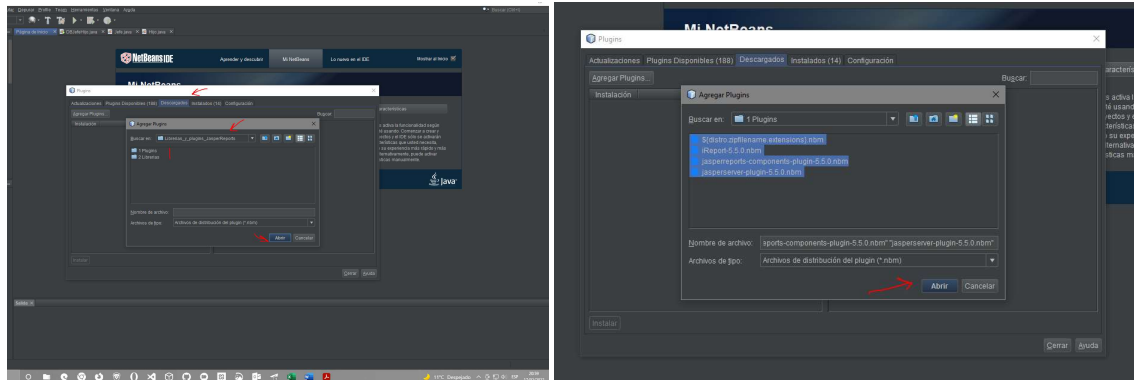


Para desarrollar las actividades de esta unidad, es necesario que tengas instalado MySQL e importada la base de datos perteneciente a la compañía **SumiMetalicos S.A** que se ha utilizado para explicar los ejemplos de esta unidad.

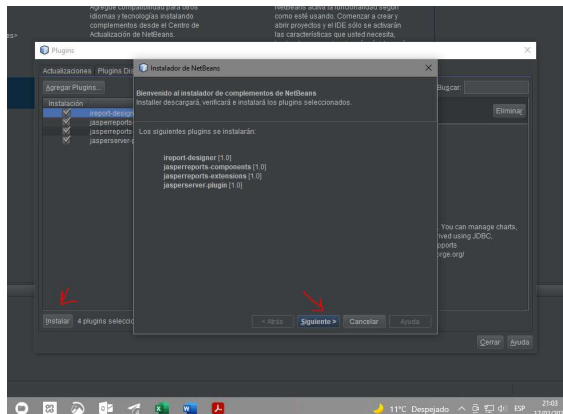
También será necesario que tengas instalado NetBeans junto con los plugins y librerías de JasperReport como se te indica en el contenido de la unidad.

Siguiendo las pautas marcadas desde el temario, y tras haber descargado la carpeta de plugins y librerías, procedo a instalar las mismas abriendo Netbeans y accediendo al menú **Herramientas/plugins** pestaña **Descargados/Agregar plugins**, donde buscamos las carpetas descargadas para su instalación



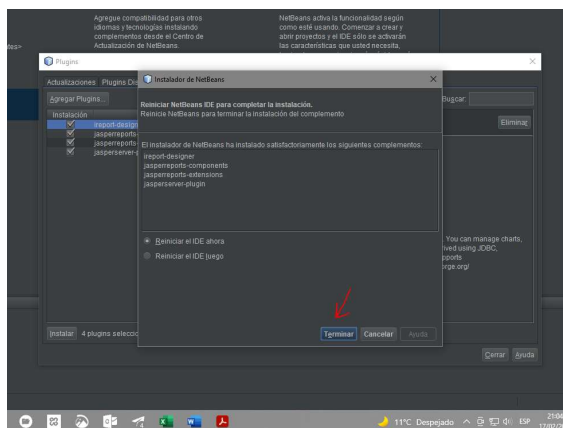
Selección de archivos.

Seguimos el proceso de instalación que nos va mostrando.



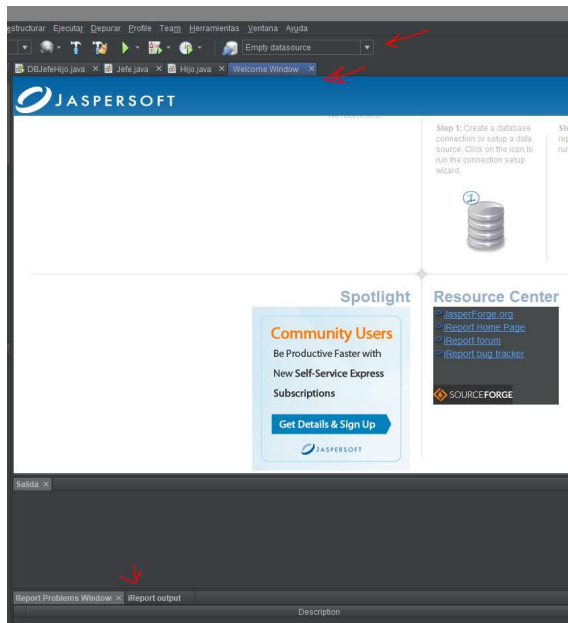
Instalación de plugin.

Nos pedirá reiniciar para completar la instalación.



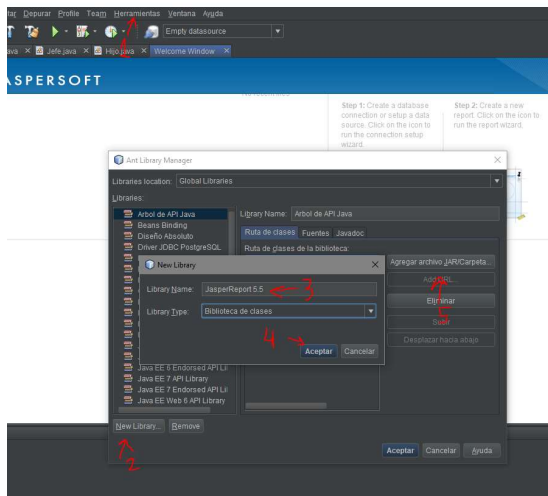
Podemos reiniciar en ese momento o más tarde.

Tras reiniciar, ya vemos disponible JasperReport para NetBeans.

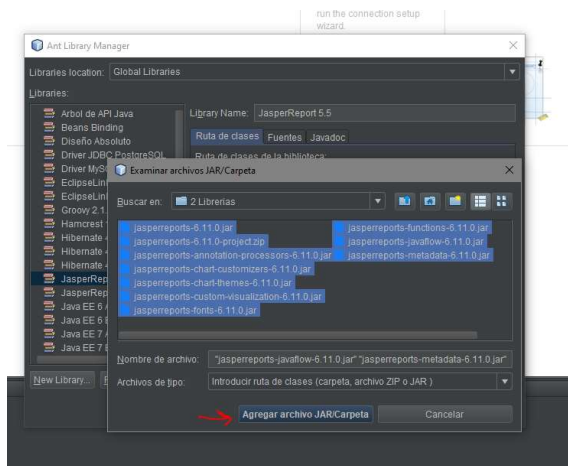


Ventana de bienvenida JasperSoft.

Ahora, les toca el turno a las librerías (podemos ponerle el nombre que queramos), con un proceso similar al anterior.

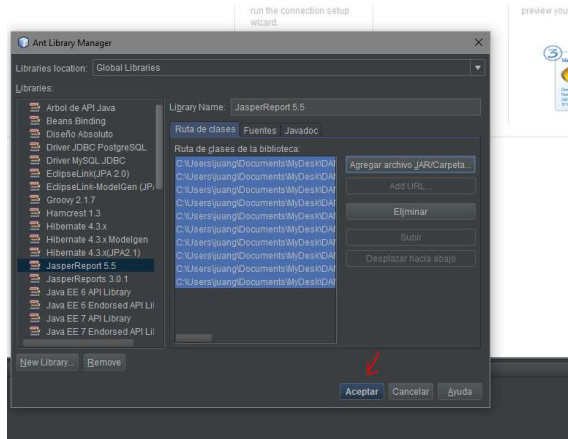


Pasos instalación de librerías.



Detalle proceso de instalación de .jar.

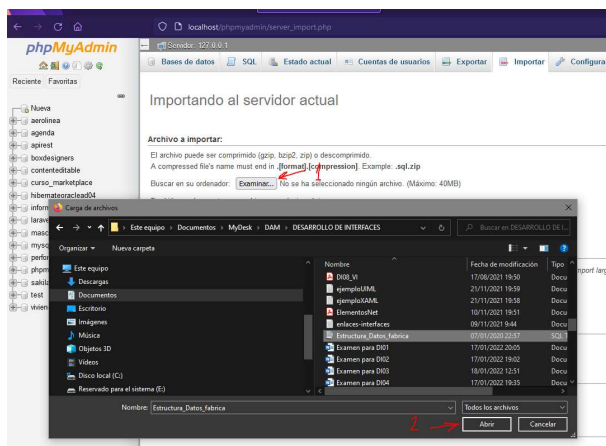
Tras seleccionar y agregar los **.jar** necesarios, sólo queda aceptar la instalación para tener todo preparado para empezar a generar informes con **JasperReport**.



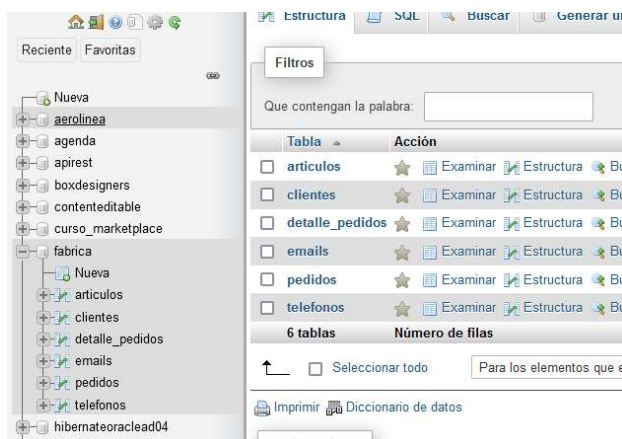
Comprobar archivos y aceptar instalación.

Ya tenemos disponible el plugin y las librerías necesarias para trabajar con **JasperReport**.

Voy a importar la base de datos descargada del temario a través de **phpMyAdmin** y en mi caso, le pongo el mismo nombre que en la plataforma: **fabrica**.



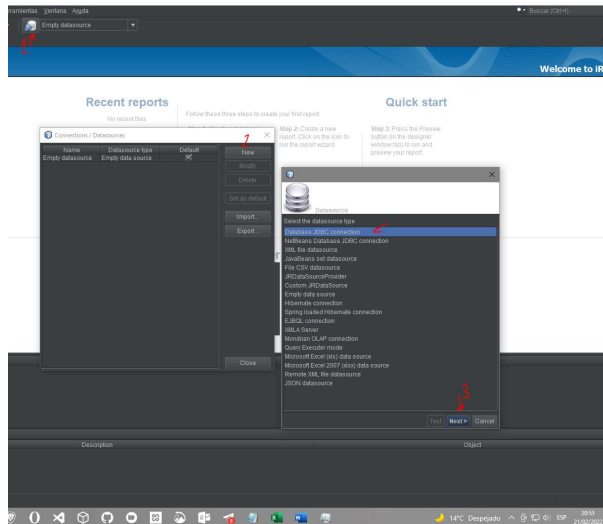
Importación de archivo sql con base de datos.



Detalle de BD y tablas importadas.

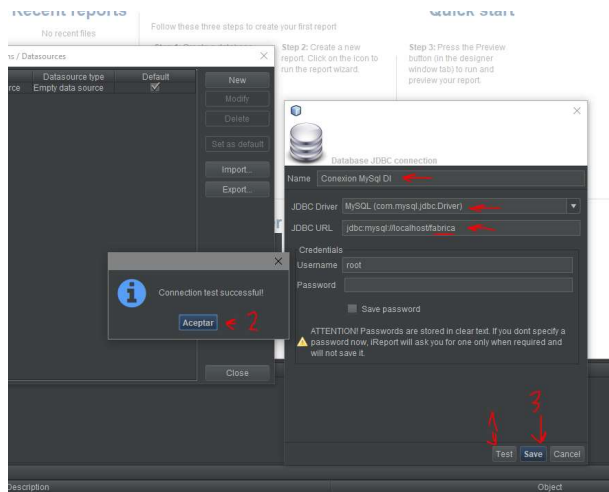
Teniendo **JasperReport** y la base de datos **Fabrica** en **MySQL**, el siguiente paso es crear la conexión en **NetBeans**.

Para ello, hacemos click sobre el icono de conexión, seleccionamos **new/Database JDBC connection**.



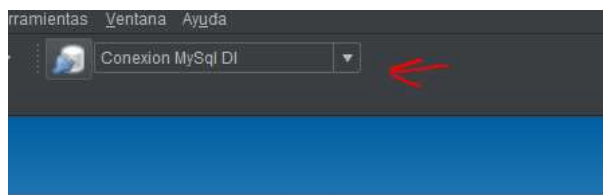
Selección de conexión a BD.

Completamos los datos de la conexión, hacemos un test para probarla y guardamos los cambios.



Ingresa datos y probar la conexión.

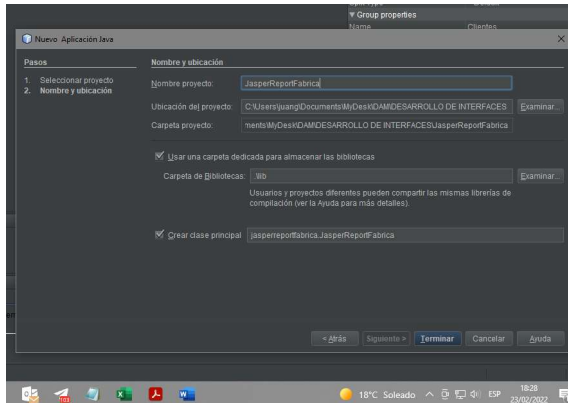
Con ello, ya está disponible la conexión para nuestra base de datos.



Conexión configurada.

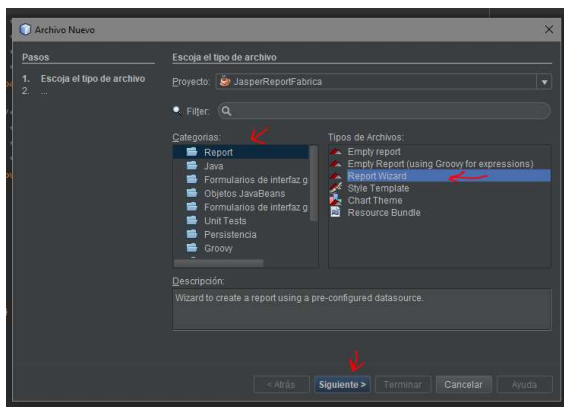
1. **Genera un informe llamado "facturas"**, en el que aparezca los datos de las empresas clientes (nombre, dirección y Ciudad) (tabla **Clientes**) y un listado con los pedidos que han realizado. Por cada pedido se mostrará el **ID_pedido**, fecha del pedido e importe a pagar. El importe a pagar deberá de calcularse teniendo en cuenta los artículos que se vendieron en cada pedido junto con el precio al cual fue vendido. Esta información la debes de recuperar de la tabla **detalle_pedidos**. También es necesario añadir el importe total para cada cliente. Para resolver este ejercicio **no se puede emplear subinformes**.

Con la conexión configurada, lo siguiente es crear un nuevo proyecto de tipo **Java application** que voy a llamar **JasperReportFabrica**.



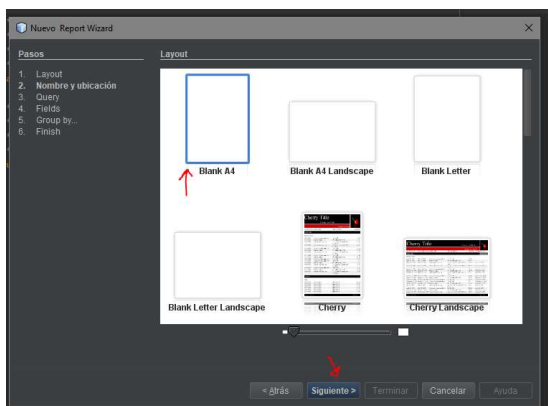
Creación de aplicación java para nuestra tarea.

Dentro del proyecto, hay que crear un nuevo archivo. Ya nos muestra la categoría **Report/Report Wizard**.

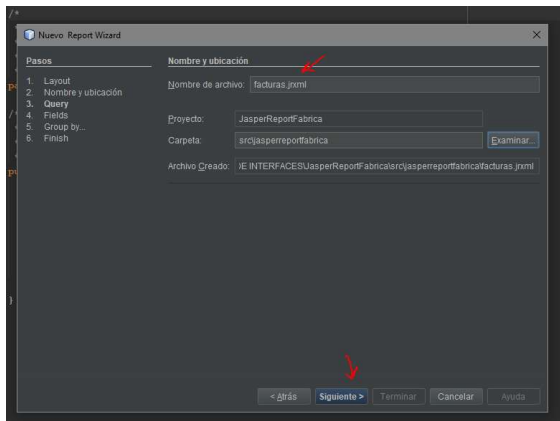


Creación de archivo con la plantilla para nuestro informe.

Seleccionamos **Blank A4** como **Layout**.

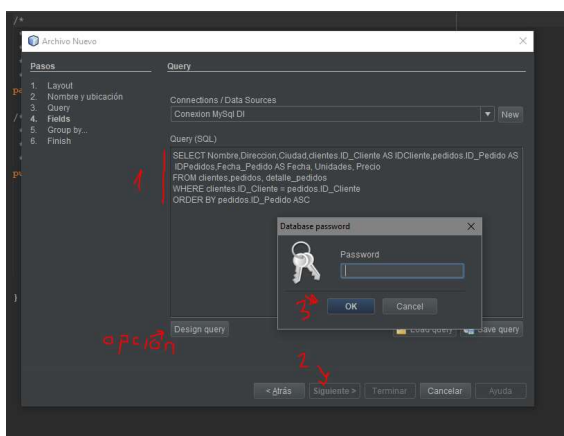


Selección de Layout.



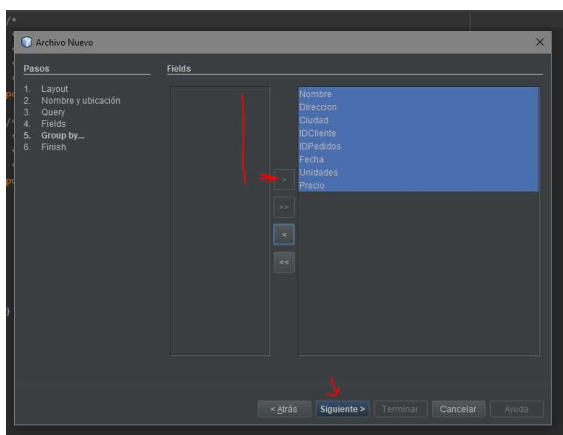
Detalle de creación de archivo para informe.

Hay que añadir la **query** de consulta de datos. Si se quiere, hay posibilidad de usar el diseñador que ofrece. Tras aceptar, nos pide la contraseña.



Añadimos conexión y query.

Al acceder, recoge los campos y seleccionamos los que necesitamos. En mi caso, todos.



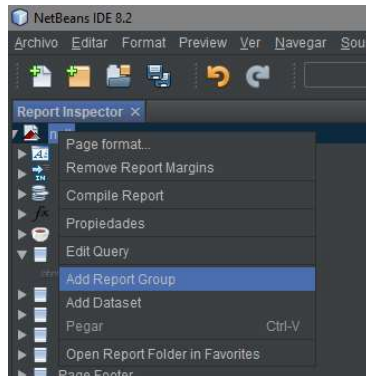
Selección de campos.

Tras ello, nos da la opción de agruparlos si es necesario (no en este caso) y finaliza la creación de la nueva plantilla de informe.

Le añado un **Static Text** en la banda **Title**, al que personalizo fuente y le añado un color de fondo, mediante las propiedades **Backcolor** y **Opaque**.

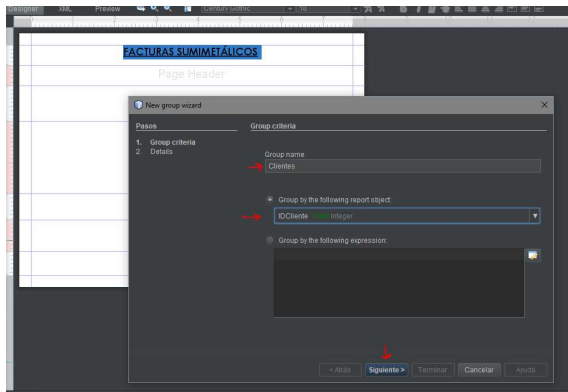
El siguiente paso es crear dos grupos para manejar los datos de clientes y pedidos.

Desde la raíz del **Report Inspector**, creamos un nuevo grupo mediante menú contextual/Add Report Group.



Menú añadir grupo.

Le pongo de nombre **Clientes** y selecciono el campo por el que va a recoger los datos.



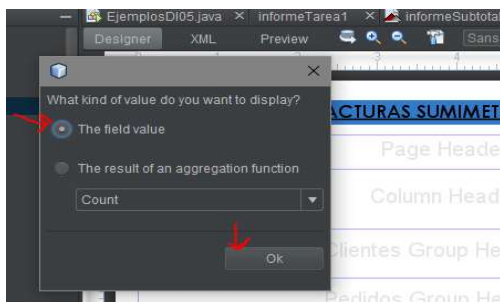
Poner campo con el que vamos a agrupar las peticiones.

Nos pregunta a continuación si queremos que genere un **header** y un **footer** para el grupo, a lo que decimos que sí, y terminamos la creación.

Seguiremos los mismos pasos, pero para crear el grupo **Pedidos**, que usará el campo **IDPedidos**.

Tras ello, nos habrá generado nuevas bandas con esos nombres y que se usarán para mostrar los campos solicitados.

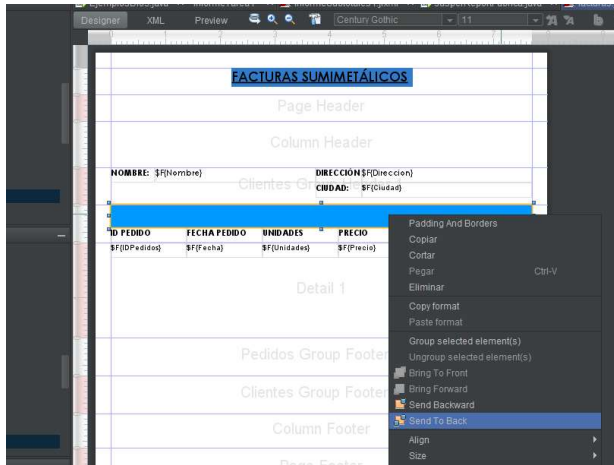
En la banda **Clientes Group Header 1**, se añaden, arrastrando desde el menú del **Report Inspector/Fields**, los campos **Nombre**, **Dirección** y **Ciudad**. Para este caso, solo queremos el valor, por lo que debe seleccionarse. Agregaremos al lado de los campos un **Static Text** con el nombre de cada uno, para una mejor comprensión. Además, desde las propiedades del grupo, selecciono la opción **Start on a new page**, para que cada cliente se muestre luego en una página.



Seleccionar opción para mostrar valor.

Con el mismo método, en la banda **Detail**, añadimos los campos **IDPedidos**, **Fecha**, **Unidades** y **Precio** (estos dos por una cuestión de mejor comprensión del documento. Se “arrastran “ generando un **Static Text** con el nombre del campo y su par valor. Los nombres de los campos, los alojamos en la banda **Pedidos Group Header 1**, y los personalizamos desde sus propiedades, igual que el resto.

Personalizamos esta banda mediante un **frame** que arrastramos desde la paleta, al que damos color y pasamos hacia atrás (**Send to Back** desde el menú contextual del **frame**).



Menú contextual del frame.

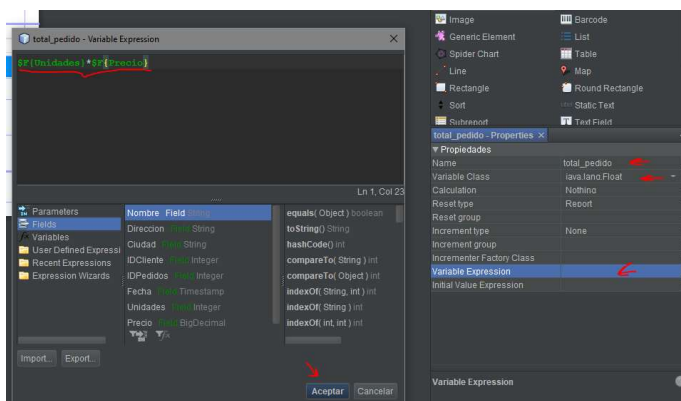
Ajustamos alturas de la bandas y probamos si va bien orientada antes de proseguir, mediante la pestaña **Preview**.



Vista preliminar.

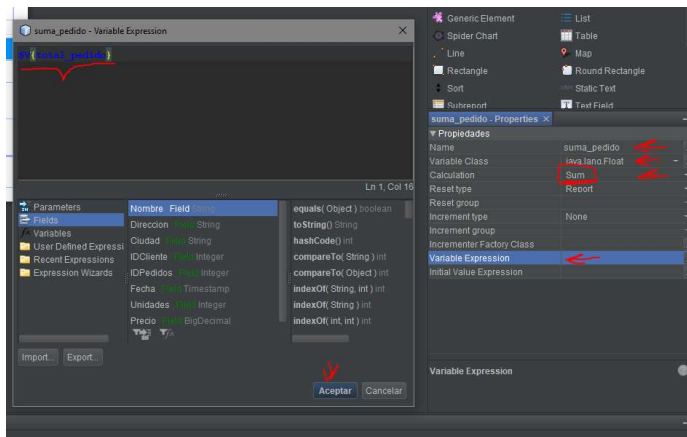
Tras comprobarlo, podemos ir a crear dos variables: **total_pedido** que recoge la multiplicación de los campos **Unidades** por **precio** y **suma_pedido**, que recoge el **importe total a pagar**.

Se generan desde **Report Inspector/variables** menú contextual/**Agregar Variable**.



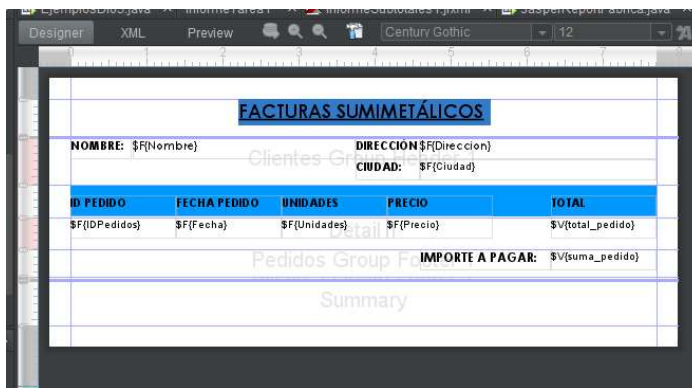
Creación de variable y su expresión.

Es importante en esta segunda variable, añadir el operador **SUM** en la propiedad **Calculation**.



Creación de variable y su expresión, añadiendo fórmula.

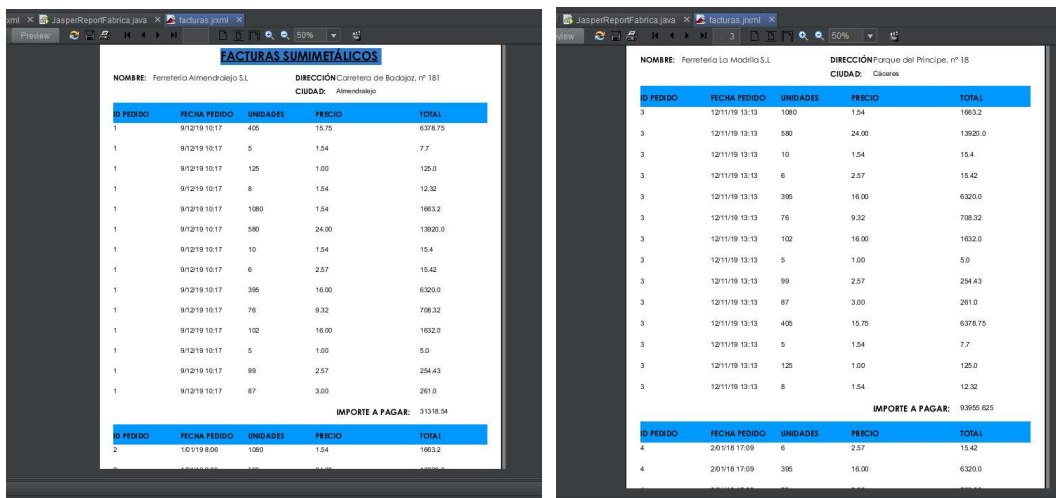
Con ellas creadas, las añadimos al informe y ajustamos alturas de bandas (también las podemos eliminar), quedando así finalmente.



Detalle diseñador con los campos definidos.

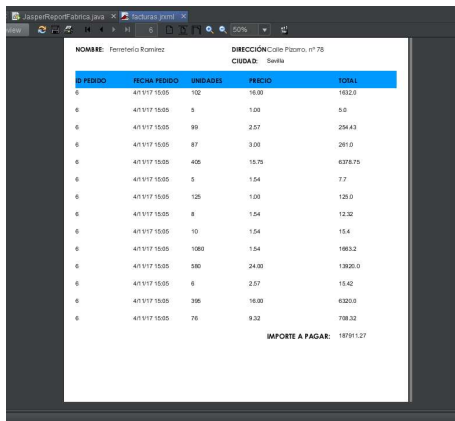
Comprobamos la ejecución. Vemos los pedidos de cada cliente de forma detallada y ordenada.

Si avanzamos, el siguiente solo se mostrará en una nueva página.



Vista preliminar.

Y de forma sucesiva, cliente por cliente.



Reporte for 'Ferretería Ramirez' showing a table of items and their prices.

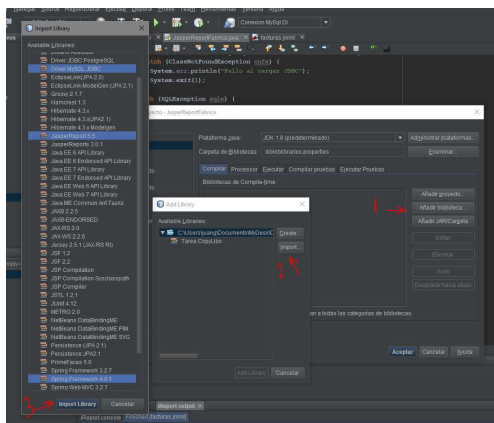
ID PEDIDO	FECHA PEDIDO	UNIDADES	PRECIO	TOTAL
6	4/1/17 15:00	100	16.00	1602.0
6	4/1/17 15:00	5	1.00	5.0
6	4/1/17 15:00	99	2.57	254.43
6	4/1/17 15:00	87	3.00	261.0
6	4/1/17 15:00	400	15.70	6278.70
6	4/1/17 15:00	5	1.54	7.7
6	4/1/17 15:00	125	1.00	125.0
6	4/1/17 15:00	8	1.54	12.32
6	4/1/17 15:00	10	1.54	15.4
6	4/1/17 15:00	1000	1.54	1540.0
6	4/1/17 15:00	500	24.00	12000.0
6	4/1/17 15:00	6	2.57	15.42
6	4/1/17 15:00	390	16.00	6240.0
6	4/1/17 15:00	76	9.30	706.80

IMPORTE A PAGAR: 18791.27

Hoja final.

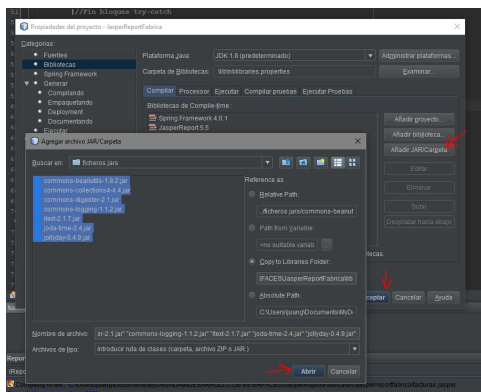
- Incluir el informe para generar facturas en una aplicación java que lo muestre en formato PDF, teniendo en cuenta que el código del cliente (**ID_Cliente**) se pasa al informe como parámetro. Por lo tanto, al ejecutarse la aplicación se generará el informe creado en el apartado anterior pero mostrando solo los datos para el Identificador de cliente (**Id_cliente**) pasado por parámetro.

Para incluir un informe en una aplicación Java, necesitaremos añadir en el proyecto las librerías **Driver MySql JDBC**, **Spring Framework 4.0.1** y **JasperReport 6.11**.



Pasos para añadir las librerías al proyecto.

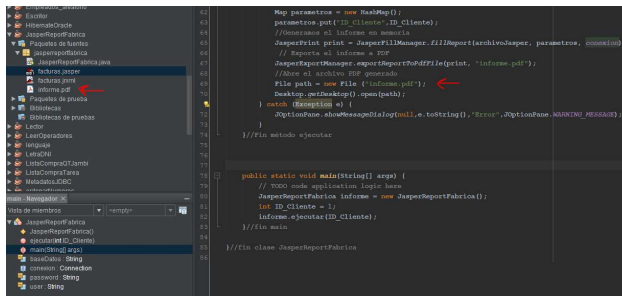
Y los archivos **jar** necesarios según el temario y que hemos podido descargar desde la plataforma, **commons-beanutils-1.8.2.jar**, **commons-collections4-4.4.jar**, **commons-digester-2.1.jar**, **commons-logging-1.1.2.jar**, **itext-2.1.7.jar**, **joda-time-2.4.jar**, **jollyday-0.4.9.jar**.



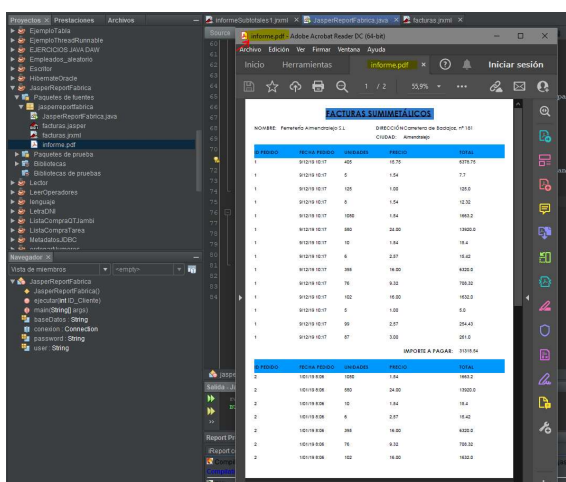
Añadiendo los **jar** necesarios.

Adaptamos el código ofrecido en el temario para crear la aplicación Java, y que así apunte a nuestro informe y conexión configurados (puede verse el código completo y documentado, en los archivos de la tarea).

Tras ejecutarlo, podemos ver que nos añade el nuevo archivo al directorio.



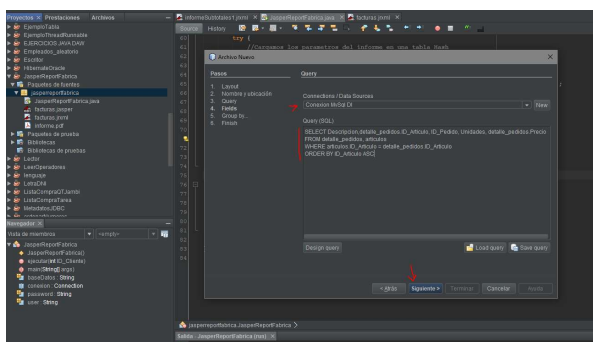
Ejecutando aplicación java para crear un informe.pdf.



Comprobamos la creación correcta del archivo generado.

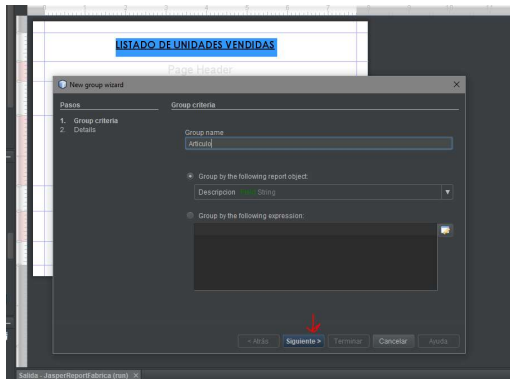
3. Crear un informe en el que aparezca un listado con las ventas totales (cantidad total) de cada artículo (se visualiza su nombre). Por cada artículo vendido, debe de mostrarse la descripción del mismo junto con el número de unidades totales vendidas. Añadir un gráfico al informe que muestre por cada artículo vendido, el número total de unidades vendidas. Esta información la puedes obtener de la tabla **detalle_pedidos**.

Este punto pide crear un nuevo informe, por lo que lo generamos con el mismo procedimiento detallado anteriormente, pero adaptando la **query** a nuestras necesidades (recordemos que siempre podemos editarla luego si necesitamos ajustar más la consulta).



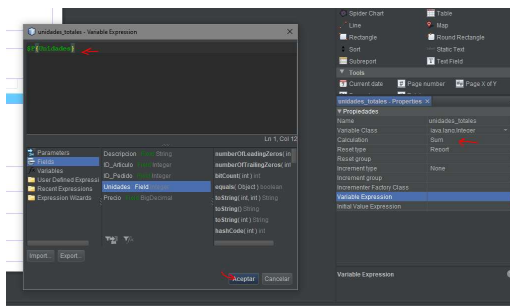
Creación de informe, conexión y query.

Creamos un **nuevo grupo** **Articulo** donde señalamos el campo descripción para el agrupamiento.



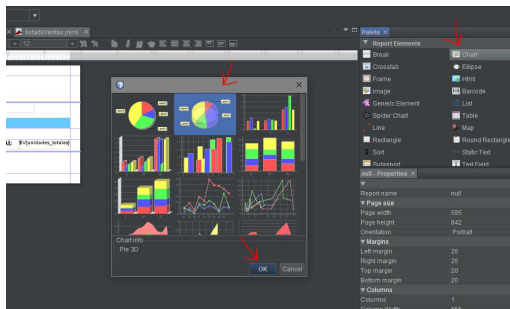
Creación de grupo Articulo.

Creamos una variable para sumar las unidades.



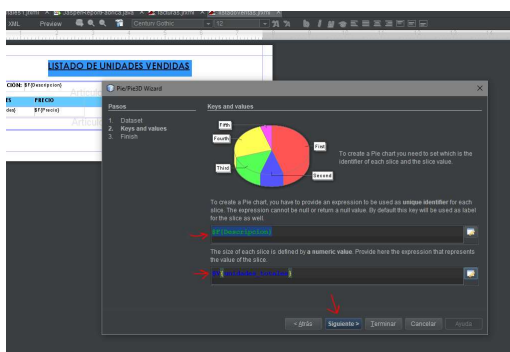
Creación de variable con suma.

Para el gráfico elegimos desde la paleta un **Chart** de tipo **Pie 3D**. Al ser pocos elementos, pueden distinguirse de forma gráficamente correcta aunque algún texto se pueda sobreponer a otro.



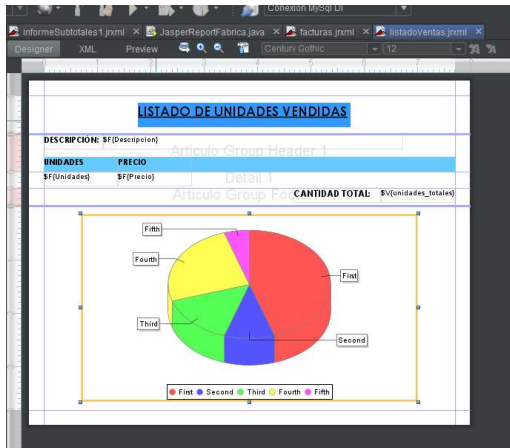
Selección de modelo de gráfico.

Indicamos en el siguiente paso los campos y valores a mostrar.



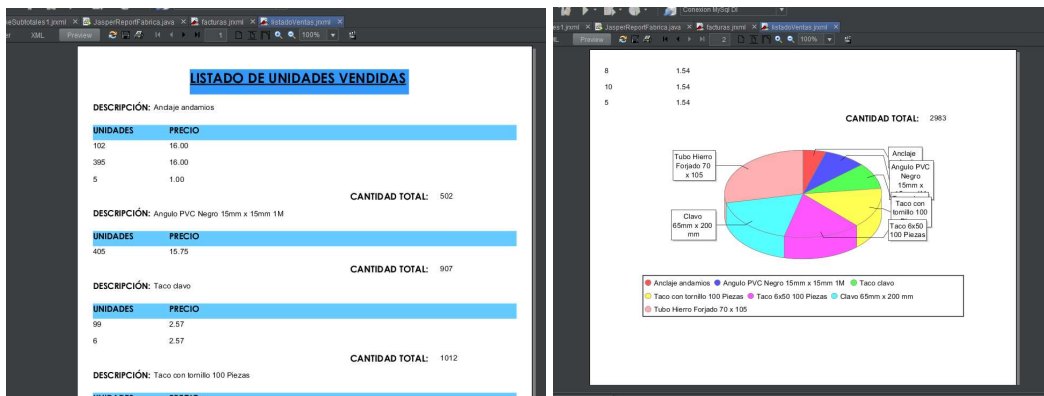
Selección campo y valor.

Puede ubicarse en la banda de título, o como en este caso, al final del listado, en la banda **summary**.



Detalle diseñador.

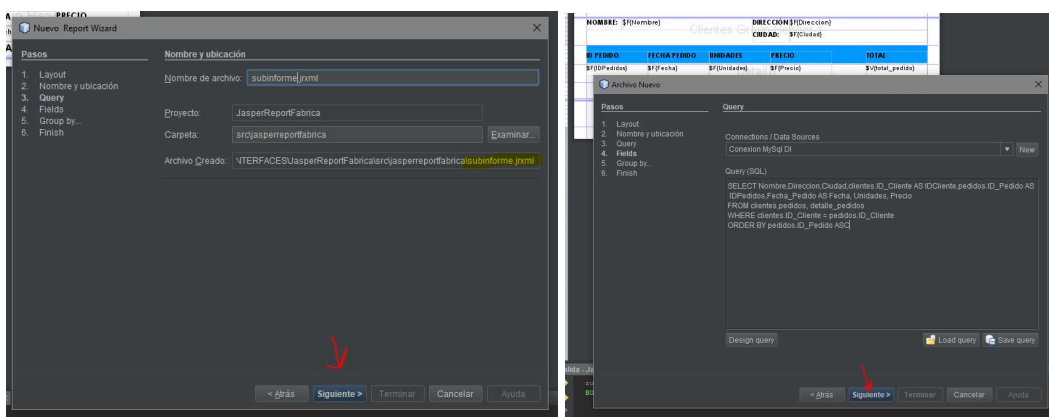
Comprobamos la ejecución desde la pestaña de **preview** y el gráfico al final.



Vista previa con gráfico.

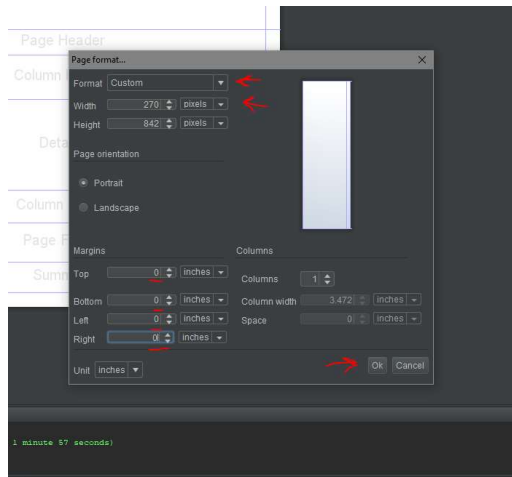
4. Repite el ejercicio 1 (sin subtotales) utilizando subinformes.

Para nuestro subinforme creamos un nuevo informe llamado **subinforme.jrxml**, al que añadimos la **query** necesaria para la ejecución.



Creación de **subinforme.jrxml** y configuración.

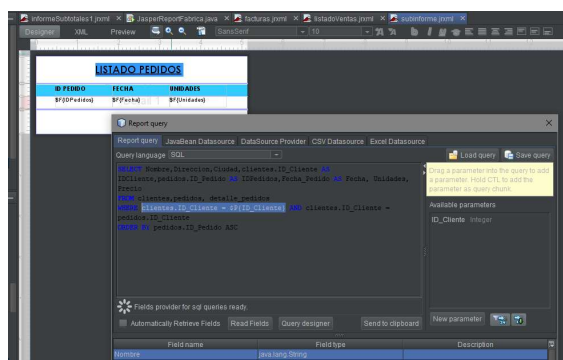
En los subinformes, debemos quitar márgenes para reducir su formato (van integrados en el informe principal y debemos evitar espacios innecesarios). Por ello, desde el menú secundario, accedemos a **Format/Page format** y reducimos su anchura en lo posible (la altura la gestiona el informe principal) y dejamos el resto a cero.



Formato subinformes sin márgenes.

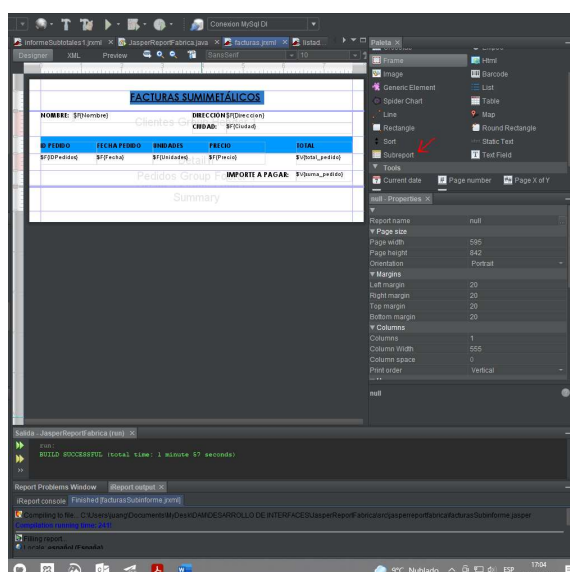
Desde **Report Inspector/Parameters**, creo un nuevo parámetro, llamado **ID_Cliente** de tipo **java.lang.integer** para poder gestionar el listado.

Creamos nuestro informe en el diseñador (ya no nos pedía subtotales a diferencia del principal), y modificamos la consulta antes de probarlo, para añadir el parámetro creado.



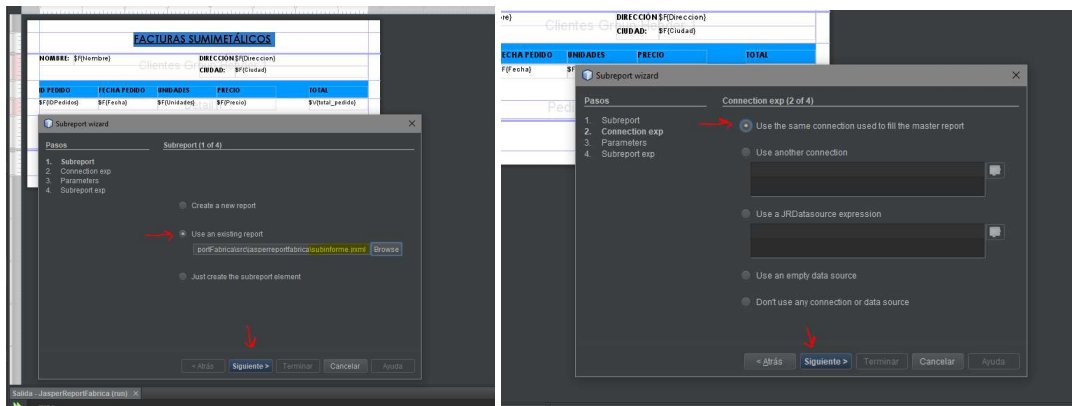
Incluimos el parámetro creado en la query.

Queda integrarlo en el informe principal. Para ello, vamos a la paleta de nuestro proyecto y arrastramos hasta nuestra plantilla un elemento **subreport**.



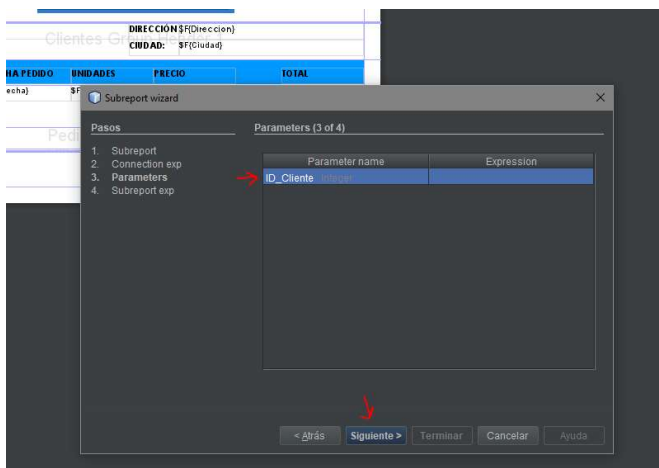
Integrando subinforme en informe principal.

Localizamos el subinforme creado a través de **Browse** en la opción de **Use an existing report** (usar un report existente). En la siguiente ventana, optaremos por usar la misma conexión que el informe principal.



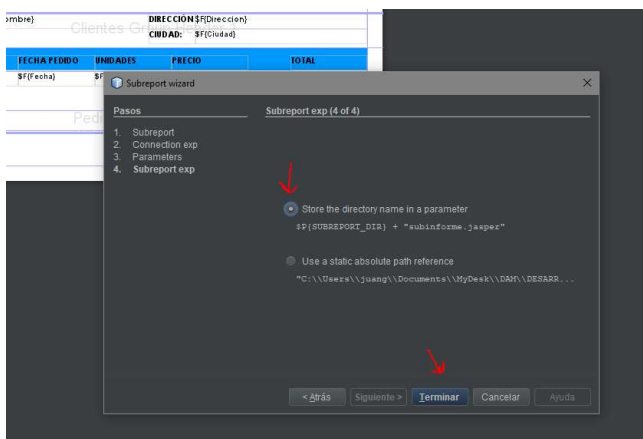
Indicamos archivo a integrar y conexión.

Nos mostrará los parámetros que podamos tener definidos. Seleccionamos el **ID**.



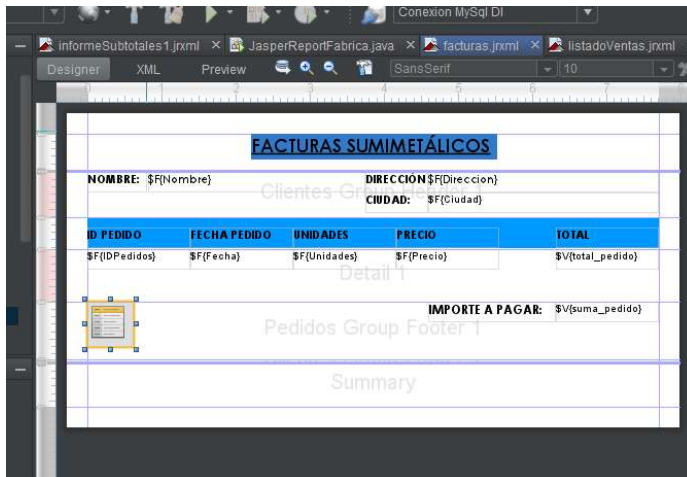
Selección de parámetros.

Finalizamos, seleccionando la opción **Store the directory name in a parameter** para almacenar carpeta y parámetro.



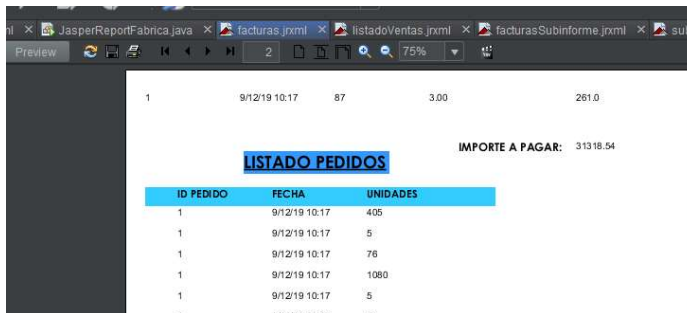
Finalizando definición.

Como último paso, ubicamos el subinforme en la posición donde queremos mostrarlo.



Ubicamos el subinforme en el layout.

Al visualizarlo a través de la pestaña **preview**, podemos ver como queda insertado al final del cliente.



Detalle de vista previa.