Generazione File Excel

Per iniziare la generazione di un excel bisogna prima di tutto aggiungere nel pom questa dipendenza:

<dependency>

<groupId>com.github.bld-commons.excel</groupId>

<artifactId>generator-excel</artifactId>

<version>1.0.3</version>

</dependency>

e nella classe del main dobbiamo aggiungere la seguente annotazione:

* @ComponentScan(basePackages= "your.project.package","bld.generator.report" })

Per la creazione di un excel ci sono 3 oggetti i quali rapprensentano:

1. Excel (contenitore di Sheet)
2. Foglio (conteniotre di Righe)
3. Riga

Per poter ottenere questo risultato, in generator-excel ci sono 3 oggetti base i quali rappresentano le 3 situazioni indicate sopra:

* **class**
  + ReportExcel (Excel)
* **abstract** **class**
  + BaseSheet (Foglio)
    - SheetSummary
    - SheetData
    - MergeSheet
  + RowSheet (Riga)

I 3 oggetti SheetSummary, SheetData e MergeSheet estendono BaseSheet, essi servono per definire una tipologia di foglio es:

* SheetSummary:

|  |  |
| --- | --- |
| Autore | |
| Nome | Mario |
| Cognome | Rossi |
| Data di nascita | 23-04-1960 |
| Città | Roma |

* SheetData:

|  |  |  |  |
| --- | --- | --- | --- |
| NOME | COGNOME | DATA DI NASCITA | CITTÀ |
| Mario | Rossi | 23-04-1960 | Roma |
| Luca | Bianchi | 20-01-1970 | Milano |
| Martina | Verdi | 18-04-1989 | Firenze |

* MergeData sarà una concatenazione delle diverse tabelle

Il layout delle tabelle e delle celle per pagina sono configurabili attraverso alcune annotazioni sotto il pacchetto:

* **package** bld.generator.report.excel.annotation;

Le annotazioni fondamentali per poter disegnare l’excel sono:

* Da BaseSheet:
  + @ExcelSheetLayout
  + @ExcelMarginSheet
  + @ExcelMarginSheet
* Da RowSheet:
  + @ExcelColumn
  + @ExcelCellLayout

Sull’annotazione @ExcelSheetLayout se aggiugessimo l’opzione “notMerge=false” si otterrebbe un report con le righe unite se il campo rimanesse uguale al record precedente, per gestire il merge delle righe abbiamo bisogno di un’altra annotazione all’interno della classe di tipo RowSheet che è:

* @ExcelMergeRow(referenceField = "matricola")

Un esempio di classe che genera un report con righe mergiate è:

**package** bld.generator.report.junit.entity;

**import** java.util.Calendar;

**import** org.apache.poi.ss.usermodel.HorizontalAlignment;

**import** bld.generator.report.excel.DynamicRowSheet;

**import** bld.generator.report.excel.annotation.ExcelCellLayout;

**import** bld.generator.report.excel.annotation.ExcelColumn;

**import** bld.generator.report.excel.annotation.ExcelDate;

**import** bld.generator.report.excel.annotation.ExcelFunction;

**import** bld.generator.report.excel.annotation.ExcelFunctionMergeRow;

**import** bld.generator.report.excel.annotation.ExcelFunctionRow;

**import** bld.generator.report.excel.annotation.ExcelFunctionRows;

**import** bld.generator.report.excel.annotation.ExcelMergeRow;

@ExcelFunctionRows(excelFunctions = {@ExcelFunctionRow(excelCellsLayout = @ExcelCellLayout(horizontalAlignment =HorizontalAlignment.***RIGHT***, precision = 2),

excelColumn = @ExcelColumn(indexColumn = 9, nameColumn = "Prezzo Totale"),

excelFunction = @ExcelFunction(function = "sum(${prezzo},${supplemento})", nameFunction = "prezzoTotale"))

},

excelFunctionMerges = {

@ExcelFunctionMergeRow(excelCellsLayout = @ExcelCellLayout(horizontalAlignment = HorizontalAlignment.***RIGHT***, precision = 2),

excelColumn = @ExcelColumn(indexColumn = 7.1, nameColumn = "Prezzo Totale per Autore"),

excelMergeRow = @ExcelMergeRow(referenceField = "matricola"), excelFunction = @ExcelFunction(function = "sum(${prezzoFrom}:${prezzoTo})", nameFunction = "prezzoTotalePerAutore"))

}

)

**public** **class** AutoreLibriRowDynamic **extends** DynamicRowSheet {

@ExcelColumn(nameColumn = "Nome", indexColumn = 2)

@ExcelCellLayout

@ExcelMergeRow(referenceField = "matricola")

**private** String nome;

@ExcelColumn(nameColumn = "Cognome", indexColumn = 3)

@ExcelCellLayout

@ExcelMergeRow(referenceField = "matricola")

**private** String cognome;

@ExcelColumn(nameColumn = "Data di Nascita", indexColumn = 4)

@ExcelDate

@ExcelCellLayout(horizontalAlignment = HorizontalAlignment.***CENTER***)

@ExcelMergeRow(referenceField = "matricola")

**private** Calendar dataDiNascita;

@ExcelColumn(nameColumn = "Titolo", indexColumn = 6)

@ExcelCellLayout

**private** String titolo;

@ExcelColumn(nameColumn = "Genere", indexColumn = 5)

@ExcelCellLayout

@ExcelMergeRow(referenceField = "cognome")

**private** String genere;

@ExcelColumn(nameColumn = "Matricola", indexColumn = 1)

@ExcelCellLayout(horizontalAlignment = HorizontalAlignment.***RIGHT***)

@ExcelMergeRow(referenceField = "")

**private** Integer matricola;

@ExcelColumn(nameColumn = "Prezzo", indexColumn = 7)

@ExcelCellLayout(horizontalAlignment = HorizontalAlignment.***RIGHT***, precision = 2)

**private** Double prezzo;

@ExcelColumn(nameColumn = "Supplemento", indexColumn = 8)

@ExcelCellLayout(horizontalAlignment = HorizontalAlignment.***RIGHT***, precision = 2)

**private** Double supplemento;

...get and set

}

Il risultato è:

* [TestExcel](https://github.com/bld-commons/dev-excel/blob/master/Mondadori Dynamic.xlsx?raw=true)