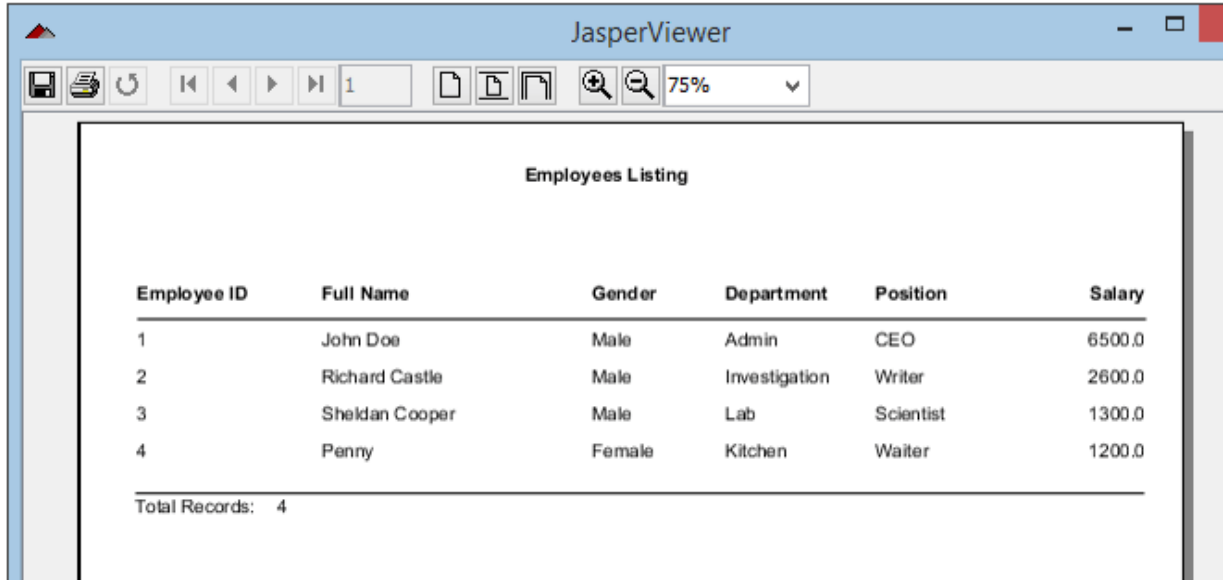


BÀI TẬP TUẦN 6

JasperReports JDBC Datasource Tutorial



Employee ID	Full Name	Gender	Department	Position	Salary
1	John Doe	Male	Admin	CEO	6500.0
2	Richard Castle	Male	Investigation	Writer	2600.0
3	Sheldan Cooper	Male	Lab	Scientist	1300.0
4	Penny	Female	Kitchen	Waiter	1200.0

Total Records: 4

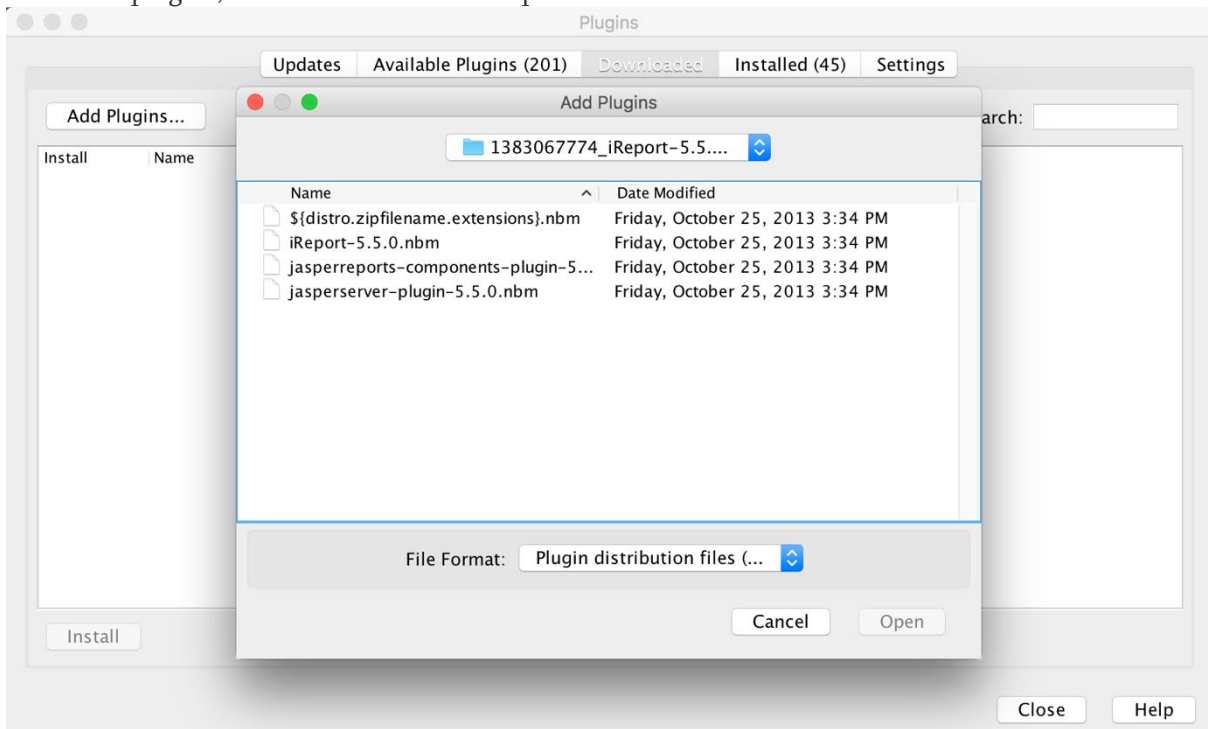
1. Getting Started

Download the JasperReports plugin for NetBeans IDE using the link below [NetBeans IDE iReport Plugin Download](#)

Install the plugin and it will be integrated into NetBeans IDE.

Click menu Tool > Plugin > Downloaded

Click Add plugins, add all files in Folder iReports – 5.5



Adding JasperReports jar files to a NetBeans IDE Java Project.

Right click on Libraries node Select add Jar/Folder option Browse to the libs folder of the iReport directory.

Download and Add file jasperreports-5.6.0.jar

<https://sourceforge.net/projects/jasperreports/files/jasperreports/JasperReports%205.6.0/jasperreports-5.6.0.jar/download>

Download and Add the following jar files:

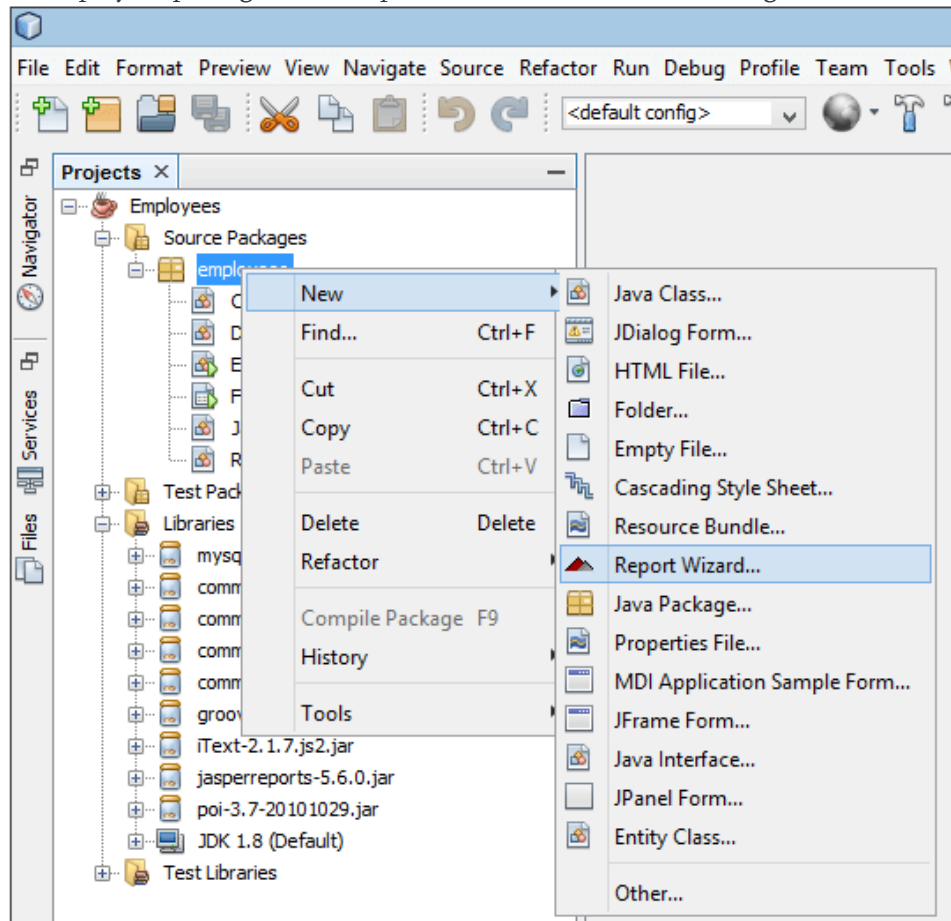
<https://sourceforge.net/projects/jasperreports/files/jasperreports/JasperReports%205.6.0/jasperreports-5.6.0-project.zip/download>

- commons-beanutils-1.8.0.jar
- commons-collections-3.2.1.jar
- commons-digester-2.1.jar
- commons-logging-1.1.1.jar
- groovy-all-2.0.1.jar
- iText-2.1.7.js2.jar
- poi-3.7-20101029.jar

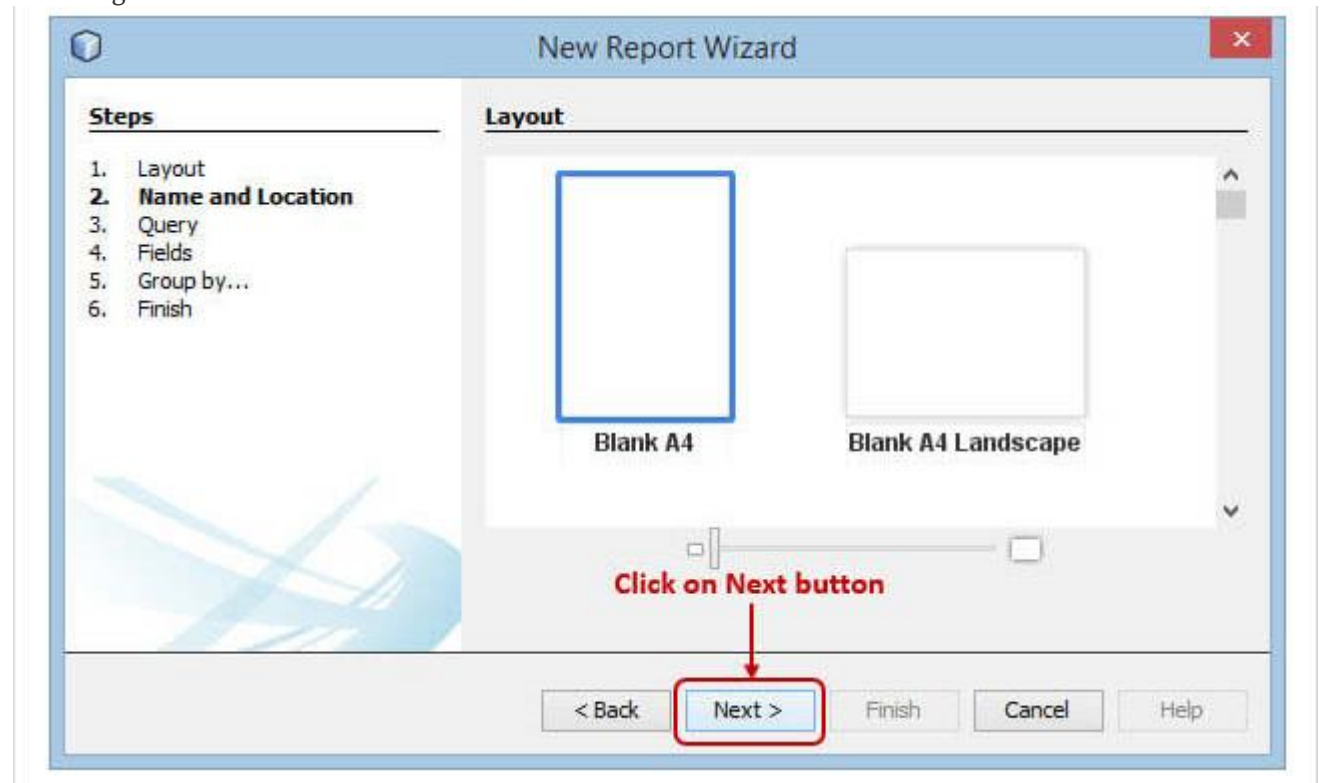
Note: the file versions may be different from yours depending on the version of iReports that you download.

2. NetBeans IDE Creating JasperReports

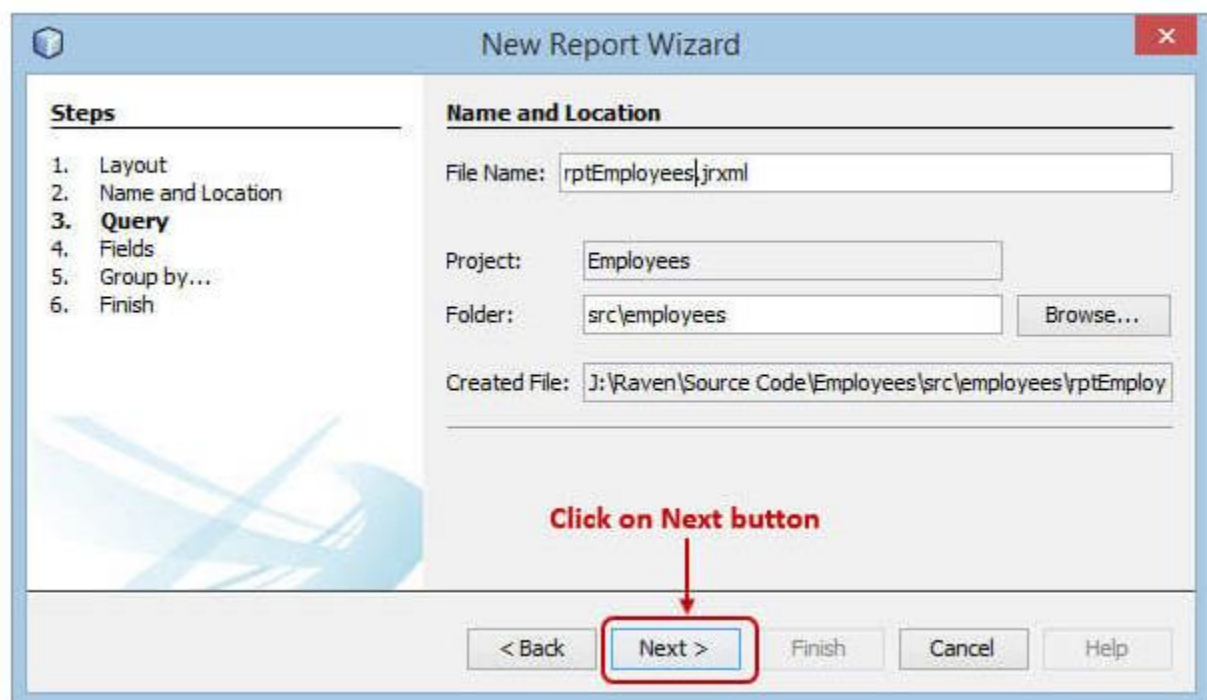
Right click on employees package Select Report Wizard as shown in the image below



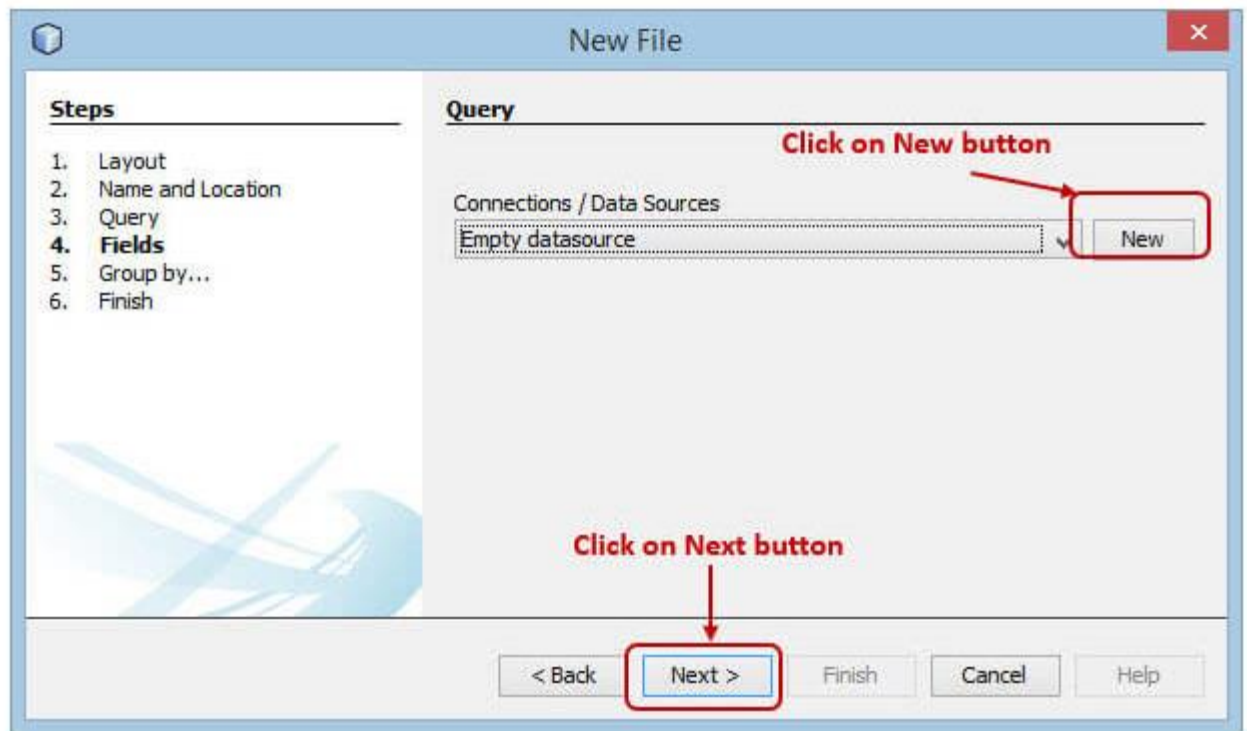
Note: if Report Wizard is not showed, select Other... option. You will get it from there You will get the following window



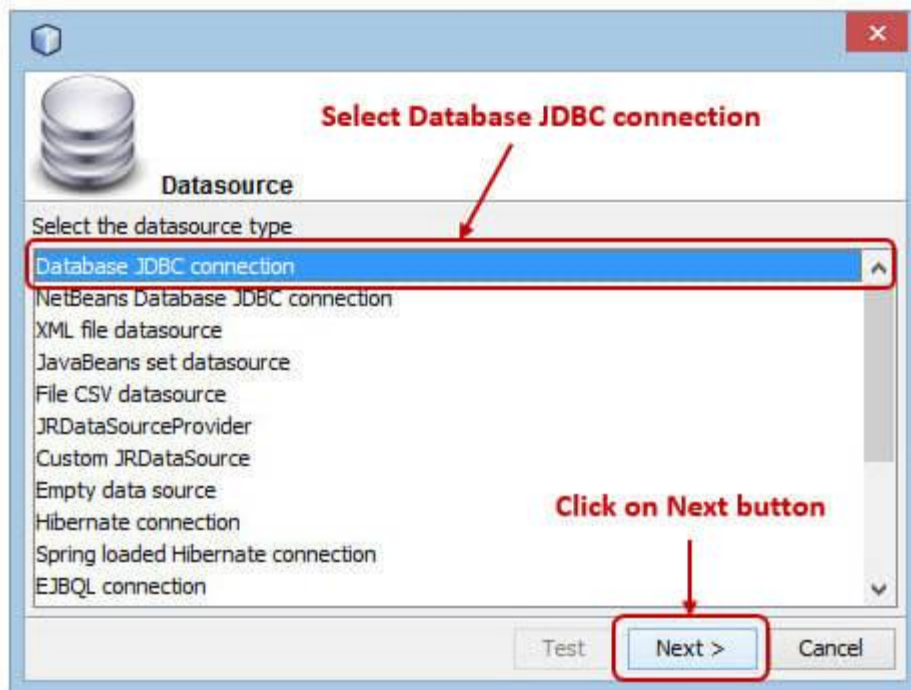
Choose Blank A4 or other Templates. Click on Next button



Enter rptEmployees.jrxml as the file name and extension Click on Next button



Click on New button to create a new data Source



Click on Next Button

Enter the required parameters as shown below

Database JDBC connection

Name: employees link

JDBC Driver: MySQL (com.mysql.jdbc.Driver)

JDBC URL: jdbc:mysql://localhost/employees

Credentials

Username: root

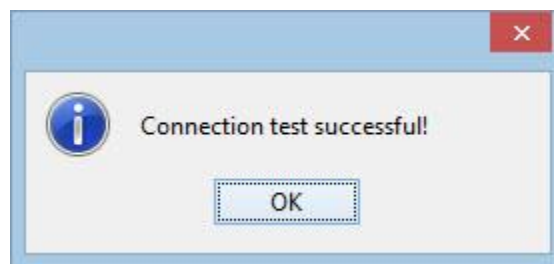
Password:

☒ Save password

Click on Next button

Test Save Cancel

Click on Test button



Click on save button Enter the following query

New File

Steps

1. Layout
2. Name and Location
3. Query
4. **Fields**
5. Group by...
6. Finish

Query

Connections / Data Sources

employees link

Click on New button

Query (SQL)

SELECT * FROM employees

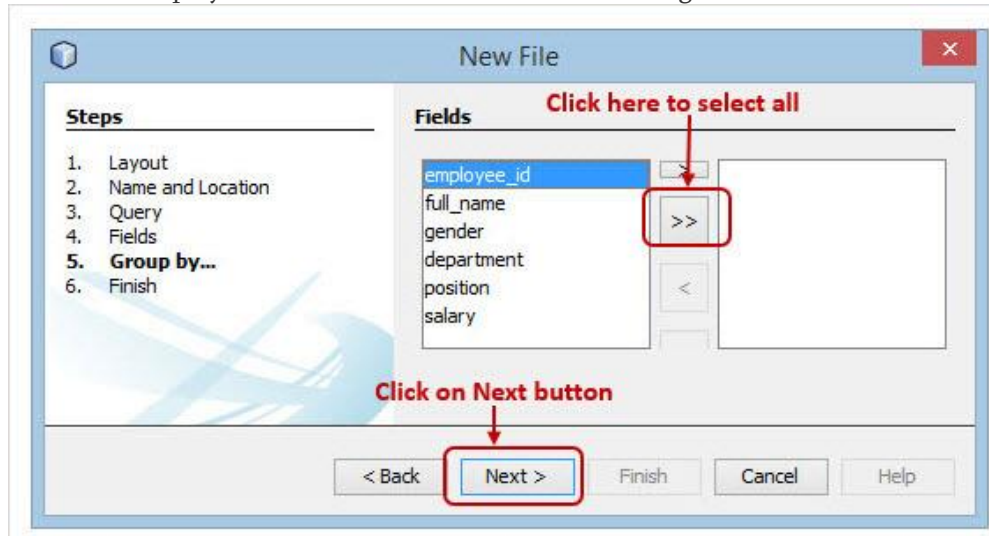
Design query Load query Save query

Click on Next button

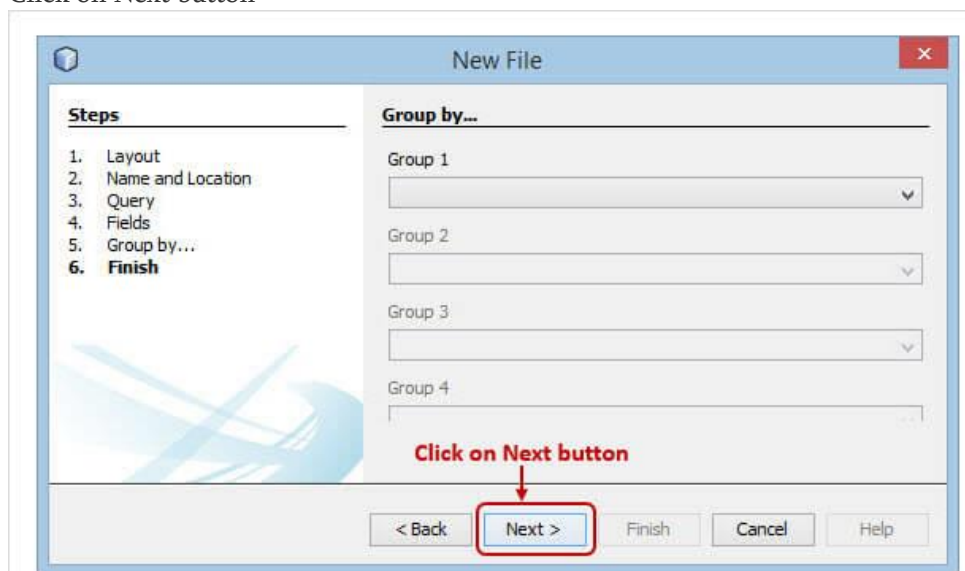
< Back Next > Finish Cancel Help

SELECT * FROM employees

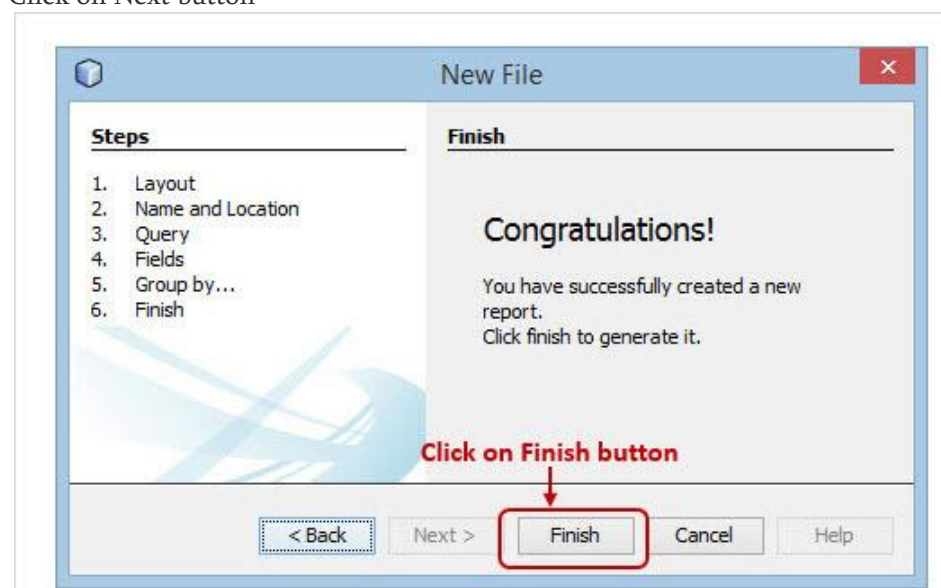
Make sure employees link is selected as shown in the image above Click on Next button



Click on Next button



Click on Next button



Click on Finish button

3. Designing a JasperReport in NetBeans IDE

Double click on rptEmployees.jrxml to open the report designer Press Ctrl + Shift + 8 to open the Report palette

Design the report as shown below

The screenshot shows the JasperReport Designer interface. The report is titled 'Employees Listing'. It features a 'Page Header' section with the title. Below this is a 'Column Header' section with a table structure. The table has six columns: 'Employee ID', 'Full Name', 'Gender', 'Department', 'Position', and 'Salary'. Each column contains a field reference: '\$F{employee_id}', '\$F{full_name}', '\$F{gender}', '\$F{department}', '\$F{position}', and '\$F{salary}' respectively. Below the column header is a 'Column Footer' section. At the bottom is a 'Page Footer' section containing 'Page \$V of \$V'. A 'Summary' section at the very bottom shows 'Total Records: \$V'.

Employees Listing					
Employee ID	Full Name	Gender	Department	Position	Salary
\$F{employee_id}	\$F{full_name}	\$F{gender}	\$F{department}	\$F{position}	\$F{salary}
Total Records: \$V					

Note: Use the Report Inspector to add fields to the report

3.1. Create a class JasperReports.java Add the following code

```
package employees;

import java.io.InputStream;
import java.sql.SQLException;
import java.util.HashMap;
import java.util.Map;
import net.sf.jasperreports.engine.JRException;
import net.sf.jasperreports.engine.JasperCompileManager;
import net.sf.jasperreports.engine.JasperFillManager;
import net.sf.jasperreports.engine.JasperPrint;
import net.sf.jasperreports.engine.JasperReport;
import net.sf.jasperreports.engine.design.JRDesignQuery;
import net.sf.jasperreports.engine.design.JasperDesign;
import net.sf.jasperreports.engine.xml.JRXmlLoader;
import net.sf.jasperreports.view.JasperViewer;

public class JasperReports {

    String m_where;
    String m_report_source = "/employees/";
    String m_sql_stmt;
    Map parametersMap = new HashMap();

    protected void showReport() {
        try {
            DBUtilities dbUtilities = new DBUtilities();
            InputStream is =
                getClass().getResourceAsStream(m_report_source);

            JRDesignQuery jrDesignQuery = new JRDesignQuery();
            jrDesignQuery.setText(m_sql_stmt);
```

```

        JasperDesign jasperDesign = JRXmlLoader.load(is);
        jasperDesign.setQuery(jrDesignQuery);

        JasperReport jasperReport =
JasperCompileManager.compileReport(jasperDesign);
        JasperPrint jasperPrint =
JasperFillManager.fillReport(jasperReport, parametersMap,
dbUtilities.getConnection());
        JasperViewer.viewReport(jasperPrint, false);

    } catch (SQLException | JRException e) {
        System.out.println("Exception message " + e.getMessage());
    }
}
}

```

“showReport()” this method connects to the database dynamically to load data, specifies a report to be loaded, executes a JRDesignQuery query, compiles a JasperReport and displays it

3.2. Create a class DisplayReports.java. This class extends JasperReports.java class Add the following code

```

package employees;

public class DisplayReports extends JasperReports {

    public void showEmployees() {
        m_report_source = "rptEmployees.jrxml";
        m_sql_stmt = "SELECT * FROM employees ORDER BY employee_id";
        showReport();
    }
}

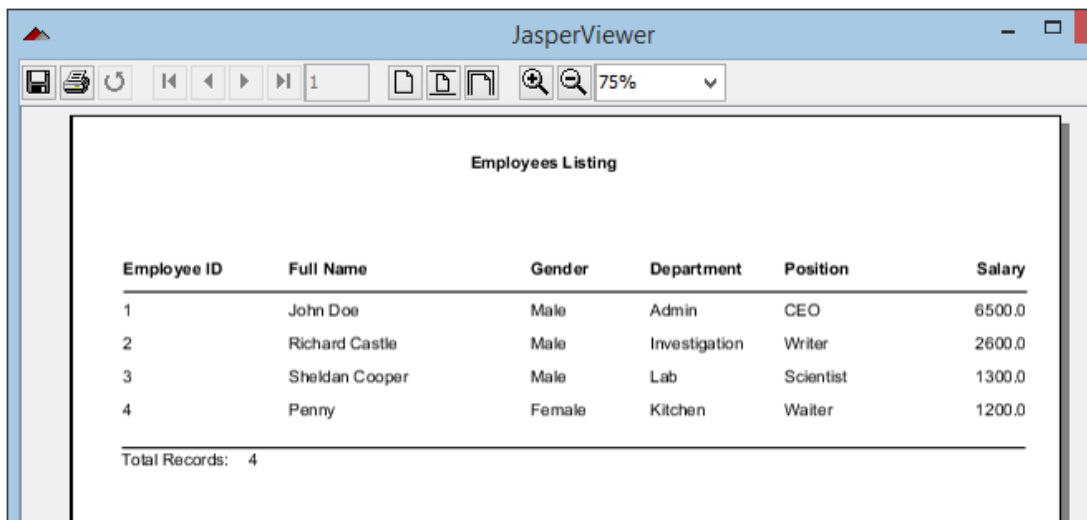
```

“showEmployees()” this methods specifies the report to be loaded and the SELECT SQL statement to be executed by JRDesignQuery.

3.3. Add the following code to btnPrintAllActionPerformed in FORMEmployees.Java

4. Testing the project

Run the project Click on Print All button You will get the following results



Employee ID	Full Name	Gender	Department	Position	Salary
1	John Doe	Male	Admin	CEO	6500.0
2	Richard Castle	Male	Investigation	Writer	2600.0
3	Sheldon Cooper	Male	Lab	Scientist	1300.0
4	Penny	Female	Kitchen	Waiter	1200.0

Total Records: 4