte imaxe:

"Número de factura: " + \$F{ORDERID}		
Page Header		
Descrición	Cantidade	Precio por unidade
Column Header		
\$F{PRODUCTNAME}	\$F{QUANTITY}	\$F{PRICE}
Detail 1		

NOTA: o texto “Número de factura” estableceuse editando a expresión do mesmo campo.

A continuación crearase un novo informe, nun ficheiro separado, que finalmente será incrustado no anterior. Para iso pulsárase no menú File -> Jasper Report -> escoller o modelo Blank A4 e poñerlle de nome **subinformeCliente**:

Report file

Please select your reports file name with .jrxml extension.

Enter or select the parent folder:

informes



> MyReports [2023-DI-cparis main]

> informes [2023-DI-cparis main]

File name: subinformeCliente.jrxml



< Back

Next >

Cancel

Finish

Pulsar o botón Next e escoller o mesmo adaptador que o do informe anterior. De momento non se escribirá a consulta, polo que pode pulsarse o botón **Finish**.

Data Source

Select a Data Source and design the query.

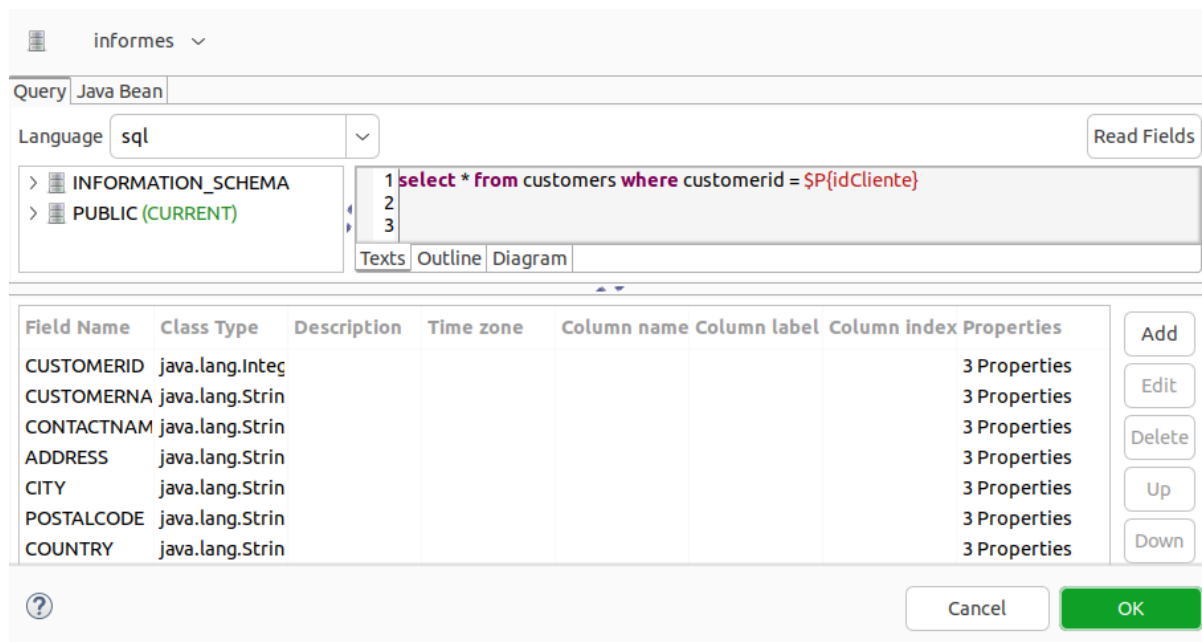
Eliminar todas as bandas do informe, excepto a banda “Detalle”.



No subinforme, é necesario crear un parámetro **idCliente**, de tipo integer, que fará referencia ao cliente.

A continuación hai que deseñar a consulta do subinforme para mostrar os datos do cliente. Observar que se pasa como parámetro o id do cliente que se quere mostrar:

```
select * from customers where customerid = $P{idCliente}
```





Nota: pulsar **Read Fields** para que se carguen as columnas da base de datos

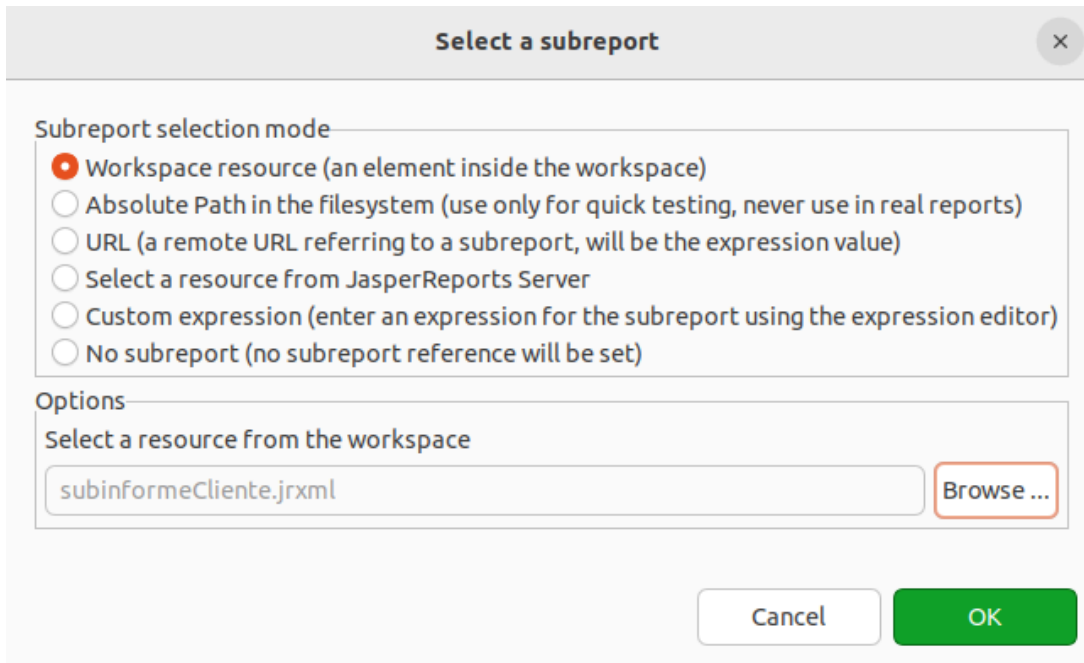
Deseñar un informe o máis parecido ao seguinte. Observar que se eliminaron todas as bandas que non son necesarias.



Pode probarse a facer a previsualización do subinforme e comprobar que todo funciona. Como o informe debe recibir un parámetro, para facer mostralo pregunta polo id do cliente.

A continuación vanse unir os dous informes. Para iso, no **informePrincipal** hai que engadir un elemento subinforme e colocalo na banda **Page Header**. Para iso só hai que arrastrar o obxecto  **Subreport** da paleta á banda correspondente.

Ao arrastralo aparece un asistente para crealo. Primeiramente hai que seleccionar un informe existente  **Select an existing report** -> **Select a report file...** e seleccionar o informe **subinformeCliente** creado en apartados anteriores:



Select a subreport

Subreport selection mode

- ☒ Workspace resource (an element inside the workspace)
- ☐ Absolute Path in the filesystem (use only for quick testing, never use in real reports)
- ☐ URL (a remote URL referring to a subreport, will be the expression value)
- ☐ Select a resource from JasperReports Server
- ☐ Custom expression (enter an expression for the subreport using the expression editor)
- ☐ No subreport (no subreport reference will be set)

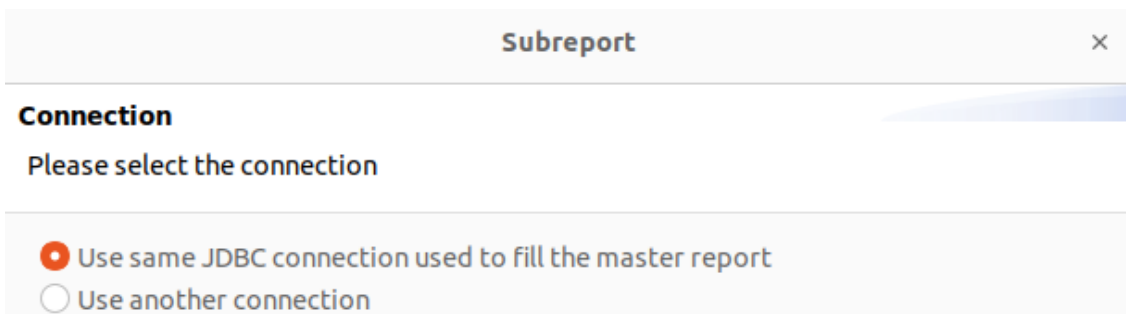
Options

Select a resource from the workspace

subinformeCliente.jrxml Browse ...

Cancel OK

Pulsar Next e seleccionar a mesma conexión que o informe principal:



Subreport

Connection

Please select the connection

- ☒ Use same JDBC connection used to fill the master report
- ☐ Use another connection

A continuación hai que establecer a comunicación entre o informe e o subinforme. É dicir, hai que pasar o parámetro **idCliente** ao subinforme e este parámetro debe ter o valor do campo "IDCLIENTE" da base de datos:

Name	Expression
idCliente	\$F{IDCLIENTE}

Pode modificarse o aspecto do informe principal para que se pareza ao da seguinte imaxe:

Con isto xa se pode facer unha previsualización do informe probando a mostrar o pedido 10248.

Se todo vai ben deben mostrarse os produtos pedidos e o nome do cliente.

Número de factura: 10248

Matti Karttunen

Helsinki

Descrición	Cantidade	Precio por unidade
Queso Cabrales	12	21
Singaporean Hokkien	10	14
Mozzarella di Giovanni	5	35

Cando se fai a previsualización dun informe con un subinforme incrustado realízanse dúas consultas á base de datos. Se o informe incrustado se inclúe na banda **Detail**, farase unha nova consulta á base de datos por cada fila a mostrar. Moito coidado ao facer este tipo de informes porque pode ser moi lento. Polo tanto, como norma xeral, non se deben utilizar subinformes na banda de detalle, salvo contadas excepcións.

Exercicio:

1. Fai un informe e subinforme similar ao anterior onde se mostren os datos dun pedido. Como subinforme deben mostrarse os datos do empregado que atendeu o pedido: nome e apelidos.

Gráficos

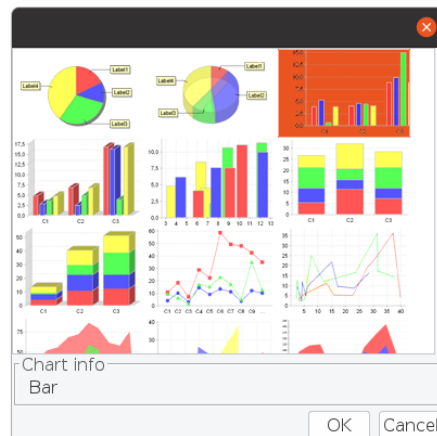
Neste apartado vaise partir do informe elaborado en exercicios anteriores que mostraba o nome e cidade dos clientes agrupados por país.

Nome de contacto	Cidade
"Clientes de " + \$F{COUNTRY}	
\$F{CONTACTNAME}	\$F{CITY}
"Número de clientes: " + \$V{CUSTOMERID_1}	

Imaxinar agora que se queren mostrar os datos en forma de gráfico. É dicir, mostrar o número de clientes por país en forma de gráfico.

Para elaborar un gráfico hai que arrastra o elemento **Chart** á banda **Summary**. Engádese a esta banda porque se trata dun gráfico que non se vai repetir ao longo do informe.

A continuación hai que escoller o tipo de gráfico. Neste caso usarase un gráfico de barras:



A continuación hai que definir as series a representar no gráfico. Un diagrama de barras pode representar varias series, pero neste exemplo só se vai representar unha, polo que a expresión da mesma pode ser unha cadea estática como por exemplo **“Clientes por país”**. No caso de que este texto houboese que collelo da base de datos, especificaríase o campo de onde hai que collelo.

O diagrama tamén necesitará as categorías, que serán mostradas no eixe X e os valores de ditas categorías, que se mostran no eixe Y. A categoría a mostrar será o país, que está no campo **country** da base de datos e o valor da expresión está na variable **\$V{CUSTOMERID1}** que se usou no pé de páxina do grupo.

Chart Wizard

Chart Data Configuration
Configure how data are used by your chart

Series: "Clientes por país" ... Hyperlink

Value: ...

Label: ...

Category: ...

Dataset: ... Return Values Parameters Parameters Map

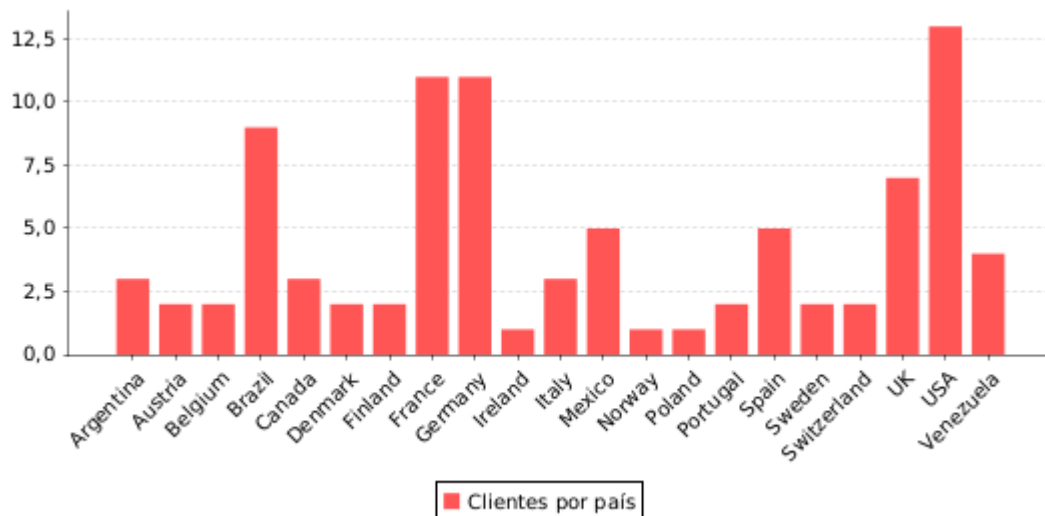
Increment on: ...

Reset on: ...

Unha vez seleccionadas as categorías a mostrar na gráfica e os seus valores, finalizar o asistente pulsando o botón **Finish**.

Por último, falta facer máis grande a banda **Summary** para facerlle espacio ao gráfico. Unha vez feito isto, pode previsualizarse o informe e ver o gráfico na última folia.

Investigar as diferentes propiedades do gráfico e facer probas cambiando os seus valores.



Exercicio:

1. Inclúe un gráfico no informe dos subministradores de tal forma que se mostre cantos subministradores hai por país.

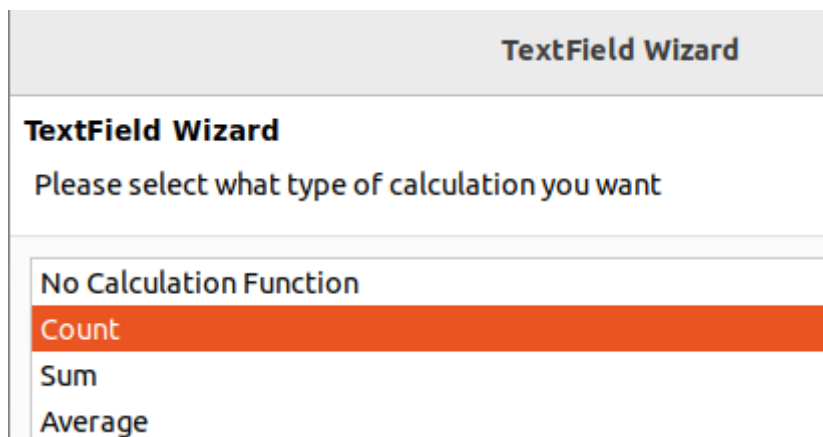
Variables

Ata agora as únicas variables que se viron foron as que creaba automaticamente Jaspersoft Studio ao arrastrar un campo a unha das bandas.

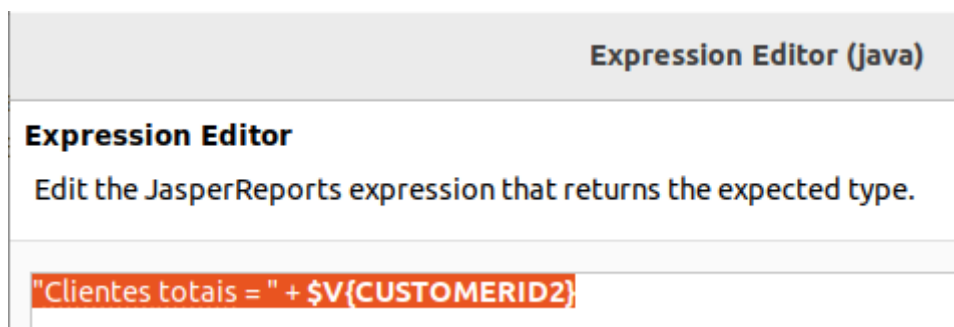
Para explicar como funcionan as variables vaise coller o informe onde están os clientes agrupados por cidade. Informe parecido ao da seguinte imaxe:

Column Header	
Nome	Cidade
"Cidade: " + \${CITY}	
grupoCidade Group Header 1	
\${CONTACTNAME}	\${CITY}
Detail 1	
"Número de clientes: " + \${CUSTOMERID_1}	
grupoCidade Group Footer 1	
Column Footer	

Moitas veces, cando se arrastra un campo a unha banda do informe, créase automaticamente unha variable. Por exemplo, se se quixera incluír na banda **Summary** o número de clientes totais, bastaría con arrastrar o campo **CUSTOMERID** á banda Summary e indicar que o valor que se quere mostrar é o resultado da función de agregación **Count**.



Para mellorar a presentación dos datos pode editarse a expresión do campo inserido no informe e engadir un texto:



Pode facerse a previsualización do informe e comprobar que o número de clientes da base de datos é 91.

A operación de agregar un campo a unha banda, como o caso anterior, crea automaticamente unha variable no informe. Agora mesmo, no informe de clientes agrupados por cidade hai dúas variables:

- CUSTOMERID_1, que mostra o número de clientes dunha cidade,
- CUSTOMERID_2, que mostra o número total de clientes.

Se se accede ás propiedades destas variables, pode observarse o seguinte:

Properties x Problems

fx Variable: CUSTOMERID1 Search Property

Object Advanced

Name: CUSTOMERID1

Value Class Name: java.lang.Integer

Description:

Calculation: Count

Expression: \$F{CUSTOMERID}

Initial Value Expression:

Increment type: None

Incrementer Factory Class Name:

Reset type: [Group] grupoCidade

Properties x Problems

fx Variable: CUSTOMERID2 Search Property

Object Advanced

Name: CUSTOMERID2

Value Class Name: java.lang.Integer

Description:

Calculation: Count

Expression: \$F{CUSTOMERID}

Initial Value Expression:

Increment type: None

Incrementer Factory Class Name:

Reset type: Report

Observar que a primeira variable (CUSTOMERID1) se resetea por cada grupo, mentres que CUSTOMERID2 resetéase ao final do informe.

Para realizar estes cálculos básicos, o asistente funciona perfectamente. Sen embargo, se o que se quere realizar é un cálculo algo máis complexo, haberá que facelo a man. Por

exemplo, imaxinar que se quere calcular o número medio de clientes por cidade. Habería que sumar o total de clientes de cada cidade e dividilo polo número de cidades.

Para facer isto é necesario crear unha variable manualmente. Para facelo hai que pulsar co botón dereito do rato sobre **variables** -> **Create Variable**. A continuación vaise cambiar o nome desta variable para poñerlle un nome máis significativo, como por exemplo **mediaCidades** e modificar as súas propiedades cos seguintes valores:

- **Name:** mediaCidades
- **Variable Class:** java.lang.Float
- **Calculation:** Average (porque se vai calcular unha media).
- **Increment type:** [Group] grupoCidade

Fai referencia a cando se quere que se teña en conta o dato para facer a media. Como se quere que se teña en conta o valor ao final do grupo, hai que indicar o grupo ao que fai referencia, neste caso **grupoCidade**.

- **Reset type:** Report. Esta propiedade indica cada canto se resetea esta variable. A variable mediaCidades vaise resetear ao final do informe.

Recordar que a variable que contaba o número de clientes por grupo se reseteaba por grupo. É dicir, con cada novo grupo debe volver a empezar en 0.

Sen embargo, a variable mediaCidades non é necesario reseteala.

- **Expression:** \$V{CUSTOMERID_1}

É o valor sobre o que se vai facer a media. A variable \$V{CUSTOMERID_1} fai referencia ao número de clientes por cidade.

The screenshot shows the 'Properties' window for a variable named 'mediaCidades'. The window has tabs for 'Object' and 'Advanced', with 'Advanced' selected. The 'Variable: mediaCidades' is shown at the top. Below it, the 'Object' tab is active, displaying various properties:

- Name:** mediaCidades
- Value Class Name:** java.lang.Float
- Description:** (empty field)
- Calculation:** Average
- Expression:** \$V{CUSTOMERID1}
- Initial Value Expression:** (empty field)
- Increment type:** [Group] grupoCidade
- Incrementer Factory Class Name:** (empty field)
- Reset type:** Report

Tan só queda arrastrar a variable **mediaCidades** á banda **Summary** e facer a previsualización do informe para comprobar que todo funciona:

```
Cientes totais = 91
Media clientes por cidade = 1.3188406
```

NOTA: A orde de creación/declaración de variables inflúe no resultado dos informes. Por exemplo, na imaxe anterior, `$V{CUSTOMERID_1}` debe estar declarada antes da variable `mediaCidades`.

Exercicios:

1. Engade ao informe dos subministradores información de cantos subministradores hai na base de datos e o número medio de subministradores por país.
2. Crea un informe onde se mostren os produtos dun pedido pasado como parámetro. Debe mostrarse o nome de produto, a cantidade pedida e o prezo por unidade.

Engade a este informe unha variable que informe do prezo total por produto e o prezo total do pedido.

Referencias

Para a elaboración deste material utilizáronse, entre outros, os recursos que se enumeran a continuación:

- https://ioc.xtec.cat/materials/FP/Materials/2252_DAM/DAM_2252_M07/web/html/index.html
- <https://github.com/clebermatheus/NorthwindDerby>
- [Jasper Report](#)
- [Tutorial de Jaspersoft Studio](#)