

GROUPING, SUBTOTALS, AND TOTALS

Using Genero Report Writer GRS 3.00

© 2010 Four J's Development Tools



Objectives

After this instruction, you will be able to:

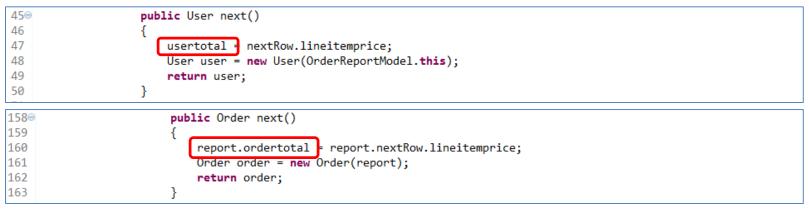
- Update your Java report application to include sub-totals and totals in the report data stream
- Add sub-totals to a report
- Add totals to a report



- Example: 'OrderReportModel.java' in Demo
- Steps
 - Define appropriate variables



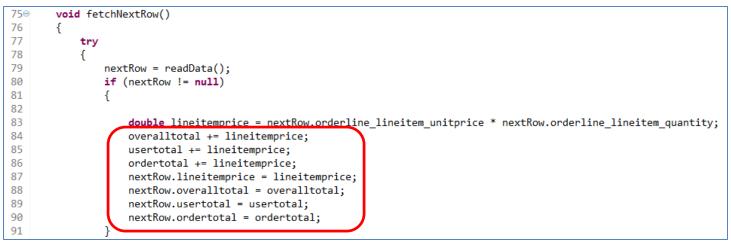
Reinitialize totals on appropriate iterator





Modify the Java report application

- Calculate totals on each row of data



Specify variables in Row class, with specific XML annotations (for relevant info in XSD schema)

| 298⊝ | static class Row |
|------|---|
| 299 | { |
| 3000 | <pre>@XmlElement(name = "orderline.orders.orderid", required = true, nillable = true)</pre> |
| 301 | <pre>public int orderline_orders_orderid;</pre> |
| | |
| 440⊝ | <pre>@XmlElement(name = "overalltotal", required = true, nillable = true)</pre> |
| 441 | public Double overalltotal; |
| 4429 | <pre>@XmlElement(name = "usertotal", required = true, nillable = true)</pre> |
| 443 | public Double usertotal; |
| | |
| 444⊝ | <pre>@XmlElement(name = "ordertotal", required = true, nillable = true)</pre> |
| 445 | public Double ordertotal; |
| 446 | } |



- Prefer calculating values in the Java report program over calculating values in the Report Design Document
 - Use the power of the Java language
- Create and name variables to store the results.
 - Use XML annotations
 - Naming the variable makes it easier to access in the Report Design document

```
@XmlElement(name = "overalltotal", required = true, nillable = true)
public Double overalltotal;
@XmlElement(name = "usertotal", required = true, nillable = true)
public Double usertotal;
@XmlElement(name = "ordertotal", required = true, nillable = true)
public Double ordertotal;
```



Totals problem characterized by: Should the totals print

- floating at the end of the detail?
- or in a fixed position on each page?



Totals positioning solutions include:

- Calculate totals in Java source for each row of data
- Positioning totals: either
 - Position totals after the detail by leaving Y property blank
 - Position totals in fixed position by setting Y property (normally relative to bottom of page)

– e.g. Y=max-30mm



An alternative totals positioning solution

- Calculate totals in Java source for each row of data
- Total value can be printed in PAGE HEADER and/or PAGE FOOTER (covered in later module)







- Exercise 1
 - Open the original 'Sales' project
 - Check how the total is calculated in 'Sales.java' source code
 - Add the total price at the bottom of the list; position it so that it gets properly aligned with the price column
- Exercise 2
 - Open the 'OrderReportJava' demo project
 - Check how totals are calculated in the 'OrderReportModel.java' source code