

 免費電子書

學習

jasper-reports

Free unaffiliated eBook created from
Stack Overflow contributors.

#jasper-
reports

[illegible]

.....	5
.....	5
Jasper.....	5
2:	7
.....	7
.....	7
Examples.....	7
.....	7
.....	8
3:	9
.....	9
Examples.....	9
IDE.....	9
JasperSoft	9
JavaJasperReport.....	9
.....	10
.....	10
.....	10
.....	10
4:	11
Examples.....	11
.....	11
.....	11
.....	11
.....	11
5: JasperReports .jrxml.jasper	12
Examples.....	12
IDE.....	12
Apache Ant.....	14
Java.....	15
Apache Maven.....	15

6: xls / xlsx	16
Examples	16
Java	16
	16
7: pdf	18
	18
Examples	18
IDE	18
JasperSoft	18
Java	19
JasperPrintjrxml	19
JasperPrintjrxml	19
	20

You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: [jasper-reports](#)

It is an unofficial and free jasper-reports ebook created for educational purposes. All the content is extracted from [Stack Overflow Documentation](#), which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official jasper-reports.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

1: jasper-reports

*JasperReports Java API*Java

- [DynamicReports](#)
- [DynamicJasper](#)

jrxml“”

JasperReports

6.3.0	2016620
6.2.0	20151111
5.6.0	2014527
5.5.0	20131024
5.0.4	2013326
5.0.0	20121112
4.8.0	2012115
4.7.0	2012-07-02
4.6.0	2012-05-21
4.5.0	2011-12-06
4.1.1	2011-04-18
4.0.0	2010-12-31
3.7.6	2010-10-27
3.7.5	2010-09-22
3.7.0	2009-12-08
3.6.0	2009-08-31
3.5.3	2009-07-29
3.5.0	2009-03-25
3.1.4	2009-02-10

3.1.2	2008-11-04
3.1.0	2008-09-17
3.0.1	2008-08-07
3.0.0	2008-05-19
2.0.5	2008-03-12
2.0.3	2007-12-12
2.0.0	2007-08-14
1.3.4	2007-06-11
1.3.0	2006-12-22
1.2.8	2006-11-14
1.2.0	2006-02-06
1.1.0	2005-10-21
1.0.3	2005-10-10
1.0.0	2005-07-20
0.6.8	2005-05-31
0.2.3	2002-02-06

IDE

Eclipse JasperSoft Studio ◦

NetBeans iReport Designer ◦

iReport Designer - [iReport Classic](#)

Examples

JasperReports

JasperReportsJava ◦ [JasperSoft JasperReports Library](#) ◦

`libjar pom.xml` ◦ [Maven/](#) ◦

Jaspersoft StudioIDE

[Jaspersoft Studio](#) JasperReports - Eclipse - iReport Designer。

iReport DesignerIDE

[iReport Designer](#) JasperReports。 5。 6。 020145;2015。

JasperReport Commuity

JasperReports Library

-
- [JasperReports](#)
-
- [JasperReports](#)
-
-
-

Bug

- [Bug Tracker](#)

jasper-reports

1. jrxml。 texteditorjrxmlIDEJasperSoft StudioiReport。
2. jrxml.jasper[JasperReport](#)。 .class.java。
3. [JasperPrint](#) .jprint
4. /JasperPrint。 pdfexcelwordhtmlcvs。

◦ isTitleNewPage="true"

isSummaryWithPageHeaderAndFooter="false"

◦

◦ 12n

```
Row 1
  detail 1
  detail 2
  detail n
Row 2
  detail 1
  detail 2
  detail n
```

◦ isFloatColumnFooter="true"

“”
◦

isSummaryNewPage="true" isSummaryWithPageHeaderAndFooter="true"

◦

◦

◦

0whenNoDataType="NoDataSection"◦

Jasper

- .jrxmlXML.jasperJasperReport.jasper
- .jasper.jrxmlJasperReport
- .jrprintJasperPrint/◦
- .jrpxmlJasperPrintXML rappresentativoJasperPrint

jasper-reports <https://riptutorial.com/zh-TW/jasper-reports/topic/3594/jasper-reports>

2:

parametersMapExpression	Map◦
subreportParameter	subreportParameterExpression ◦ ◦
connectionExpression	◦
dataSourceExpression	Datasource◦
subreportExpression	/ URIJasperReport◦
returnValue	◦ ◦

- ◦ ◦
- <subreport>◦
- [subreportExpression](#) *JasperReports Server JasperReports IDE API*◦

JasperReports Server

```
<subreportExpression><![CDATA["repo:subreport.jrxml"]]></subreportExpression>
```

JasperReports

```
<subreportExpression><![CDATA["/somePath/subreport.jasper"]]></subreportExpression>
```

@AndreasDietrich [JasperServer](#)

- <subreport>◦ ◦

Examples

;

◦ *jdbc* ◦

```
<subreport>
  <reportElement x="0" y="80" width="200" height="100"/>
  <subreportParameter name="someSubreportParameter">

<subreportParameterExpression><![CDATA[${someMasterReportParamter}]]></subreportParameterExpression>

  </subreportParameter>
  <subreportParameter name="anotherSubreportParameter">
```

```

        <subreportParameterExpression><![CDATA["Some text - constant
value"]]></subreportParameterExpression>
    </subreportParameter>
    <connectionExpression><![CDATA[${P{REPORT_CONNECTION}}]></connectionExpression>
    <returnValue subreportVariable="someVariableInSubreport"
toVariable="someVariableInMasterReport"/>
    <subreportExpression><![CDATA[${P{SUBREPORT_DIR} +
"subreport.jasper"}]></subreportExpression>
</subreport>

```

◦ [*net.sf.jasperreports.engine.data.JRBeanCollectionDataSource*](#)

```

<field name="someFieldWithList" class="java.util.List"/>
<!-- ..... -->
<subreport>
    <reportElement x="0" y="0" width="200" height="70"/>
    <parametersMapExpression><![CDATA[${P{REPORT_PARAMETERS_MAP}}]></parametersMapExpression>

    <dataSourceExpression><![CDATA[net.sf.jasperreports.engine.data.JRBeanCollectionDataSource(${F{someField}})]></dataSourceExpression>

    <subreportExpression><![CDATA[${P{SUBREPORT_DIR} +
"subreport.jasper"}]></subreportExpression>
</subreport>

```

<https://riptutorial.com/zh-TW/jasper-reports/topic/5452/>

3:

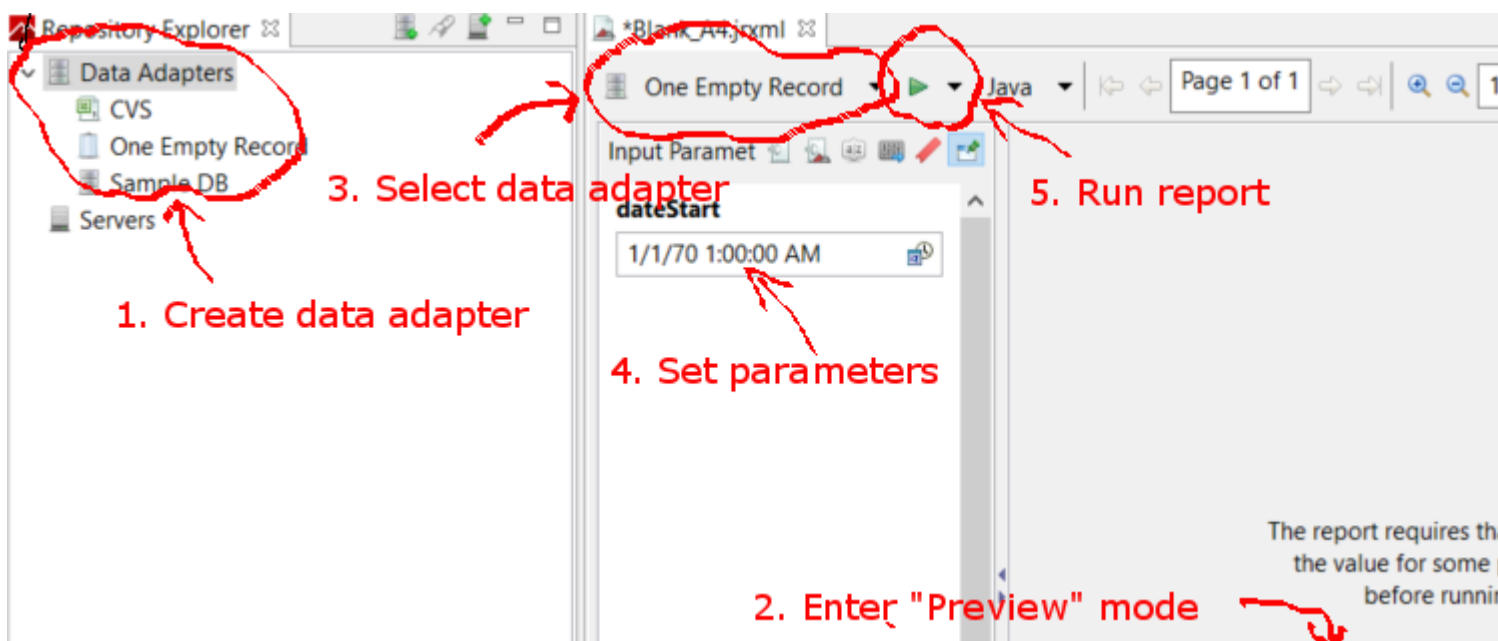
jasperPrint	
reportTemplate	.jasper
	Map \$P{key}
	net.sf.jasperreports.engine.JRDataSource
	java.sql.Connection

Examples

IDE

JasperSoft

1. ""
2. " "
3. dastasource"One Empty Record"
- 4.
5. ""



JavaJasperReport

◦

```
// Parameters passed into the report.
Map<String, Object> parameters = new HashMap<>();

// Arbitrary parameter passed into the report.
parameters.put("KEY", "Value");

// The compiled report design.
String path = "path/to/template.jasper";
```

.jrxml◦ .jrxml .jasper.jasper◦

```
// Establish a database connection.
Connection connection = DriverManager.getConnection(url, username, password);

// Fill the report, get the JasperPrint that can be exported to desired format.
JasperPrint jasperPrint = JasperFillManager.fillReport(
    path, parameters, connection);
```

```
// Populate this list of beans as per your requirements.
List<Bean> beans = new ArrayList<>();

// Wrap the beans in a beans in a JRBeanCollectionDataSource.
JRBeanCollectionDataSource datasource = new JRBeanCollectionDataSource(beans);

// Fill the report, get the JasperPrint that can be exported to desired format.
JasperPrint jasperPrint = JasperFillManager.fillReport(
    path, parameters, datasource);
```

```
// Fill the report, get the JasperPrint that can be exported to desired format.
JasperPrint jasperPrint = JasperFillManager.fillReport(path, parameters);
```

JasperReportwhenNoDataType="AllSectionsNoDetail"◦

<https://riptutorial.com/zh-TW/jasper-reports/topic/3958/>

4:

Examples

IDE。 [iReportJaspersoft Studio](#)。 。

textElementSansSerif

```
<textElement>
  <font fontName="DejaVu Sans"/>
</textElement>
```

JVM Java 。

JasperReport

。

JasperReports^{maven} distribution `jasperreports-fonts.jar`。 fontName

DejaVu Sans
DejaVu Serif
DejaVu Sans Mono

pdfitextfont

- PDF 。
- .tff [OpenType](#) UTF-8。
- iTextIdentity-H PDF。
- PDF[141](#)。

jasperiText [ligaturizer](#) ttf。 pdf

<https://riptutorial.com/zh-TW/jasper-reports/topic/5773/>

5: JasperReports .jrxml.jasper

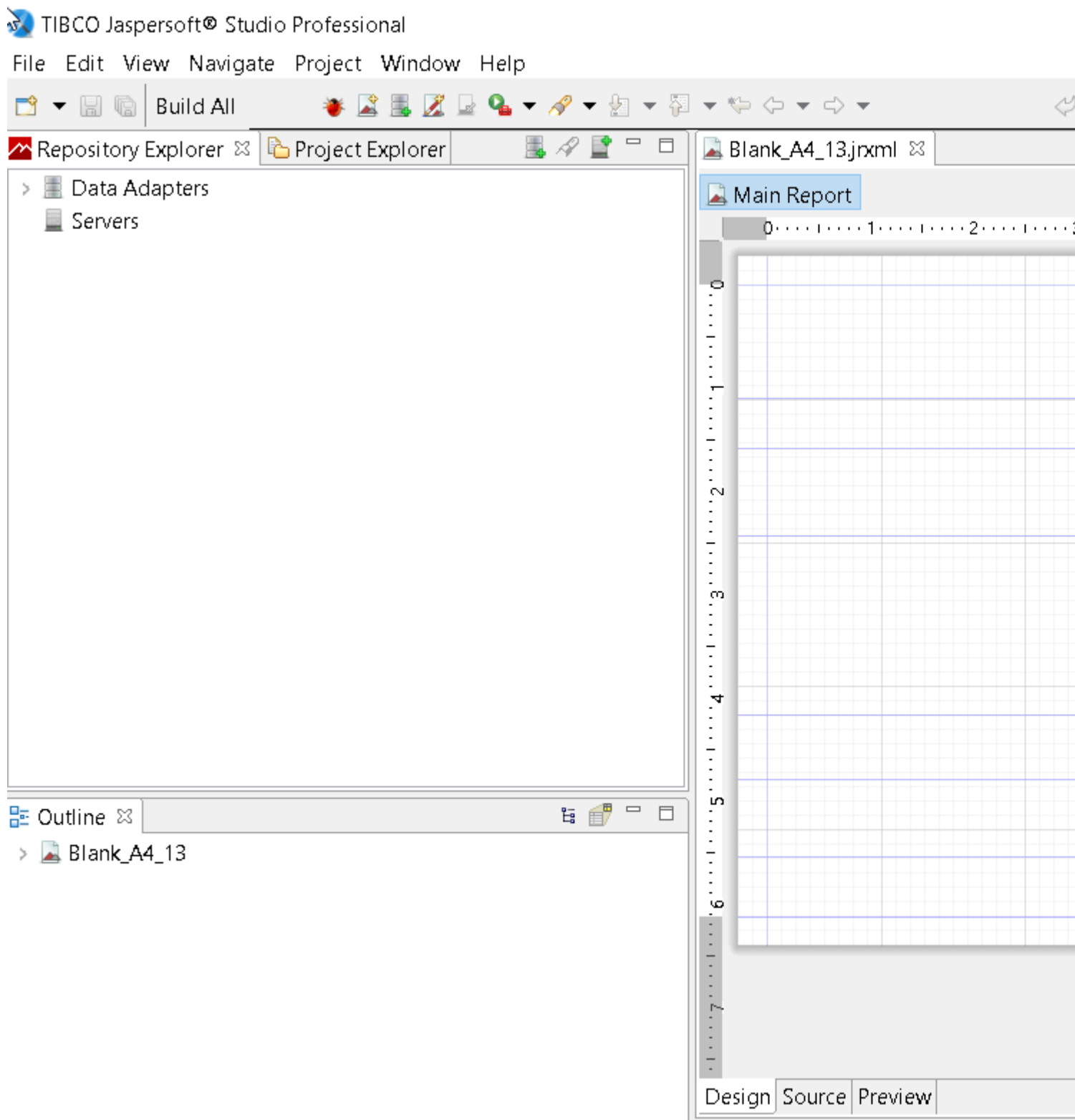
Examples

IDE

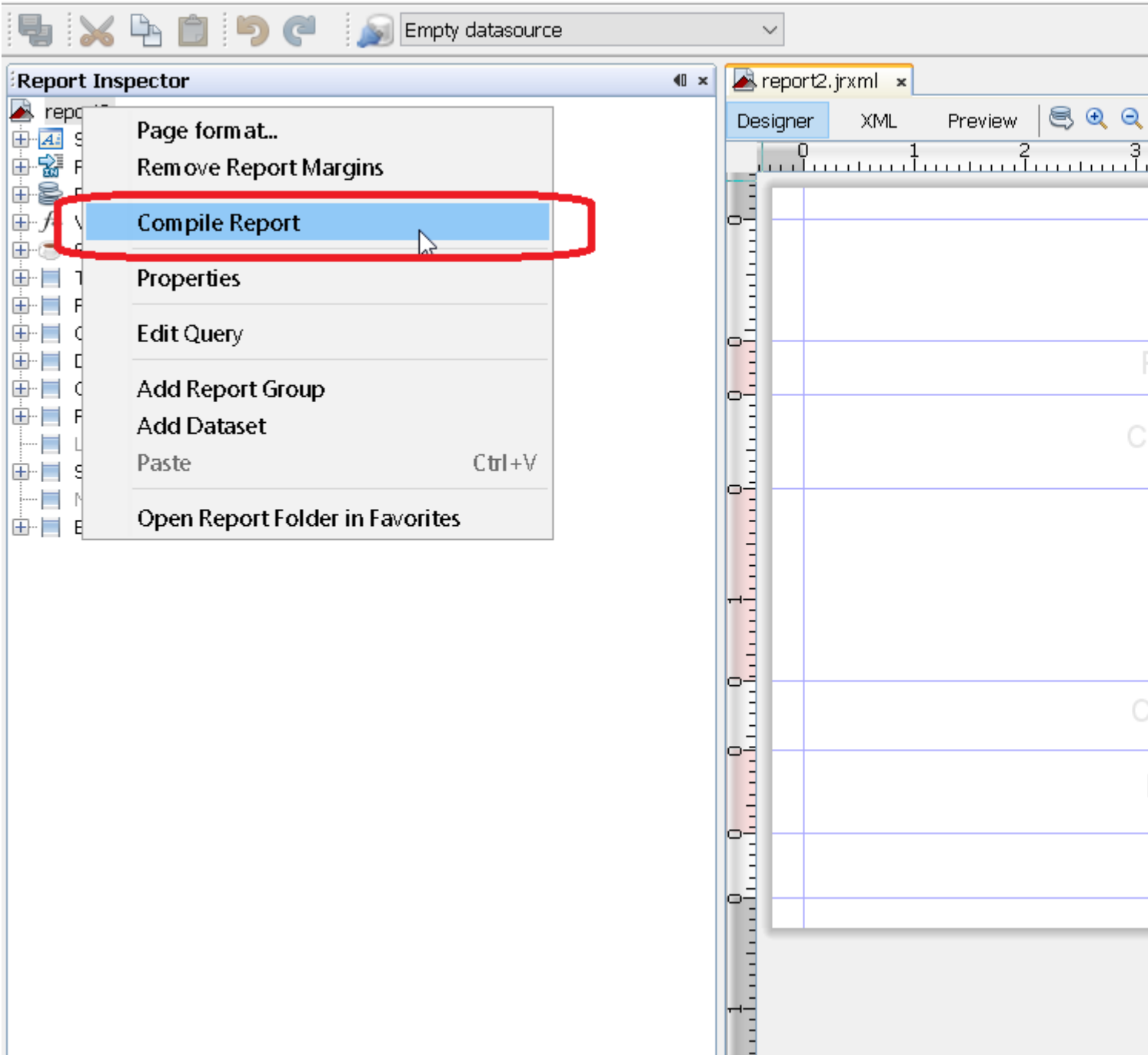
IDE [Jaspersoft Studio](#) [JSS iReport Designer](#) ◦

JasperReports.[.jrxml.jasper.jrxml](#) ◦

JSS""



iReport ""



Apache Ant

```
<target name="compile" description="Compiles report designs specified using the 'srcdir' in
the <jrc> tag." depends="prepare-compile-classpath">
  <mkdir dir="./build/reports"/>
  <taskdef name="jrc" classname="net.sf.jasperreports.ant.JRAntCompileTask">
    <classpath refid="project-classpath"/>
  </taskdef>
  <jrc
    srcdir="./reports"
    destdir="./build/reports"
    tempdir="./build/reports"
    keepjava="true"
```

```

        xmlvalidation="true">
        <classpath refid="sample-classpath"/>
        <include name="**/*.jrxml"/>
    </jrc>
</target>

```

Apache Ant

Java

.jrxml.jasperJava.jrxmlIDE◦ [JasperCompileManager](#).jrxml

```

JasperCompileManager.compileReportToFile(
    "designFile.jrxml", //Relative or absolute path to the .jrxml file to compile
    "compiled.jasper"); //Relative or absolute path to the compiled file .jasper

```

Apache Maven

Alex Nederlof[JasperReports-plugin](#)org.codehaus.mojojasperreports-maven-plugin◦

```

<build>
  <plugins>
    <plugin>
      <groupId>com.alexnaderlof</groupId>
      <artifactId>jasperreports-plugin</artifactId>
      <version>2.3</version>
      <executions>
        <execution>
          <phase>process-sources</phase>
          <goals>
            <goal>jasper</goal>
          </goals>
        </execution>
      </executions>
      <configuration>
        <sourceDirectory>src/main/resources/jrxml</sourceDirectory>
        <outputDirectory>${project.build.directory}/jasper</outputDirectory>
      </configuration>
    </plugin>
  </plugins>
</build>

```

Maven

mvn jasperreports:jasper

jasper \$ {project.build.directory} / jasper/ target / jasper

[JasperReports .jrxml.jasper](#) <https://riptutorial.com/zh-TW/jasper-reports/topic/4943/jasperreports--jrxml-jasper>

6: xls / xlsx

Examples

Java

xlsx

```
try (InputStream inputStream = JRLoader.getResourceInputStream(path)) { // read report as
input stream
    JasperReport jasperReport =
JasperCompileManager.compileReport(JRXmlLoader.load(inputStream)); // compile report

    Map<String, Object> params = new HashMap<>(); // init map with report's parameters
    params.put(JRParameter.REPORT_LOCALE, Locale.US);
    params.put(JRParameter.IS_IGNORE_PAGINATION, true);
    JasperPrint jasperPrint = JasperFillManager.fillReport(jasperReport, params, connection);
    // prepare report - passs parameters and jdbc connection

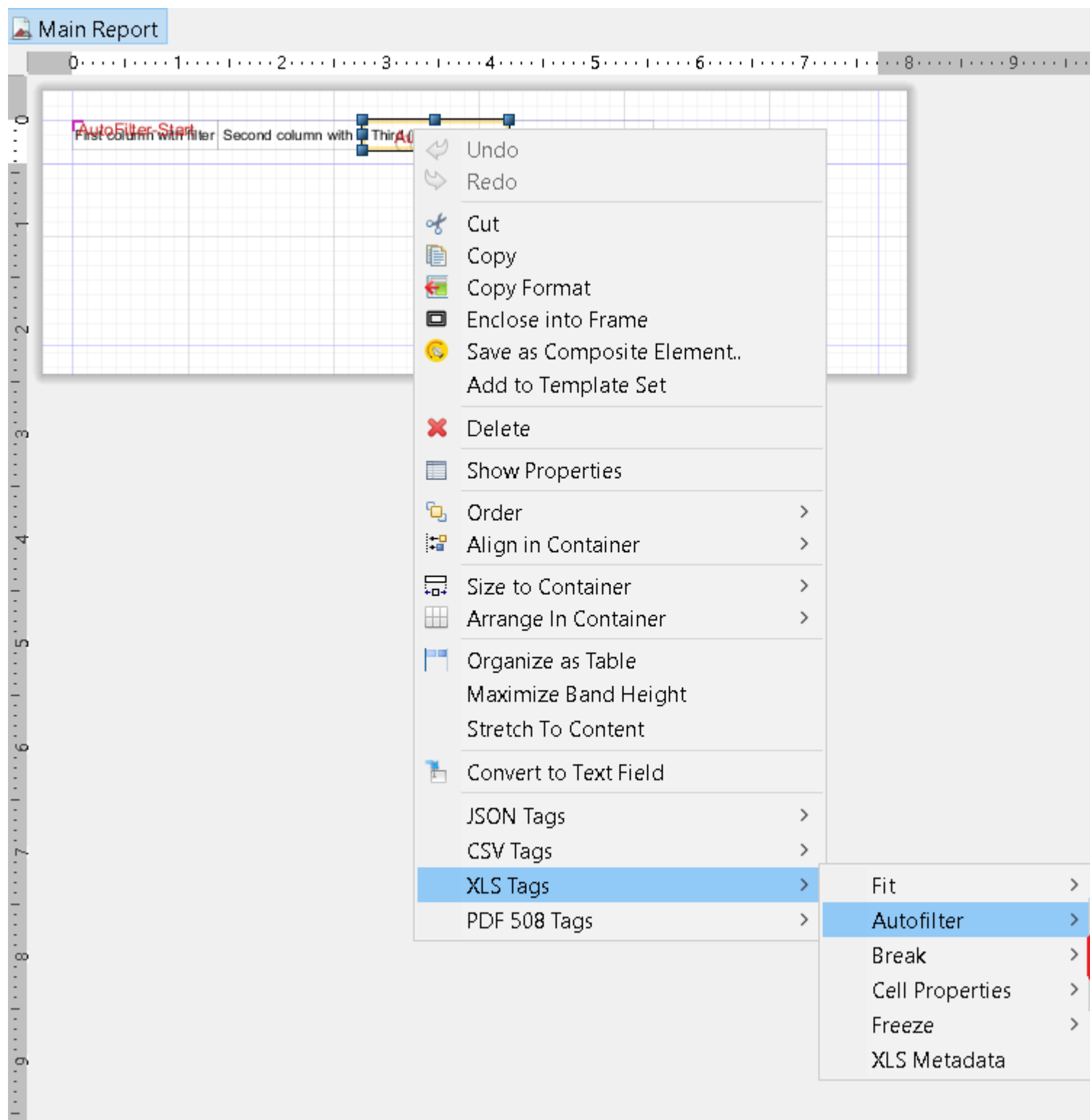
    JRXlsxExporter exporter = new JRXlsxExporter(); // initialize exporter
    exporter.setExporterInput(new SimpleExporterInput(jasperPrint)); // set compiled report as
input
    exporter.setExporterOutput(new SimpleOutputStreamExporterOutput(destFile)); // set output
file via path with filename
    SimpleXlsxReportConfiguration configuration = new SimpleXlsxReportConfiguration();
    configuration.setOnePagePerSheet(true); // setup configuration
    configuration.setDetectCellType(true);
    exporter.setConfiguration(configuration); // set configuration
    exporter.exportReport();
}
```

[net.sf.jasperreports.export.xls.auto.filter](#)^{xls}.

```
<columnHeader>
  <band height="30" splitType="Stretch">
    <staticText>
      <reportElement x="0" y="0" width="100" height="20">
        <property name="net.sf.jasperreports.export.xls.auto.filter" value="Start"/>
      </reportElement>
      <text><![CDATA[First column with filter]]></text>
    </staticText>
    <staticText>
      <reportElement x="100" y="0" width="100" height="20"/>
      <text><![CDATA[Second column with filter]]></text>
    </staticText>
    <staticText>
      <reportElement x="200" y="0" width="100" height="20">
        <property name="net.sf.jasperreports.export.xls.auto.filter" value="End"/>
      </reportElement>
      <text><![CDATA[Third (Last) column with filter]]></text>
    </staticText>
    <staticText>
      <reportElement x="300" y="0" width="100" height="20"/>
      <text><![CDATA[Fourth column without filter]]></text>
    </staticText>
```

```
</band>  
</columnHeader>
```

Jaspersoft Studiojrxml.



[xls / xlsx https://riptutorial.com/zh-TW/jasper-reports/topic/5008/xls---xlsx](https://riptutorial.com/zh-TW/jasper-reports/topic/5008/xls---xlsx)

7: pdf

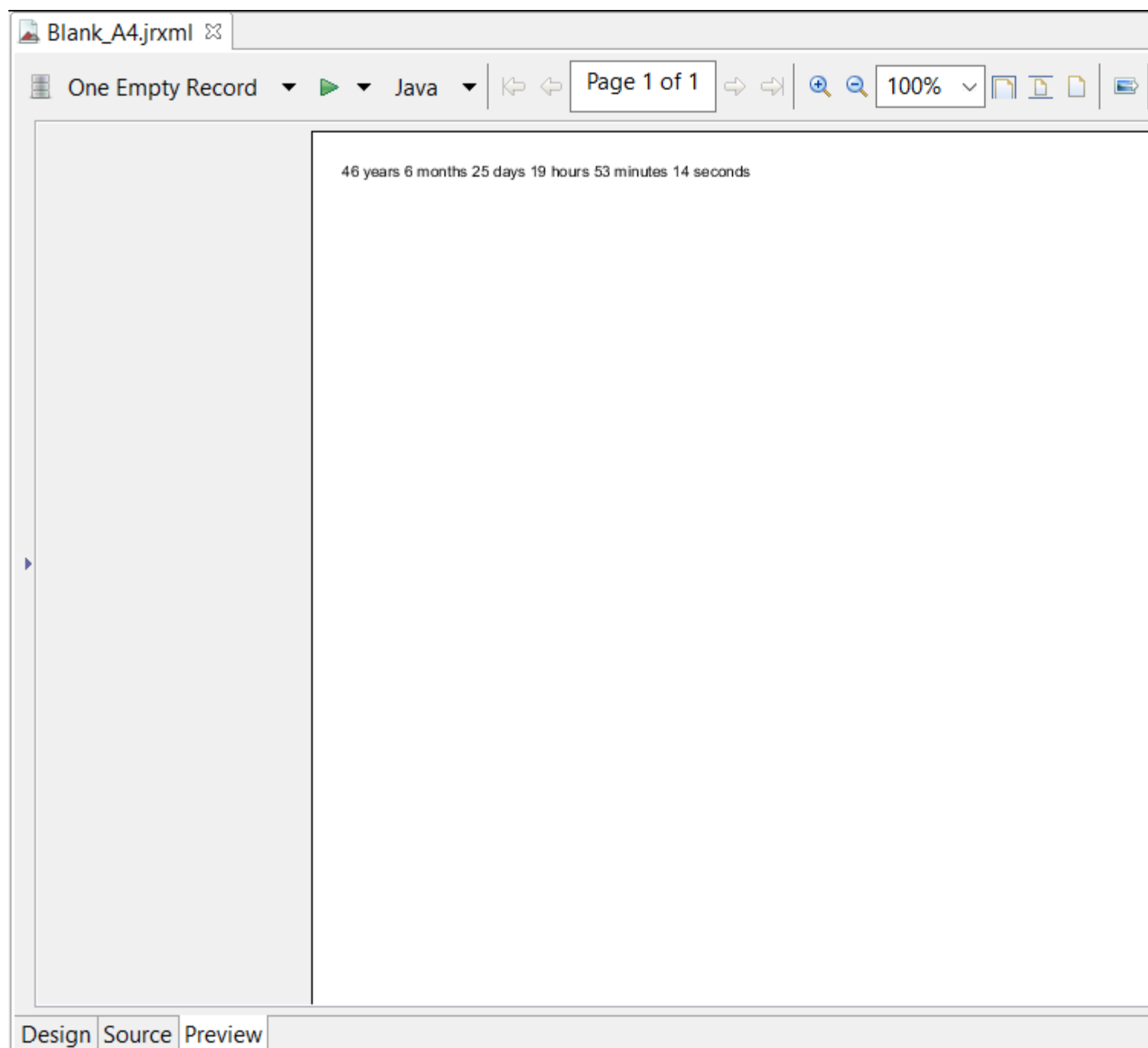
pdf

Examples

IDE

JasperSoft

“Pdf”



JasperPrintjrxml

```
// 1. Create exporter instance
JRPdfExporter exporter = new JRPdfExporter();

// 2. Set exporter input document
exporter.setExporterInput(new SimpleExporterInput(jasperPrint));

// 3. Set file path for exporter output
exporter.setExporterOutput(new SimpleOutputStreamExporterOutput("/path/filename.pdf"));

// 4. Create configuration instance
SimplePdfExporterConfiguration configuration = new SimplePdfExporterConfiguration();

// 5. Associate configuration with exporter
exporter.setConfiguration(configuration);

// 6. Fill export and write to file path
exporter.exportReport();
```

JasperPrintjrxml

```
List<JasperPrint> jasperPrintList = new ArrayList<>();
jasperPrintList.add(jasperPrint1);
jasperPrintList.add(jasperPrint2);

JRPdfExporter exporter = new JRPdfExporter();
exporter.setExporterInput(SimpleExporterInput.getInstance(jasperPrintList));
```

```
exporter.setExporterOutput(new SimpleOutputStreamExporterOutput("/path/filename.pdf"));
SimplePdfExporterConfiguration configuration = new SimplePdfExporterConfiguration();
exporter.setConfiguration(configuration);
exporter.exportReport();
```

[SimplePdfExporterConfiguration API](#) ◦

pdf <https://riptutorial.com/zh-TW/jasper-reports/topic/4190/pdf>

S. No		Contributors
1	jasper-reports	Alex K , Community , Dave Jarvis , Petter Friberg
2		Alex K
3		Alex K , Dave Jarvis , Petter Friberg
4		Dave Jarvis , Petter Friberg
5	JasperReports .jrxml .jasper	Alex K , Dave Jarvis , Petter Friberg
6	xls / xlsx	Alex K
7	pdf	Alex K , Dave Jarvis , Petter Friberg , RamenChef