

# Desarrollo Avanzado de Aplicaciones I

## Introducción a JDBC 3



i

inicio

d

desarrollo

a

aplicación

t

término



idat

# Inicio

Logro de aprendizaje – Introducción



# Logro de Aprendizaje

“Al finalizar la sesión, el participante podrá implementar reportes usando la Arquitectura de Jasper Reports.

# Introducción



- Revisión Rápida de Temas de Sesión Anterior
- Revisión de Ejercicios de Sesión Anterior
- Inquietudes y/o Preguntas
- Agenda de Sesión
  - JasperReports Library
  - Jaspersoft Studio
  - Ejemplos y Ejercicios



# Desarrollo

Desarrollo del Contenido de la Sesión



- ✓ JasperReports es un motor de informes de código abierto que está completamente escrito en Java y puede usar datos provenientes de cualquier tipo de fuente de datos y producir documentos que se pueden ver, imprimir o exportar en una variedad de formatos de documentos, incluidos HTML, PDF, Excel, OpenOffice y Word.
- ✓ JasperReports funciona a través de la generación de un código XML donde se recogen las particularidades de los informes. Este XML se compila y se ejecuta para obtener los documentos resultantes.
- ✓ JasperReports es una biblioteca de clases java que es el motor central de Jaspersoft Studio y JasperReports Server.



**JasperReports Library**

Open Source Java Reporting Library



- ✓ Jaspersoft Studio es el diseñador de informes para JasperReports y JasperReports Server que permite crear diseños sofisticados de informes conteniendo gráficos, imágenes, subinformes, tabulaciones cruzadas, etc.
- ✓ Cuando se diseña un informe utilizando Jaspersoft Studio, se crea un archivo .JRXML, que es un documento XML que contiene la definición del diseño del informe. El diseño es completamente visual, por lo que se puede ignorar la estructura subyacente del archivo .JRXML.
- ✓ Antes de ejecutar un informe, el archivo .JRXML debe compilarse en un objeto binario llamado archivo .Jasper. Los archivos .Jasper son los que se necesitan en las aplicaciones cliente para ejecutar los informes.



**Jaspersoft Studio**

Open source report designer for JasperReports and JasperReports Server





# Jaspersoft Studio: Interface de Usuario



The screenshot displays the Jaspersoft Studio user interface. At the top is a menu bar (File, Edit, View, Navigate, Project, Window, Help) and a toolbar. Below the menu bar, the 'Repository Explorer' and 'Project Explorer' panels are highlighted with a red box. The 'Repository Explorer' shows a tree structure of data sources and reports. The 'Project Explorer' shows the project files. The main workspace is divided into three tabs: 'Design', 'Source', and 'Preview'. The 'Design' tab is active, showing a report titled 'Listado de Actores' with a table containing columns 'actor\_id', 'first\_name', and 'last\_name'. The 'Source' tab shows the XML code for the report. The 'Preview' tab shows a visual representation of the report. On the right side, there is a 'Palette de Diseño' (Design Palette) with various elements like Text Field, Static Text, Image, Break, Rectangle, Ellipse, and Line. Below the palette, there are 'Properties' and 'Problems' panels. The 'Properties' panel shows the properties for the selected element, 'Sakila Database'. At the bottom, there is a 'Report State' panel showing statistics like Compilation Time, Filling Time, Report Execution Time, Export Time, and Total Pages.

Vista del Explorador del Proyecto y de los Repositorios de Datos

Vista de Diseño del Informe

Código XML del Informe

Previsualización del Informe

Paleta de Diseño

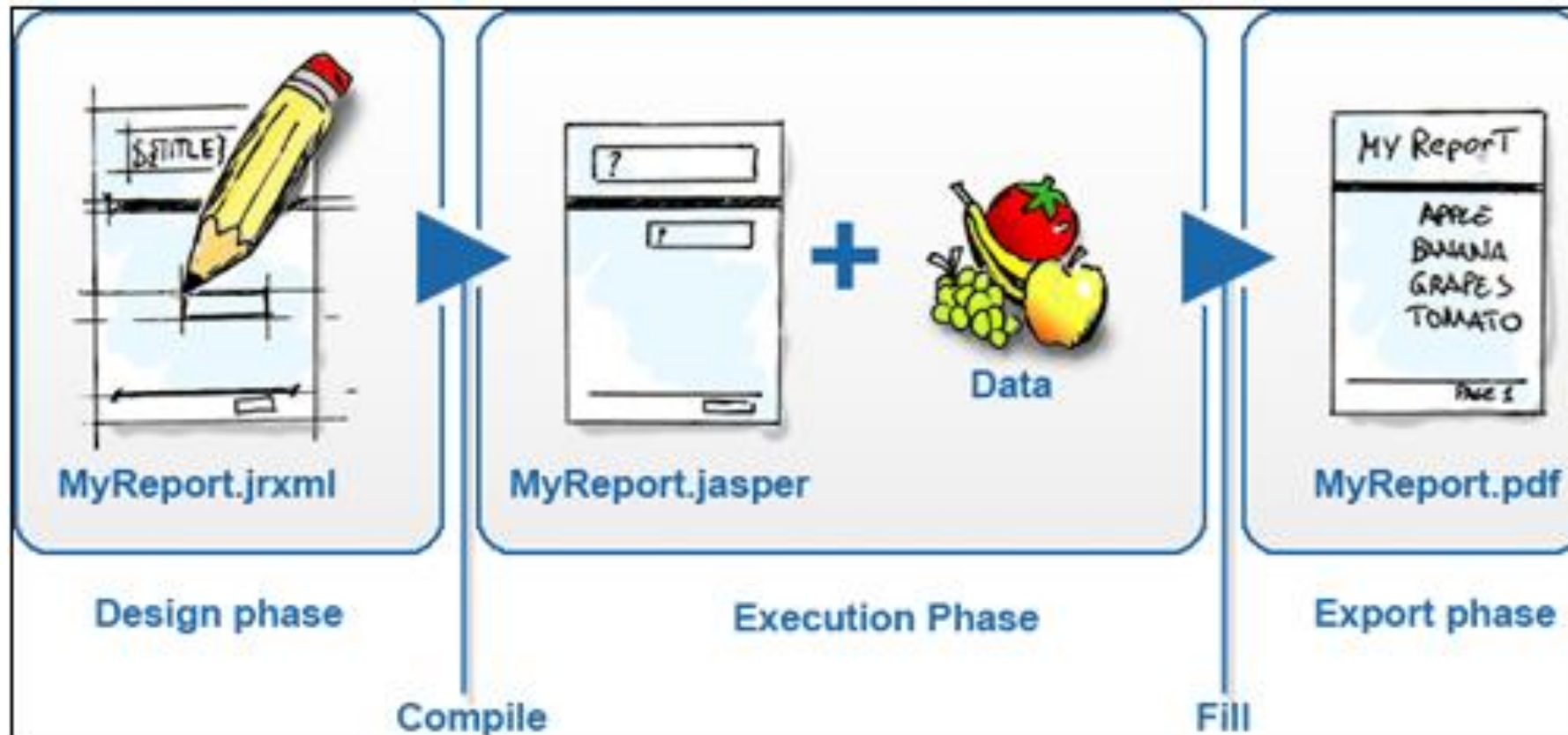
Estructura del Informe

Vista del Estado del Informe

Vista de Propiedades



# Ciclo de Vida del Informe



# Ejemplo de Uso de los Informes en NetBeans



```
import java.sql.DriverManager;
import java.sql.Connection;
import java.sql.SQLException;
import net.sf.jasperreports.engine.JRException;
import net.sf.jasperreports.engine.JasperFillManager;
import net.sf.jasperreports.engine.JasperPrint;
import net.sf.jasperreports.view.JasperViewer;
```

Importar las clases necesarias. Previamente se ha debido de agregar las librerías necesarias.

```
public class JasperReports {
    public static void main(String[] args) {
```

Utilizar una conexión a BD.

```
        Connection conn = null;
```

```
        try {
```

```
            conn = DriverManager.getConnection("jdbc:mysql://localhost/sakila?" +
                "user=root&password=admin");
```

Crear objeto JasperPrint para cargar los datos.

```
            JasperPrint jasperPrint = JasperFillManager.fillReport(
                sourceFileName: "C:\\Users\\Hp\\JaspersoftWorkspace\\MyReports\\ReporteActores.jasper",
                params: null, connection: conn);
```

Usar un objeto JasperViewer para previsualizar el informe.

```
            JasperViewer jasperViewer = new JasperViewer(jasperPrint);
            jasperViewer.setVisible(true);
```

```
        } catch (SQLException ex) {
            System.out.println("SQLException: " + ex.getMessage());
            System.out.println("SQLState: " + ex.getSQLState());
            System.out.println("VendorError: " + ex.getErrorCode());
```

Capturar las excepciones a través de JRException.

```
        } catch (JRException jre) {
            System.out.println("jre.getMessage());
```



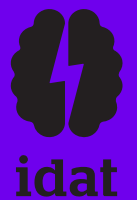
# Aplicación

Revisar ejemplos y realizar ejercicios prácticos



# Término

Indicaciones generales y/o Resumen de Sesión



# Resumen de Sesión



- JasperReports Library
- Jaspersoft Studio
- Ejemplos y Ejercicios



GRACIAS