

Experiment - 10

Lab Manual

Aim : "Delivering Reports including visualizations and customized properties."

Presentation of information in a required and user friendly manner is a must achievement for any software. Having data recorded into various data marts, data warehouses, etc is the first thing to have to be able to perform data analytics and second thing is preparing various reports out of information/knowledge extracted.

JasperSoft community edition provides an advanced way to achieve reporting which can also be embedded easily in java based applications. Here, the first example taken prepares the dataset and then using JasperSoft Studio achieves a feature-rich report.

Step 1: Start mongod

Know that JasperSoft Studio is going to communicate to mongoDB (in this exercise) for fetching data. 'mongod' utility has to be running all the time during the report development and preview.

Step 2: Load the data into mongoDB using mogoimport utility

```
mongoimport --db=mng --collection=student_master --type=csv
--headerline --file="Fundamentals of Business Analytics/Data Sheet in
Chapter 9/Source Data.csv"
```

```
> db;
mng
> db.student_master.find().count();
3796
> db.student_master.find().limit(1).pretty();
{
  "_id" : ObjectId("5cf6c2e623a0ea5de475874e"),
  "Candidate Id" : "C0002",
  "Employee Id" : "E0002",
  "First Name" : "Shailendra",
  "Middle Name" : "Kumar",
  "Last Name" : "",
  "DegreePercentage" : 72.55,
  "12th/DiplomaPercentage" : 75.77,
  "SSCPercentage" : 80.1,
  "University Name" : "RGPRV",
  "Native State" : "Madhya Pradesh",
  "Date of Birth" : "1980-01-01",
  "Gender" : "Male",
  "Marital Status" : "Single",
  "Religion" : "Hindu",
  "Blood Group" : "A+",
  "Hobbies" : "Reading, Music",
  "Languages" : "Hindi, English",
  "Address" : "123, Main Street, New Delhi",
  "Contact Number" : "9876543210",
  "Email" : "shailendra.kumar@example.com",
  "Created At" : ISODate("2023-01-01T10:00:00Z"),
  "Updated At" : ISODate("2023-01-01T10:00:00Z")
}
```

Step 3: Understanding Business Requirement

Provide multi-pages report of students having following features:

- Show First Name, Middle Name, Last Name, Background (CS/NCS) and Training Exam Average Percentage (Avg. of two exams).
- Records shall be grouped by the University Name.
- Records shall be sorted with the group based on Training Exam Average Percentage in descending order.
- User shall be able to run interactively for CS records or NCS records.
- Pagination shall be supported.

Coffee_Landscape.jrxml

MongoDB Connectio... Java Page 1 of 20 100%

Input Parameters

Background(CS\NCS)

CS

Reset

Student

University Group

The youth power.

First Name	Middle Name	Last Name	Background(CS\NCS)	TrainingExamAvgPerc
ADU				
Ruben	Gershone	Chaudhari	CS	88
ANU				
Pavithra		Gangadharan	CS	94
Ilampirai		Ganesan	CS	93
Premalatha	Soreka	Ganesh	CS	93
Anil		Dhawan	CS	91
Ranjit	Singh	Dhillon	CS	90
Sindhu		Gangadhar	CS	90
Ranjith		Dubey	CS	89
Dinesh	Babu	Ganesh	CS	89
Satheeshraj		Ganguly	CS	85
Suchithra		Gandhi	CS	84
Leno	Sharon	Fernandes	CS	81
Gokul		Eswaramorthy	CS	80
Bharathwaj	Radhakrishnan	Gangli	CS	79
Mayur		Dhiman	CS	67
AU				
Ashok	Reddy	Dhall	CS	85

Design Source Preview JasperReports Library

Step 4:

Start JasperSoft Studio IDE. Aim to achieve below like report. HighLevel To Do.

Coffee_Landscape.jrxml

Main Report

0 1 2 3 4 5 6 7 8 9 10 11

0 1 2 3

Student

University Group

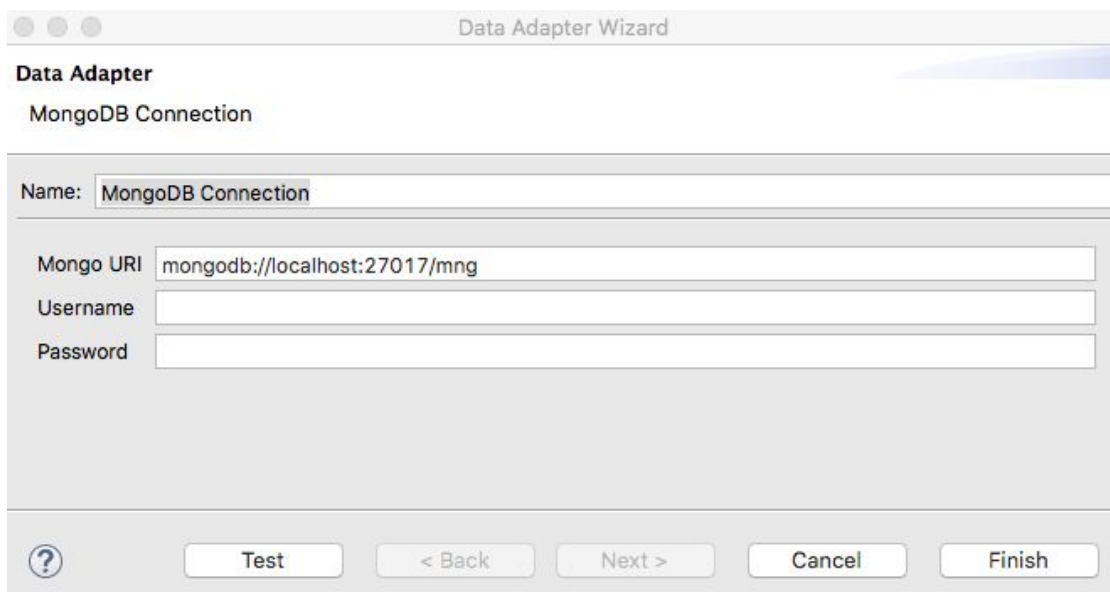
The youth power.

First Name	Middle Name	Last Name	Background(CS\NCS)	TrainingExamAvgPerc
\$F{University Name}				
\$F{First Name}	\$F{Middle Name}	\$F{Last Name}	\$F{Background	\$V
new java.util.Date()				"Page "+\$V" " + \$V

Page 3 of 7

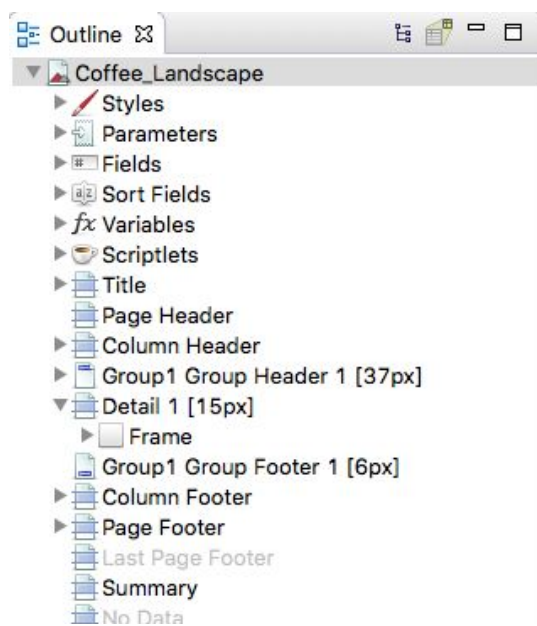
Step5: Understanding JasperSoft Report needed Technicality

1. JasperReports Project and Project Explorer
 1. *name* *MyReports*
2. JasperReports “New Report Wizard”
 1. *name* *Coffee_Landscape.jrxml*
3. Data Source
 1. Data Adapter – MongoDB Connection
 - name* *MongoDB Connection*
 - url* *mongodb://localhost:27017/mng*



The screenshot shows the 'Data Adapter Wizard' dialog box. The title bar says 'Data Adapter Wizard'. Below the title bar, it says 'Data Adapter' and 'MongoDB Connection'. The 'Name' field is set to 'MongoDB Connection'. The 'Mongo URI' field is set to 'mongodb://localhost:27017/mng'. The 'Username' and 'Password' fields are empty. At the bottom, there are buttons for '?', 'Test', '< Back', 'Next >', 'Cancel', and 'Finish'.

4. Repository Explorer
5. Outline



6. MongoDB Query

May create parameter first named “*Background(CS\NCS)*” or write the query without *\$P* line.

1. Dataset and Query Dialog

```
{
    'collectionName':'student_master',
    'findQuery': {
        {
            'TrainingExam1Percentage':{'$ne':''},
            'TrainingExam2Percentage':{'$ne':''},
            'Background(CS\NCS)':$P{Background(CS\NCS)}
        }
    }
}
```

7. Fields (Dataset)

Fields are mapped to dataset. i.e. field of collection from mongoDB.

```
<field name="University Name" class="java.lang.String"/>
<field name="First Name" class="java.lang.String"/>
  <field name="Middle Name" class="java.lang.String"/>
  <field name="Last Name" class="java.lang.String"/>
  <field name="Background(CS\NCS)" class="java.lang.String"/>
  <field name="TrainingExam1Percentage" class="java.lang.String"/>
  <field name="TrainingExam2Percentage" class="java.lang.String"/>
```

8. Group By

```
<groupExpression><![CDATA[$F{University Name}]]></groupExpression>
```

9. Sort Field

```
<sortField name="University Name"/>
      <sortField name="TrainingExamAvgPercentage" order="Descending"
type="Variable"/>
```

10. Properties Window

11. Variables – Expressions

Variables in JasperReport are extra memory created and used to do arithmetic on dataset.

i.e. here Avg Percentage is not the actual field from dataset but is calculated runtime based on the expression provided.

```
<variable name="TrainingExamAvgPercentage" class="java.lang.String">
<variableExpression><![CDATA[String.valueOf((Integer.parseInt($F{TrainingExam1Per
centage})+Integer.parseInt($F{TrainingExam2Percentage}))/2)]]></variableExpression>
</variable>
```

12. Report Parameters

Parameters in JasperReport are placeholders which can be mapped to query. Here the parameters are provided values at runtime if configured such a way. i.e. Report in here need user input 'CS' or 'NCS' to generate report of respective records only.

```
<parameter name="Background(CS\NCS)" class="java.lang.String">
<parameterDescription><![CDATA[CS Or NCS]]></parameterDescription>
</parameter>
```

13. Design-Source-Preview

The IDE provides a GUI drag and drop way to design the report. Internally which maps to xml-java source. Preview executes and displays embedded report within IDE itself. O be able to embed into external java application there are separate steps to look into.

14. Palette

15. Composite Elements

The page count, page numbers, etc are considered to be composite elements. They can be used as required in the reporting.

References :

<https://community.jaspersoft.com/wiki/designing-report-jaspersoft-studio>

<https://community.jaspersoft.com/wiki/jaspersoft-mongodb-query-language>

Exercises:

1. University Name on X-axis, Avg Percentage by various students on Y-Axis. Use bar chart.
2. Embed the JasperReport to your java application/web.
3. Populate dropdown for parameter.