ACE Inspiration
Professional Development Center

JasperReports
Basic Tutorial

About the Tutorial

JasperReports is an open source java reporting engine. JasperReports is a Java class library, and it is meant for those Java developers who need to add reporting capabilities to their applications. This simple and user-friendly tutorial covers almost all the basic of JasperReports that a beginner should know.

Audience

This tutorial is designed for Software Professionals as well as for all those beginners who would like to learn the concepts of JasperReports.

Prerequisites

Before proceeding with this tutorial, it is expected that you have a basic understanding of Java programming language. A basic understanding of Java and other associated programming will be an additional advantage to understand the topic.

ACE Inspiration
Professional Development Center

Table of Contents

ACE Inspiration
Professional Development Center

1. JasperReports - Getting Started

JasperReports is an open-source reporting tool for Java that is used for reporting of applications.
This reporting tool helps in writing rich content onto the screen, to the printer, or into a variety of files like HTML, PDF, XLS, RTF, CSV, XML, ODT and TXT.
JasperReports can also be used for Java-enabled applications like Java EE or web applications to generate dynamic content.

Features of JasperReports

- Offers flexible report layout.
- Developers can supply data in multiple ways.
- It allows you to accept data from multiple data sources.
- Watermarks can also be applied.
- Multiple data sources can be used to transfer data.
- You can generate subreports
- It can generate subreports.
- It helps you to export reports in a variety of formats like PDF, HTML, RTF, Microsoft Excel files.
- The engine allows report definitions to include charts.

Tools to generate JasperReports

- Eclipse plugin for Jaspersoft studio
- iReport
- Jaspersoft Studio

Lifecycle of JasperReports
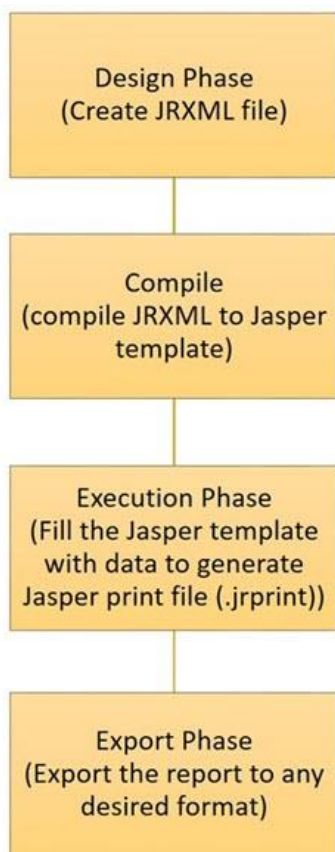
Designing the report:
This will be the first step where we create the JRXML file. It is the file is a simple XML document which defines the report layout. You can develop this file using any text editor, or report designer.

Compiling the report:
In this phase, the JRXML file created in the first phase is compiled, and a binary object with .jasper extension is generated. Jasper files are shipped along with the application to run the reports.

Executing the report:
n this step, data from the application is sent to the compiled report. A class called net.sf.jasperreports.engine.JasperFillManager offers the necessary functions to fill the data in the reports. Then a jasper print file is created, which can be used to print or export the report.

Exporting the report to the desired format:
It is the last phase of JasperReport lifecycle in which we need to export the Jasper print file using the class JasperExportManager.
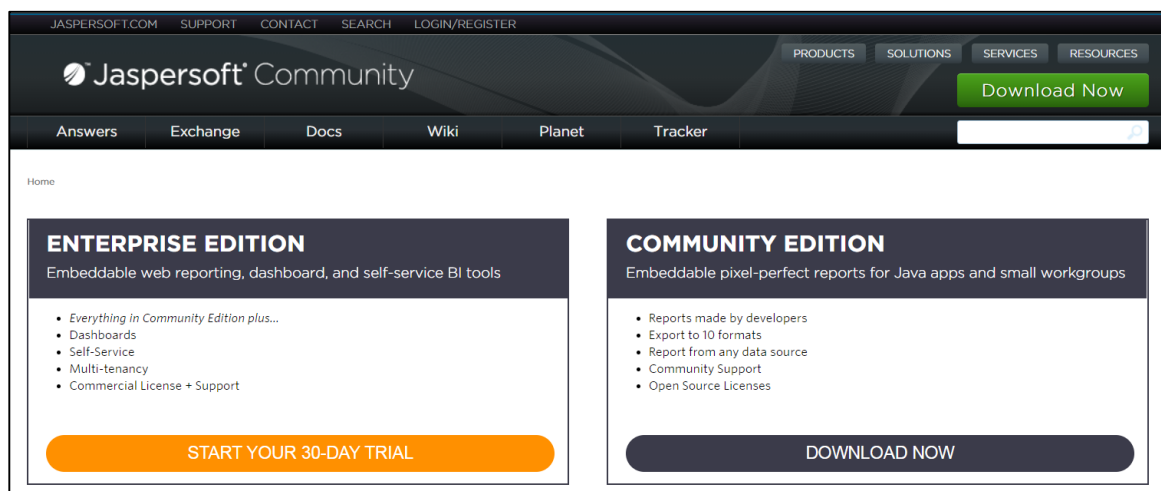
JasperReports is a pure Java library and not a standalone application.

It cannot run on its own, hence it needs to be embedded into another client or server-side Java application. As it is Java based, it can be run on any platform that supports Java (JDK 1.3 and above).

JasperReports Installation Steps

1. Go to download link : https://community.jaspersoft.com/download

2. Click **Login/Register** to download

3. Add the required fields to register account.

4. Go to your gmail to verify account and click the link in your gmail.

5. Login with your email and password.

6. Choose **Community Edition** to download:

ACE Inspiration
Professional Development Center

7. Choose **Jaspersoft Studio** to download **:**

8. Choose the file to download based on the OS :

9. Install Jaspersoft studio.

What is Report?

A report is a meaningful, well-defined, and summarized presentation of information.

Usually, the routine activities are automated and data summarized into a decision-supporting "Reports".

Reports represent usual messy data into charts, graphs, and other forms of graphical representations.

Report Template

Generally, the following layout is adopted to generate reports by most of the commercial report generating tools.

- Title
- Page header
- Column header
- Detail
- Column footer
- Page footer
- Summary

Title	• Title contains the 'Title'of the report. • It appears only once at the very beginning of the report, • for example, "Tutorials Point Report."
Page Header	• Page Header may contain date and time information and/or organization name. • This appears at the top of each page.
Column Header	• Column Header lists the names of those specific fields, which you want to display in the report, • for example, "StudentName", "Age", "Total Student", etc.
Detail	• Detail is the part where entries of the specific fields
Column Footer	• Column Footer may display summary of any of the field, • for example, "Total Student" : "100."
Page Footer	• Page Footer may contain page count information. • It appears at the bottom of each page, • for example, "1/23."
Summary	• Summary contains information inferred from "detail" part.

ACE Inspiration
Professional Development Center
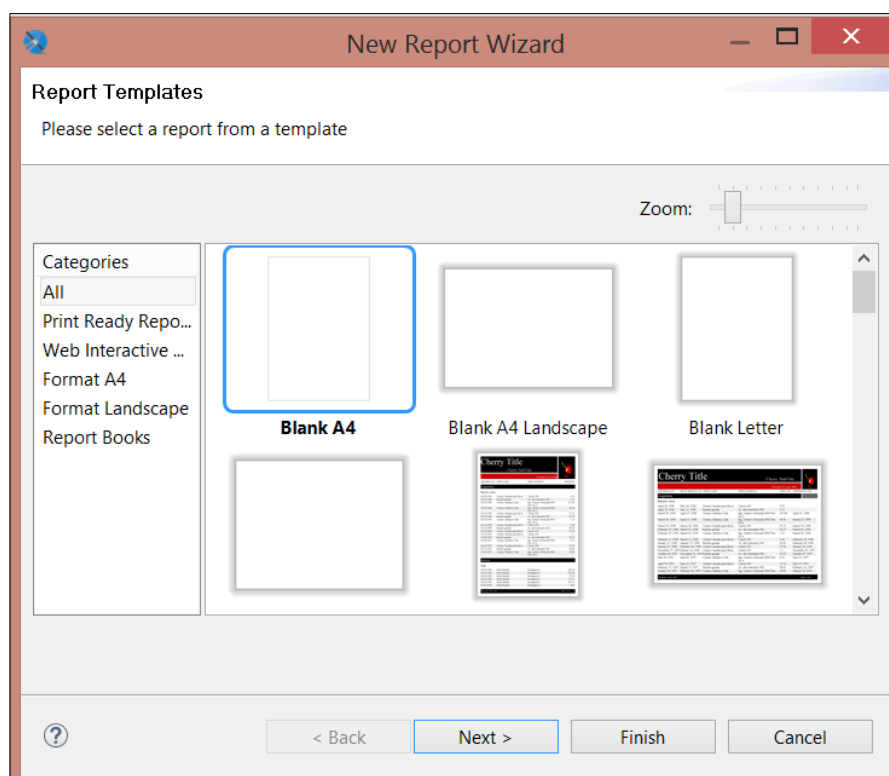
How to create a Report Template

A report is designed and created using the below-mentioned steps –

Step 1. Create JRXML file in Jaspersoft studio

- Click on the file option
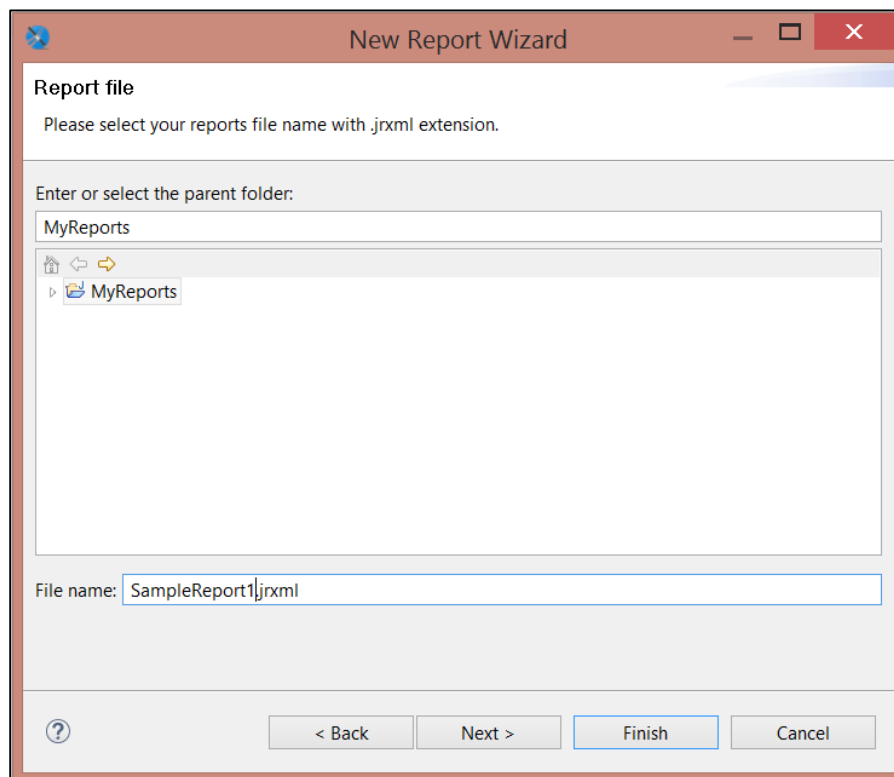- Then, select New in the dropdown menu
- Click on JasperReport option

Step 2. Selecting the desired template.

- Select a template – **Blank A4**.
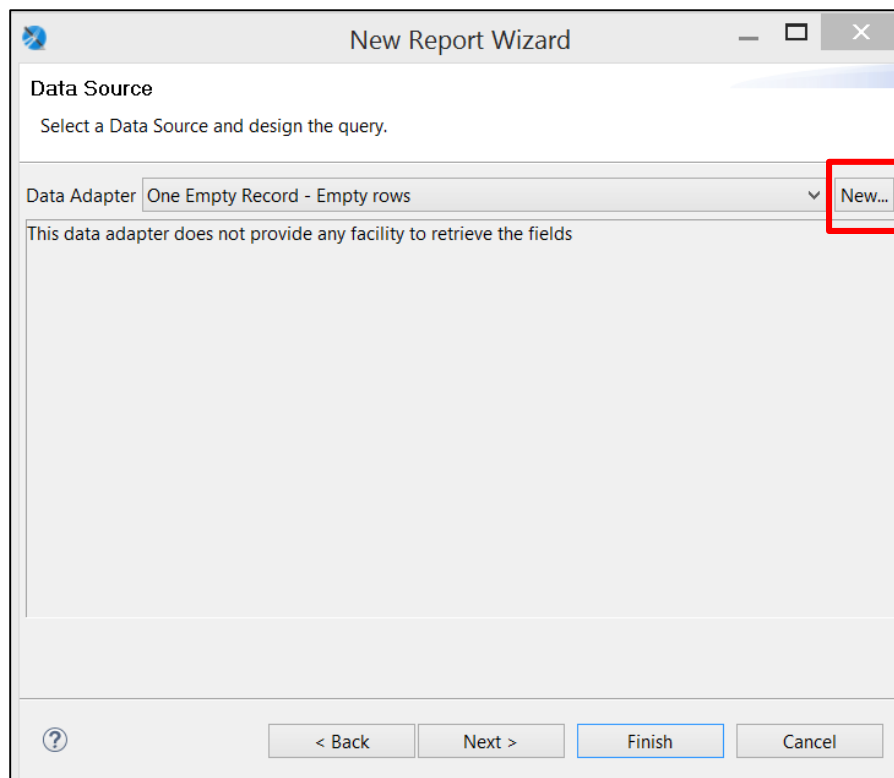- Click on the "Next" button.

Step 3. Selecting a parent folder.

- Go to MyReports folder.
- Enter the file name.
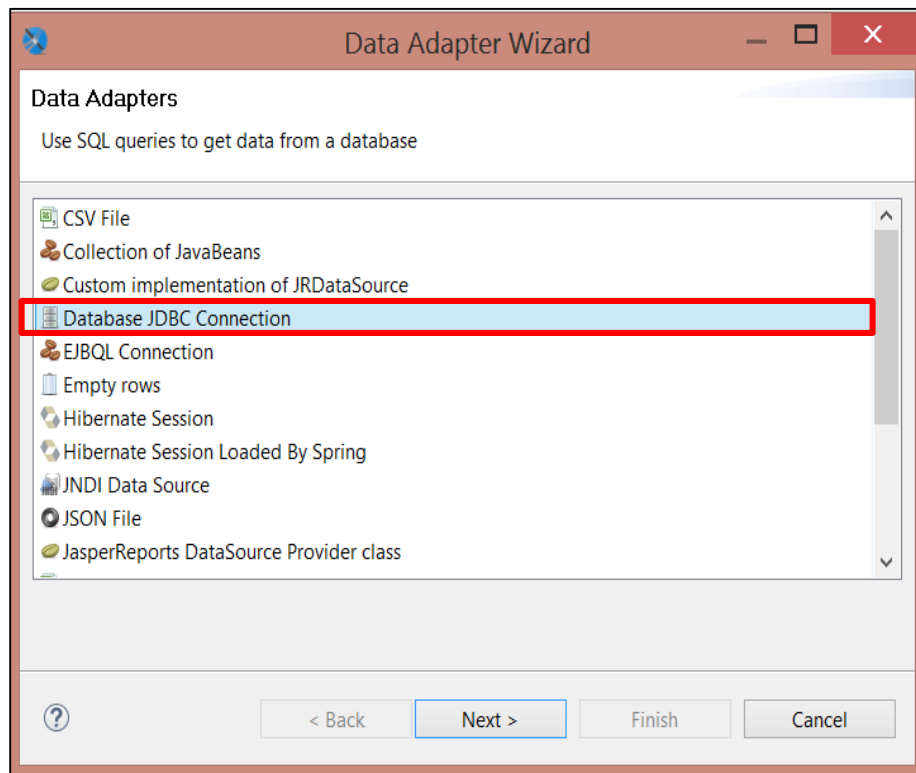- Click on the "Next" button.

Step 4. In the Data Source window, tap on "New" button and create a new data adapter.
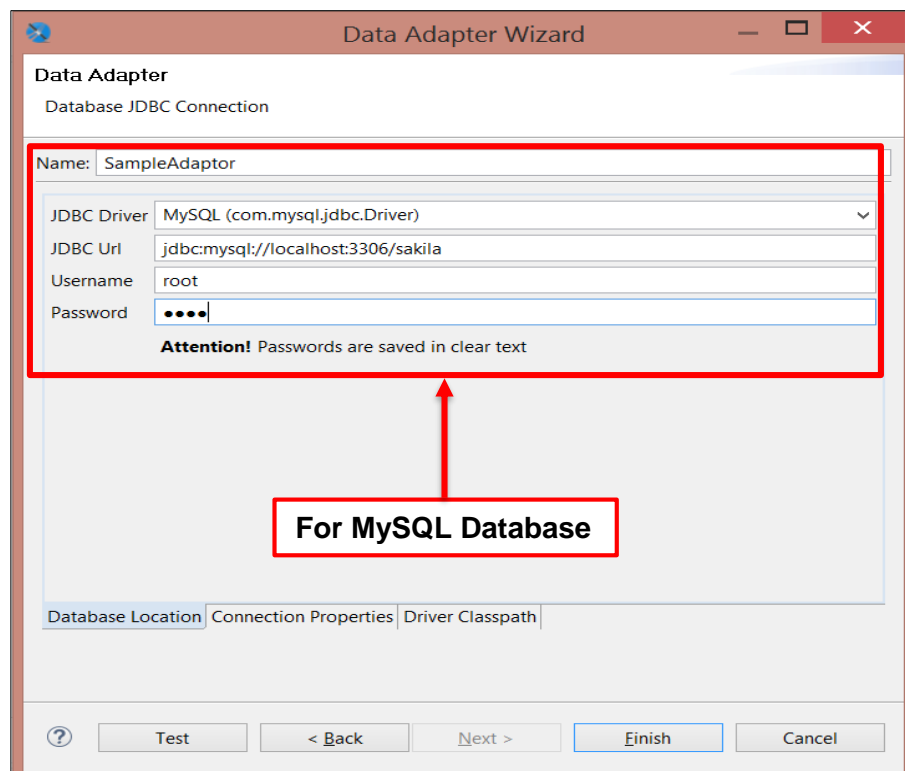
ACE Inspiration
Professional Development Center

Step 5. Select a Data Adapter.

- Select the data adapter from the list box – **Database JDBC Connection**
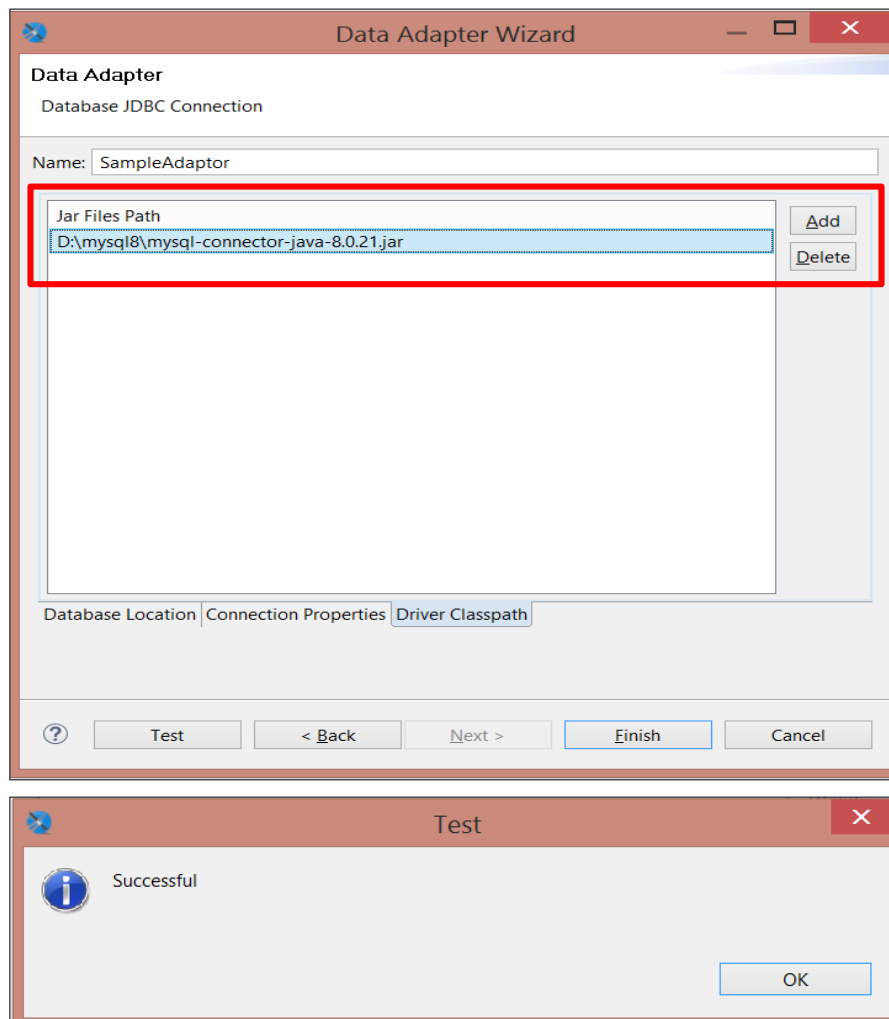- Click on the "Next" button.

Step 6. Fill up data adapter details.

- Enter a name for your adapter.
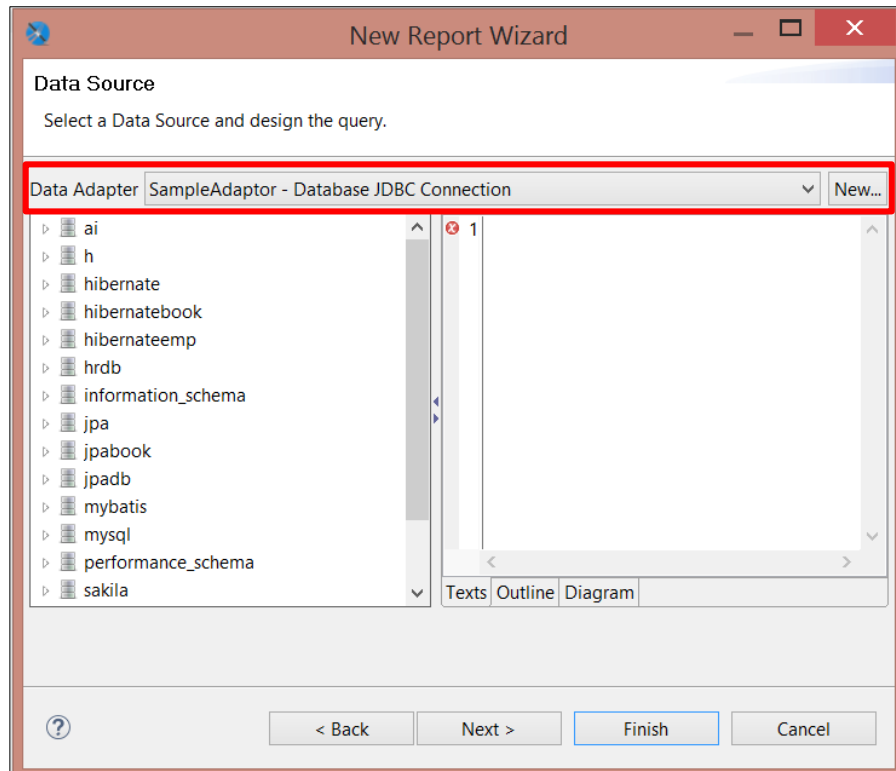- Fill up database properties.

Step 7. Select jar file in the Driver Classpath.

- Click Add button.
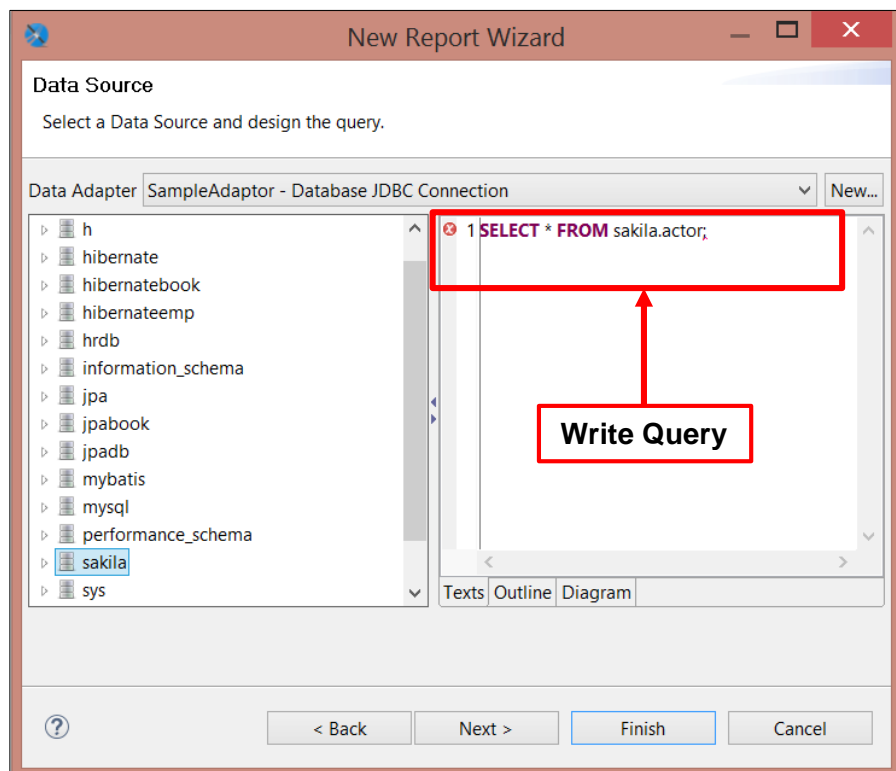- Choose database jar file.
- Click Test button.

Step 8. Congratulations! You are done with adding all important report creation information successfully. Now, click on the "Finish" button.
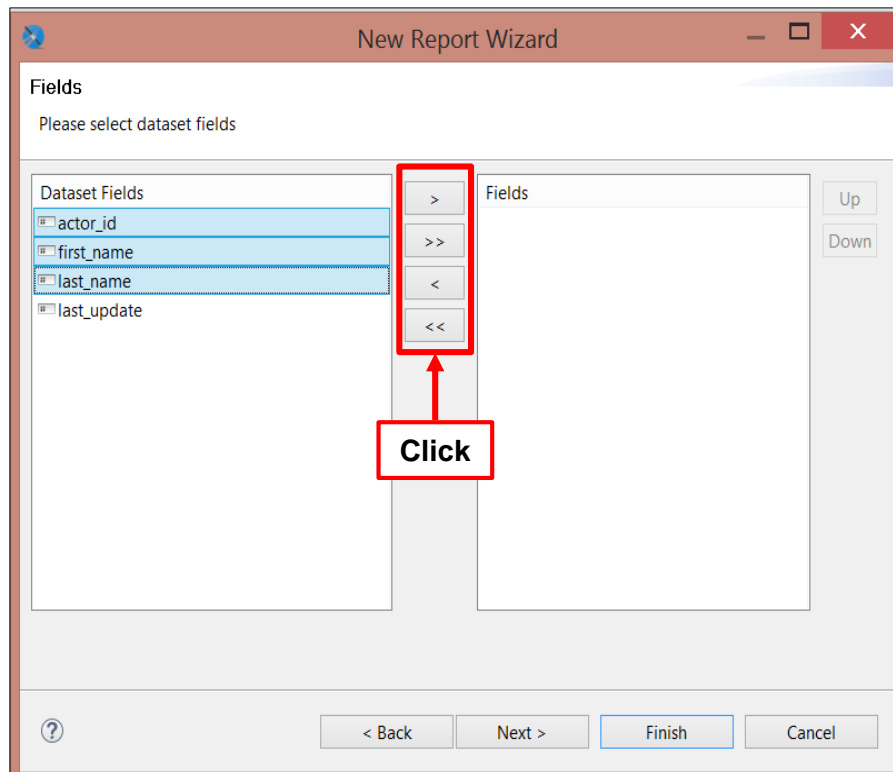
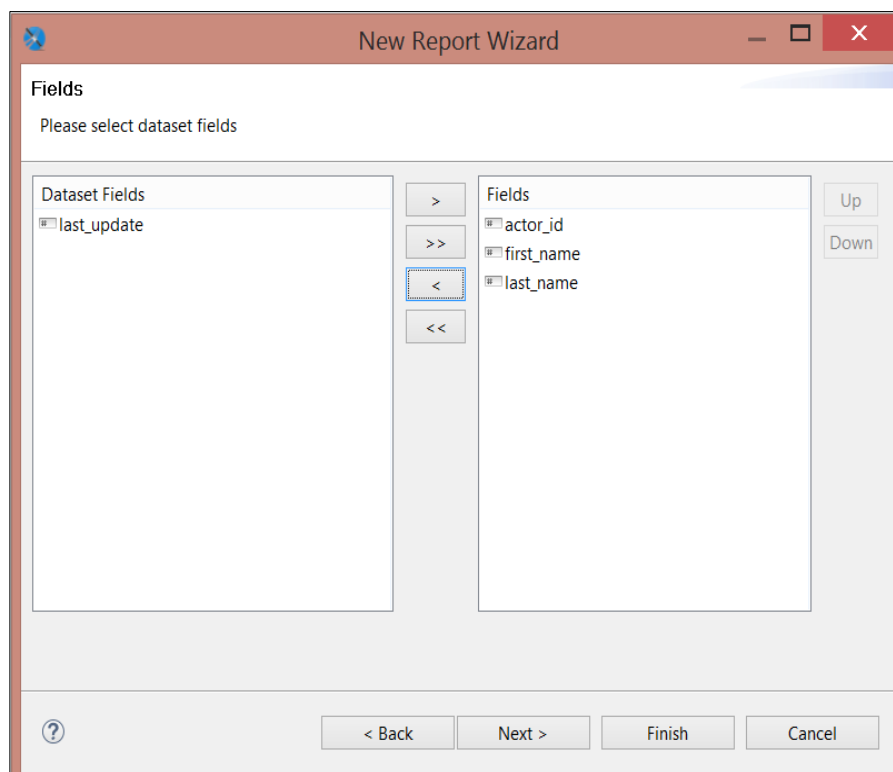**Step 9.**

Step 10. Write Query for the report, then click "Next" button.

ACE Inspiration
Professional Development Center

Step 11. Select Dataset fields to show in the report, then click "Next" button.

Step 12. Click "Next" button.

Step 13. Select field to group by clause, then click "Next" button.

Step 14. Click "Finish" button.

Step 15. The following is the generated report with Database query.
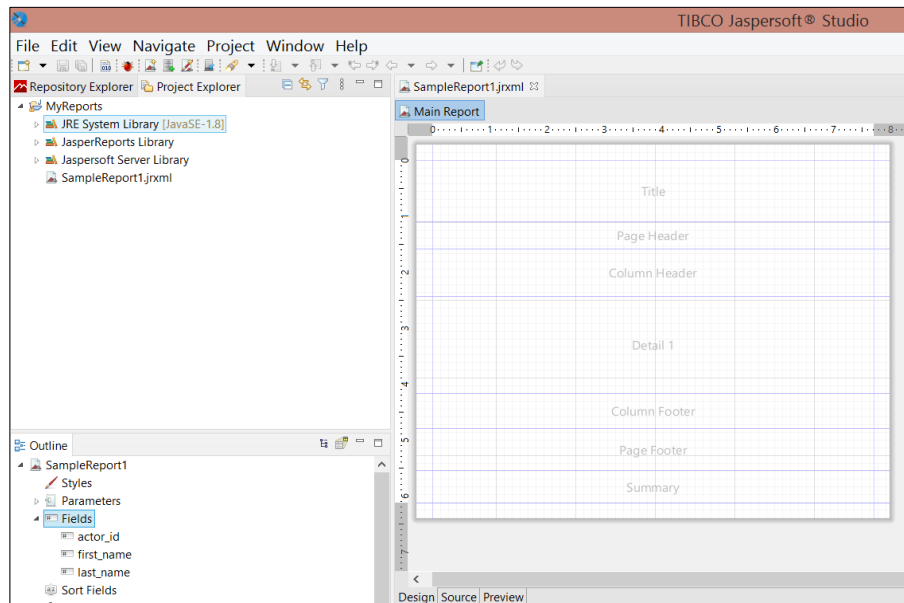
Step 16.

- Drag **"Static Text"** in the Basic Elements to show in the Column Header.

- Change style in the **Properties** window.
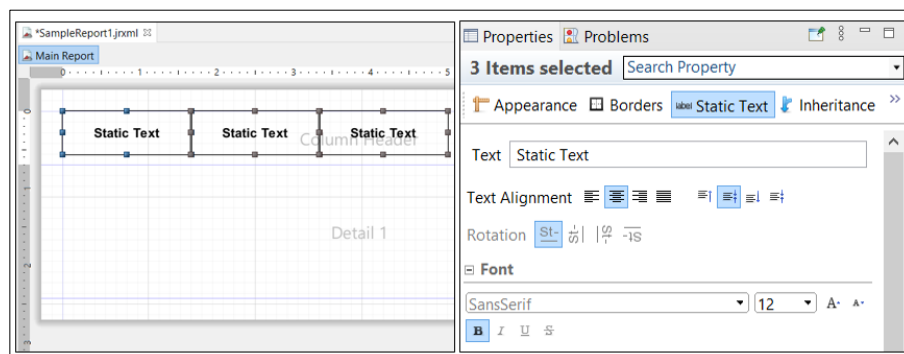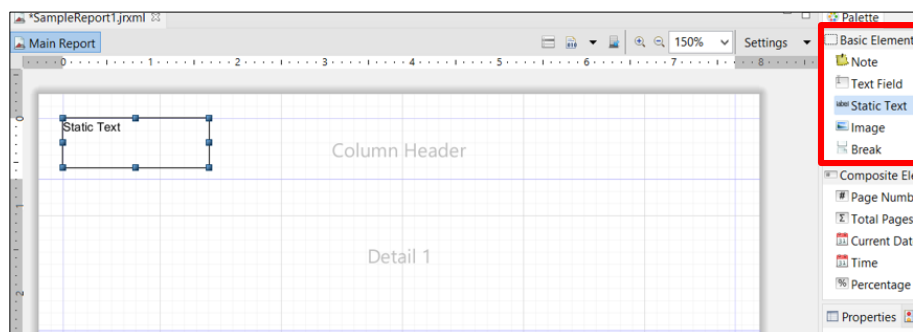
Step 17.

- Drag **"Text Field"** in the Basic Elements to show in the Detail.
- Change style in the **Properties** window.
- Click **browse button in the Expression** of Text Field to show dataset fields in the related Text Field.
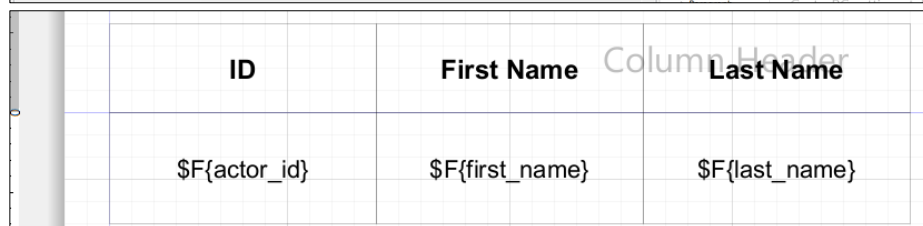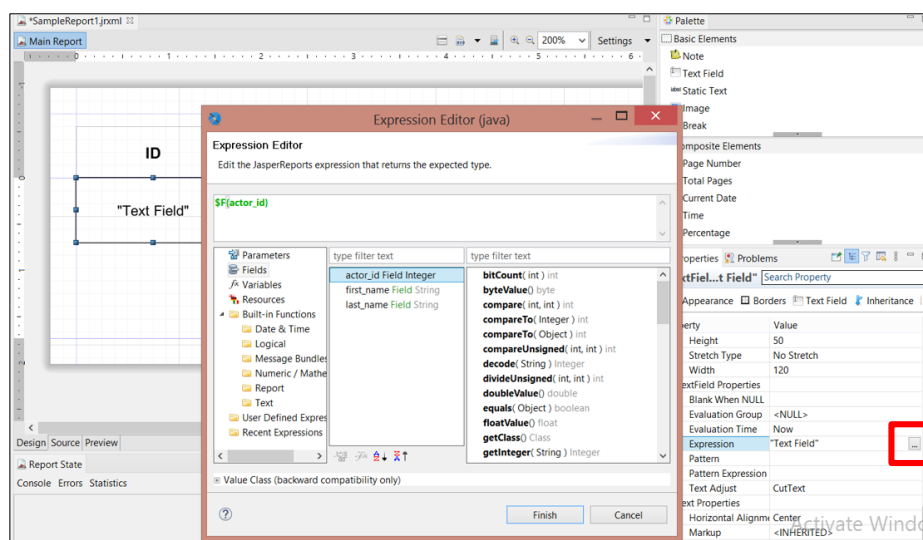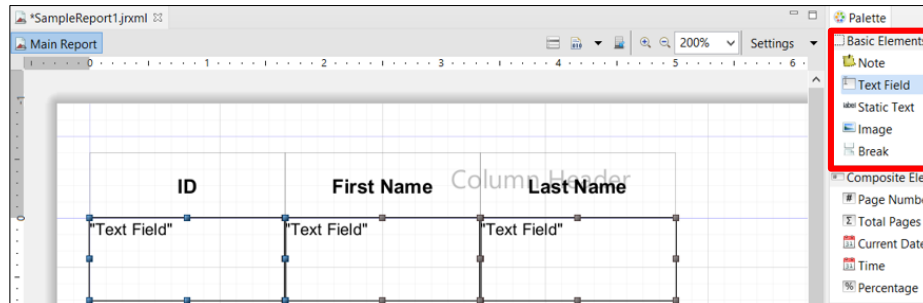
Step 18. SampleReport1.jasper is generated when jrxml is compiled.

15
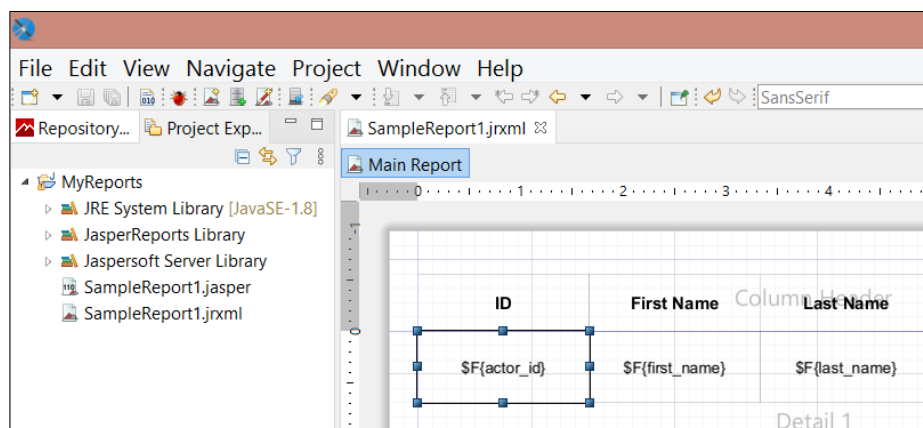
Step 19.

- Add "Actor List" in the Static Text for Title.
- Add Date in the Page Header.
- Add Static Text for Column Header.
- Add Text Field for Detail.
- Add Page number for Page Footer.

Step 20. Final Report is generated with the database query.

To use JasperReports with Java, there are some steps:

Step 1: Download JasperReports Library

Step 2: Download MySQL jar

Step 3: Create Java Web Project

Step 4: Prepare Report

Step 1: Download JasperReports Library

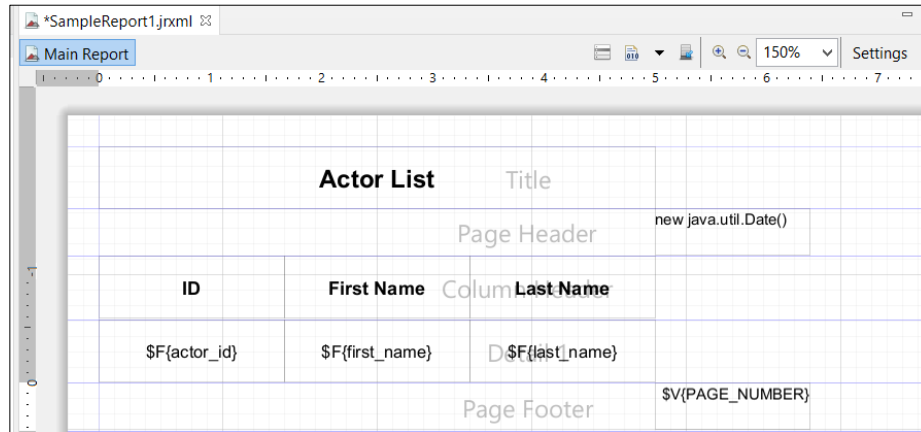|			
bcprov-jdk15on-1.64.jar	3/13/2021 9:58 PM	Executable Jar File	4,660 KB
castor-core-1.4.1.jar	3/13/2021 9:58 PM	Executable Jar File	47 KB
castor-xml-1.4.1.jar	3/13/2021 9:58 PM	Executable Jar File	890 KB
commons-beanutils-1.9.4.jar	3/13/2021 9:58 PM	Executable Jar File	242 KB
commons-collections-3.2.2.jar	3/13/2021 9:58 PM	Executable Jar File	575 KB
commons-collections4-4.2.jar	3/13/2021 9:58 PM	Executable Jar File	736 KB
commons-digester-2.1.jar	3/13/2021 9:58 PM	Executable Jar File	193 KB
commons-lang3-3.4.jar	3/13/2021 9:58 PM	Executable Jar File	425 KB
commons-logging-1.1.1.jar	3/13/2021 9:58 PM	Executable Jar File	60 KB
ecj-3.21.0.jar	3/13/2021 9:58 PM	Executable Jar File	2,920 KB
itext-2.1.7.js8.jar	3/13/2021 9:58 PM	Executable Jar File	1,106 KB
jackson-annotations-2.10.0.jar	3/13/2021 9:58 PM	Executable Jar File	67 KB
jackson-core-2.10.0.jar	3/13/2021 9:58 PM	Executable Jar File	341 KB
jackson-databind-2.10.0.jar	3/13/2021 9:58 PM	Executable Jar File	1,369 KB
jasperreports-6.16.0.jar	3/13/2021 9:58 PM	Executable Jar File	5,475 KB
javax.inject-1.jar	3/13/2021 9:58 PM	Executable Jar File	3 KB
jcommon-1.0.23.jar	3/13/2021 9:58 PM	Executable Jar File	323 KB
jfreechart-1.0.19.jar	3/13/2021 9:58 PM	Executable Jar File	1,529 KB
poi-3.17.jar	3/14/2021 2:12 PM	Executable Jar File	2,638 KB
poi-ooxml-3.17.jar	3/14/2021 2:03 PM	Executable Jar File	1,445 KB

Step 2: Download MySQL jar

mysql-connector-java-8.0.21.jar	6/16/2020 5:17 PM	Executable Jar File	2,342 KB

ACE Inspiration
Professional Development Center

Step 3: Create Java Web Project

1: Project Structure

```
▲ 🗂 bookmanagementdb
   ▷ 🗄 Deployment Descriptor: bookmanagementdb
   ▷ 🗄 JAX-WS Web Services
   ▲ 🗄 Java Resources
      ▲ 🗄 src
         ▲ 🗄 bookmanagement.controller
            ▷ 🗄 AddBookServlet.java
            ▷ 🗄 DeleteBookServlet.java
            ▷ 🗄 DisplayServlet.java
            ▷ 🗄 ReportServlet.java
            ▷ 🗄 UpdateBookServlet.java
         ▷ 🗄 bookmanagement.model
         ▷ 🗄 bookmanagement.persistant.dao
         ▷ 🗄 bookmanagement.persistant.dto
      ▷ 🗄 Libraries
   ▷ 🗄 JavaScript Resources
   ▷ 🗄 Referenced Libraries
   ▷ 🗄 build
   ▲ 🗄 WebContent
      ▷ 🗄 META-INF
      ▲ 🗄 WEB-INF
         ▲ 🗄 jasper
               🗄 BookReport.jasper
               🗄 BookReport.jrxml
         ▷ 🗄 lib
            🗄 web.xml
      🗄 addbook.jsp
      🗄 displaybook.jsp
      🗄 header.html
      🗄 updatebook.jsp
```

2: BookBean.java

```java
public class BookBean {
        private String bookCode;
        private String bookTitle;
        private String bookAuthor;
        private String bookPrice;
// getters, setters

}
```

3: ReportServlet.java

```java
protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
                ServletContext context = request.getServletContext();
        String path = context.getRealPath("/WEB-INF/jasper/BookReport.jrxml");
                JRBeanCollectionDataSource source = null;
                JasperReport jasperReport;
                JasperPrint print;
                ArrayList<BookResponseDTO> list = null;
                if (request.getServletContext().getAttribute("list") != null) {
list = (ArrayList<BookResponseDTO>) request.getServletContext().getAttribute("list");
                }
                // Parameters for report
                Map parameters = new HashMap();
                parameters.put("ReportTitle", "Book List");
try {

                        source = new JRBeanCollectionDataSource(list);
                        jasperReport = JasperCompileManager.compileReport(path);
            print = JasperFillManager.fillReport(jasperReport, parameters, source);


                        if (export.equals("excel")) {
                                response.setContentType("application/vnd.ms-excel");
response.setHeader("Content-Disposition", "attachment; filename=book.xls");
 // final OutputStream

        JRXlsExporter exporterXLS = new JRXlsExporter();
        exporterXLS.setParameter(JRXlsExporterParameter.JASPER_PRINT, print);
        exporterXLS.setParameter(JRXlsExporterParameter.OUTPUT_STREAM,
response.getOutputStream());
        exporterXLS.setParameter(JRXlsExporterParameter.IS_ONE_PAGE_PER_S
HEET, Boolean.TRUE);
        exporterXLS.setParameter(JRXlsExporterParameter.IS_WHITE_PAGE_BACK
GROUND, Boolean.FALSE);
        exporterXLS.setParameter(JRXlsExporterParameter.IS_REMOVE_EMPTY_S
PACE_BETWEEN_ROWS, Boolean.TRUE);
        exporterXLS.setParameter(JRXlsExporterParameter.IS_DETECT_CELL_TYP
E, Boolean.TRUE);
exporterXLS.exportReport();
} else {
                response.setContentType("application/pdf");
response.setHeader("Content-Disposition", "attachment; filename=book.pdf");
                JRPdfExporter exporterPdf = new JRPdfExporter();
        exporterPdf.setParameter(JRPdfExporterParameter.JASPER_PRINT, print);
        exporterPdf.setParameter(JRPdfExporterParameter.OUTPUT_STREAM,
response.getOutputStream());
        exporterPdf.exportReport();
                }

                } catch (JRException e) {
                        e.printStackTrace();
                }
        }
```

4: displaybook.jsp

```jsp
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
<style type="text/css">
        table,th,td{
                border: 1px solid blue;
                border-collapse:collapse;
        }
</style>
</head>
<body>

<jsp:include page="header.html"></jsp:include>
<br>
<div style="color:green;">${msg}</div>
<br>
<p>
<a href="ReportServlet?export=excel">excel</a>
<a href="ReportServlet?export=pdf">pdf</a>
</p>
<table>
        <tr>
                <th>Book Code</th>
                <th>Book Title</th>
                <th>Book Author</th>
                <th>Book Price</th>
                <th>Action</th>
        </tr>
        <c:forEach items="${applicationScope.list}" var="data">
                <tr>
                        <td>${data.bookCode}</td>
                        <td>${data.bookTitle}</td>
                        <td>${data.bookAuthor}</td>
                        <td>${data.bookPrice}</td>
                        <td>
                                <a
href="UpdateBookServlet?code=${data.bookCode}">Update</a>|
                                <a
href="DeleteBookServlet?code=${data.bookCode}">Delete</a>
                        </td>

                </tr>
        </c:forEach>
</table>
</body>
</html>
```
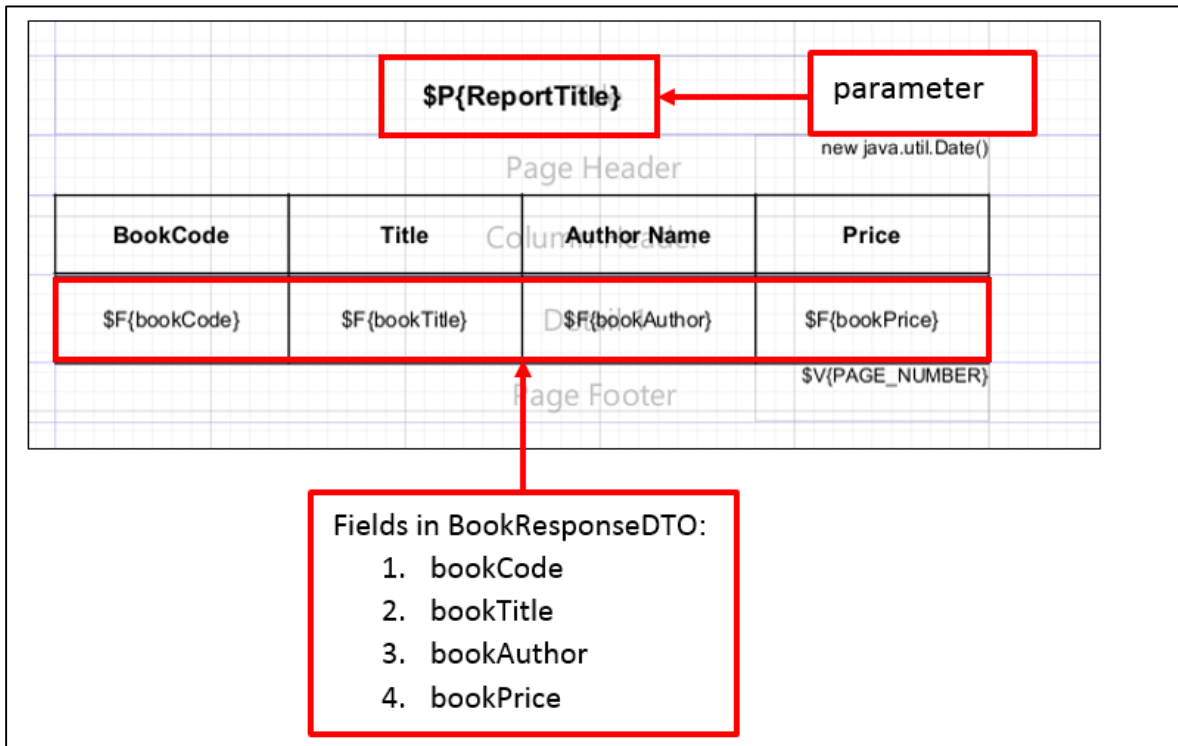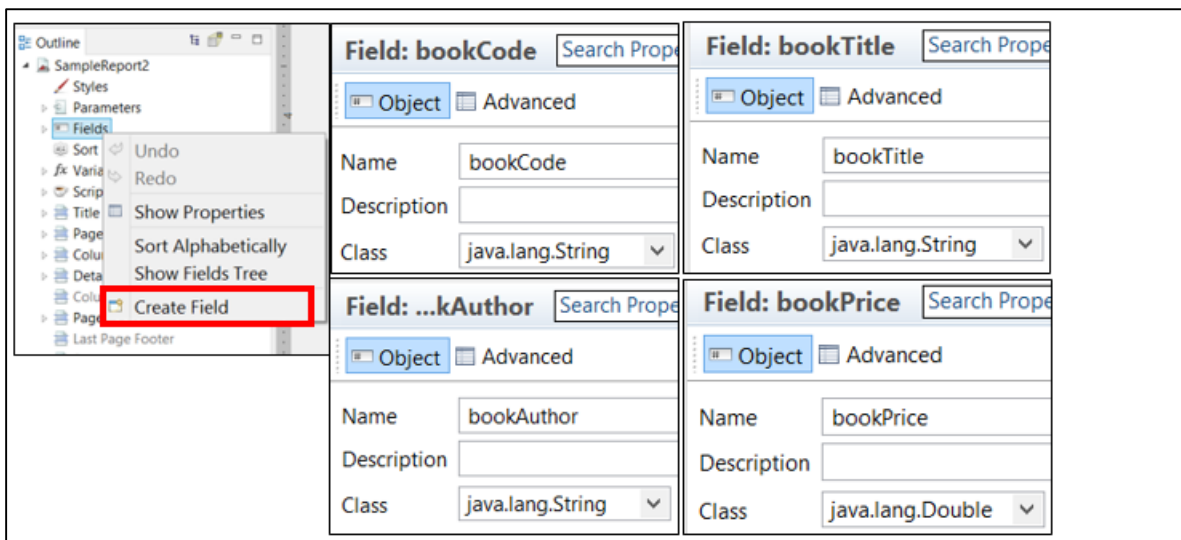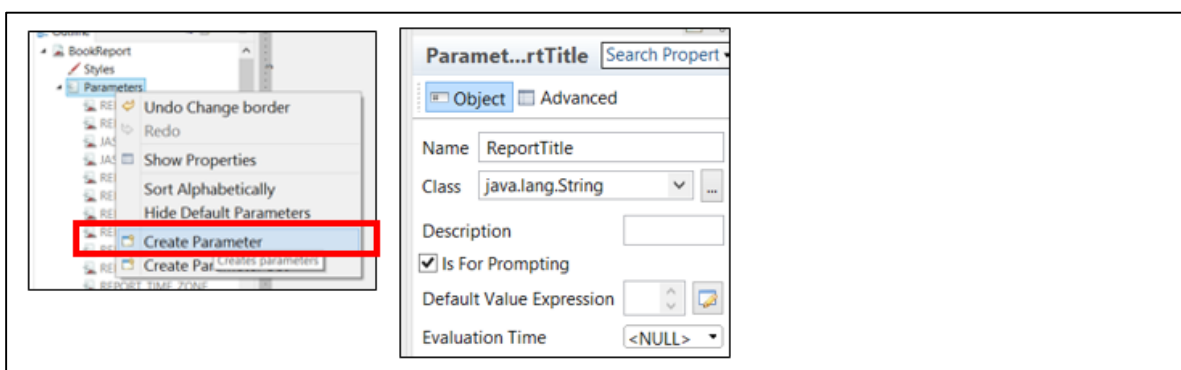
Step 4: Prepare Report

1: BookReport.jrxml

2: To add Fields in Report:

3: To add Parameters in Report:

Output:

1: index page

Book Management

Add New Book   Show All Book

2: When click **"Show All Book"**,

Book Management

Add New Book   Show All Book

excel pdf

Book Code	Book Title	Book Author	Book Price	Action
b001	Java	James	20000.0	Update| Delete
b002	PHP	Smith	20000.0	Update| Delete
b003	Asp.Net	Goy	20000.0	Update| Delete

2: When click **"excel"**,

Book List

1/26/2022

BookCode	Title	Author Name	Price
b001	Java	James	20000
b002	PHP	Smith	20000
b003	Asp.Net	Goy	20000

1

3: When click **"pdf"**,

Book List

January 26, 2022

BookCode	Title	Author Name	Price
b001	Java	James	20000.0
b002	PHP	Smith	20000.0
b003	Asp.Net	Goy	20000.0

ACE Inspiration
Professional Development Center