



链滴

MyBatis MySQL Oracle 分页插件

作者: [88250](#)

原文链接: <https://ld246.com/article/1366178431625>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

<p>MyBatis 的 MySQL、Oracle 分页插件，使用相同的分页接口。</p>

<p> </p>

```
<pre>/**
```

```
* 分页对象.
```

```
*
```

```
* @author <a href=&quot;mailto:DL88250@gmail.com&quot;&gt;Liang Ding</a>
```

```
* @version 1.0.1.0, Oct 6, 2012
```

```
*/
```

```
public final class Page implements Serializable {
```

```
/**
```

```
* 默认的序列化版本 id.
```

```
*/
```

```
private static final long serialVersionUID = 1L;
```

```
/**
```

```
* 分页查询开始记录位置.
```

```
*/
```

```
private int begin;
```

```
/**
```

```
* 分页查看下结束位置.
```

```
*/
```

```
private int end;
```

```
/**
```

```
* 每页显示记录数.
```

```
*/
```

```
private int length = 20;
```

```
/**
```

```
* 查询结果总记录数.
```

```
*/
```

```
private int totalRecords;
```

```
/**
```

```
* 当前页码.
```

```
*/
```

```
private int pageNo;
```

```
/**
```

```
* 总共页数.
```

```
*/
```

```
private int pageCount;
```

```
public Page() {
```

```
}
```

```
/**
```

```
* 构造函数.
```

```
*
```

```
* @param begin
```

```
* @param length
```

```
*/
```

```
public Page(int begin, int length) {
```

```
    this.begin = begin;
```

```
    this.length = length;
```

```
    this.end = this.begin + this.length;
```

```

    this.pageNo = (int) Math.floor((this.begin * 1.0d) / this.length) + 1;
}

/**
 * @param begin
 * @param length
 * @param count
 */
public Page(int begin, int length, int totalRecords) {
    this(begin, length);
    this.totalRecords = totalRecords;
}

/**
 * 设置页数, 自动计算数据范围.
 *
 * @param pageNo
 */
public Page(int pageNo) {
    this.pageNo = pageNo;
    pageNo = pageNo > 0 ? pageNo : 1;
    this.begin = this.length * (pageNo - 1);
    this.end = this.length * pageNo;
}

/**
 * @return the begin
 */
public int getBegin() {
    return begin;
}

/**
 * @return the end
 */
public int getEnd() {
    return end;
}

/**
 * @param end
 * the end to set
 */
public void setEnd(int end) {
    this.end = end;
}

/**
 * @param begin
 * the begin to set
 */
public void setBegin(int begin) {
    this.begin = begin;
    if (this.length != 0) {

```

```

        this.pageNo = (int) Math.floor((this.begin * 1.0d) / this.length) + 1;
    }
}

/**
 * @return the length
 */
public int getLength() {
    return length;
}

/**
 * @param length
 *         the length to set
 */
public void setLength(int length) {
    this.length = length;
    if (this.begin != 0) {
        this.pageNo = (int) Math.floor((this.begin * 1.0d) / this.length) + 1;
    }
}

/**
 * @return the totalRecords
 */
public int getTotalRecords() {
    return totalRecords;
}

/**
 * @param totalRecords
 *         the totalRecords to set
 */
public void setTotalRecords(int totalRecords) {
    this.totalRecords = totalRecords;
    this.pageCount = (int) Math.floor((this.totalRecords * 1.0d) / this.length);
    if (this.totalRecords % this.length != 0) {
        this.pageCount++;
    }
}

/**
 * @return the pageNo
 */
public int getPageNo() {
    return pageNo;
}

/**
 * @param pageNo
 *         the pageNo to set
 */
public void setPageNo(int pageNo) {
    this.pageNo = pageNo;
}

```

```

        pageNo = pageNo > 0 ? pageNo : 1;
        this.begin = this.length * (pageNo - 1);
        this.end = this.length * pageNo;
    }

    /**
     * @return the pageCount
     */
    public int getPageCount() {
        if (pageCount == 0) {
            return 1;
        }
        return pageCount;
    }

    /**
     * @param pageCount
     *      the pageCount to set
     */
    public void setPageCount(int pageCount) {
        this.pageCount = pageCount;
    }

    @Override
    public String toString() {
        final StringBuilder builder = new StringBuilder("&begin=");
        builder.append(begin).append("&end=");
        builder.append(end).append("&length=");
        builder.append(length).append("&totalRecords=");
        builder.append(totalRecords).append("&pageNo=");
        builder.append(pageNo).append("&pageCount=");
        builder.append(pageCount);

        return builder.toString();
    }
}
</pre>

```



```
<pre>/**
```

```
* Oracle 分页生成插件。
```

```
*
```

```
* @author <a href="mailto:DL88250@gmail.com">Liang Ding</a>
```

```
* @version 1.0.0.0, May 31, 2012
```

```
*/
```

```
public class OraclePaginationPlugin extends PluginAdapter {
```

```
@Override
```

```
public boolean modelExampleClassGenerated(TopLevelClass topLevelClass,
    IntrospectedTable introspectedTable) {
```

```
    // add field, getter, setter for limit clause
```

```
    addPage(topLevelClass, introspectedTable, "page");
```

```
    return super.modelExampleClassGenerated(topLevelClass, introspectedTable);
```

```

}

@Override
public boolean sqlMapDocumentGenerated(Document document, introspectedTable introspectedTable) {
    XmlElement parentElement = document.getRootElement();

    // 产生分页语句前半部分
    XmlElement paginationPrefixElement = new XmlElement("&quot;sql&quot;");
    paginationPrefixElement.addAttribute(new Attribute("&quot;id&quot;", "&quot;OracleDialectPrefix&quot;"));
    XmlElement pageStart = new XmlElement("&quot;if&quot;");
    pageStart.addAttribute(new Attribute("&quot;test&quot;", "&quot;page != null&quot;"));
    pageStart.addElement(new TextElement(
        "&quot;select * from ( select row_*, rownum rownum_ from ( &quot;));"));
    paginationPrefixElement.addElement(pageStart);
    parentElement.addElement(paginationPrefixElement);

    // 产生分页语句后半部分
    XmlElement paginationSuffixElement = new XmlElement("&quot;sql&quot;");
    paginationSuffixElement.addAttribute(new Attribute("&quot;id&quot;", "&quot;OracleDialectSuffix&quot;"));
    XmlElement pageEnd = new XmlElement("&quot;if&quot;");
    pageEnd.addAttribute(new Attribute("&quot;test&quot;", "&quot;page != null&quot;"));
    pageEnd
        .addElement(new TextElement(
            "&quot;&lt;!&lt;CDATA[ ] row_ where rownum &lt;= #{page.end} ) where rownum_ > #{page.begin} ]&gt;&quot;"));
    paginationSuffixElement.addElement(pageEnd);
    parentElement.addElement(paginationSuffixElement);

    return super.sqlMapDocumentGenerated(document, introspectedTable);
}

@Override
public boolean sqlMapSelectByExampleWithoutBLOBsElementGenerated(XmlElement element, introspectedTable introspectedTable) {
    XmlElement pageStart = new XmlElement("&quot;include&quot;"); //NON-NLS-1$
    pageStart.addAttribute(new Attribute("&quot;refid&quot;", "&quot;OracleDialectPrefix&quot;"));
    ;
    element.getElements().add(0, pageStart);

    XmlElement isNotNullElement = new XmlElement("&quot;include&quot;"); //NON-NLS-1$
    isNotNullElement.addAttribute(new Attribute("&quot;refid&quot;", "&quot;OracleDialectSuffix&quot;"));
    element.getElements().add(isNotNullElement);

    return super.sqlMapUpdateByExampleWithoutBLOBsElementGenerated(element, introspectedTable);
}

```

```

/**
 * @param topLevelClass
 * @param introspectedTable
 * @param name
 */
private void addPage(TopLevelClass topLevelClass, IntrospectedTable introspectedTable,
    String name) {
    topLevelClass.addImportedType(new FullyQualifiedJavaType(
        "com.yuanxin.framework.mybatis.Page"));
    CommentGenerator commentGenerator = context.getCommentGenerator();
    Field field = new Field();
    field.setVisibility(JavaVisibility.PROTECTED);
    field.setType(new FullyQualifiedJavaType("com.yuanxin.framework.mybatis.Page"));
    field.setName(name);
    commentGenerator.addFieldComment(field, introspectedTable);
    topLevelClass.addField(field);
    char c = name.charAt(0);
    String camel = Character.toUpperCase(c) + name.substring(1);
    Method method = new Method();
    method.setVisibility(JavaVisibility.PUBLIC);
    method.setName("set" + camel);
    method.addParameter(new Parameter(new FullyQualifiedJavaType(
        "com.yuanxin.framework.mybatis.Page"), name));
    method.addBodyLine("this." + name + "=" + name + ";");
    commentGenerator.addGeneralMethodComment(method, introspectedTable);
    topLevelClass.addMethod(method);
    method = new Method();
    method.setVisibility(JavaVisibility.PUBLIC);
    method.setReturnType(new FullyQualifiedJavaType("com.yuanxin.framework.mybatis
Page"));
    method.setName("get" + camel);
    method.addBodyLine("return " + name + ";");
    commentGenerator.addGeneralMethodComment(method, introspectedTable);
    topLevelClass.addMethod(method);
}

/**
 * This plugin is always valid - no properties are required
 */
@Override
public boolean validate(List<String> warnings) {
    return true;
}
}

```



```
<pre>/**
```

```
* MySQL 分页生成插件。
```

```
*
```

```
* @author <a href="mailto:DL88250@gmail.com">Liang Ding</a>
```

```
* @version 1.0.0.1, Oct 10, 2012
```

```

*/
public final class MySQLPaginationPlugin extends PluginAdapter {

@Override
public boolean modelExampleClassGenerated(TopLevelClass topLevelClass,
    IntrospectedTable introspectedTable) {
    // add field, getter, setter for limit clause
    addPage(topLevelClass, introspectedTable, &quot;page&quot;);
    return super.modelExampleClassGenerated(topLevelClass, introspectedTable);
}

@Override
public boolean sqlMapSelectByExampleWithoutBLOBsElementGenerated(XmlElement element

    IntrospectedTable introspectedTable) {
    XmlElement page = new XmlElement(&quot;if&quot;);
    page.addAttribute(new Attribute(&quot;test&quot;, &quot;page != null&quot;));
    page.addElement(new TextElement(&quot;limit #{page.begin} , #{page.length}&quot;));
    element.addElement(page);

    return super.sqlMapUpdateByExampleWithoutBLOBsElementGenerated(element, introspec
edTable);
}

/**
 * @param topLevelClass
 * @param introspectedTable
 * @param name
 */
private void addPage(TopLevelClass topLevelClass, IntrospectedTable introspectedTable,
    String name) {
    topLevelClass.addImportedType(new FullyQualifiedJavaType(
        &quot;com.yuanxin.framework.mybatis.Page&quot;));
    CommentGenerator commentGenerator = context.getCommentGenerator();
    Field field = new Field();
    field.setVisibility(JavaVisibility.PROTECTED);
    field.setType(new FullyQualifiedJavaType(&quot;com.yuanxin.framework.mybatis.Page&qu
t));
    field.setName(name);
    commentGenerator.addFieldComment(field, introspectedTable);
    topLevelClass.addField(field);
    char c = name.charAt(0);
    String camel = Character.toUpperCase(c) + name.substring(1);
    Method method = new Method();
    method.setVisibility(JavaVisibility.PUBLIC);
    method.setName(&quot;set&quot; + camel);
    method.addParameter(new Parameter(new FullyQualifiedJavaType(
        &quot;com.yuanxin.framework.mybatis.Page&quot;), name));
    method.addBodyLine(&quot;this.&quot; + name + &quot;=&quot; + name + &quot;;&quot;
);
    commentGenerator.addGeneralMethodComment(method, introspectedTable);
    topLevelClass.addMethod(method);
    method = new Method();

```



```
method.setVisibility(JavaVisibility.PUBLIC);
method.setReturnType(new FullyQualifiedJavaType("<code>com.yuanxin.framework.mybatis
Page</code>"));
method.setName("<code>get</code> + camel");
method.addBodyLine("<code>return </code> + name + </code>");
commentGenerator.addGeneralMethodComment(method, introspectedTable);
topLevelClass.addMethod(method);
}

/**
 * This plugin is always valid - no properties are required
 */
public boolean validate(List<String> warnings) {
    return true;
}
}</pre>
```

<p> </p>

<p>使用时在 generatorConfig.xml 中配置对应的插件即可，最终，在生成的 Criteria 中就会存在 P
ge 字段，用于设置分页。</p>

<p> </p>