

- 1.
- 2.
- 3.
- 4.
- 5./ API
- 6.
- 7. Just code it!
 - a.
 - b.
 - c.
 - d. (i18n)
 - e. Felix
 - f.
- 8.

File Link Datalist [Bean Shell](#)

1.

2.

[Datalist Column Formatter](#)

3.

Joget Workflow

```
/jw/web/client/app/{appId}/{appVersion}/form/download/{formId}/{recordId}/{fileName}.?attachment=true
```

:

```
http://localhost:8080/jw/web/client/app/test/1/form/download/testFile/f6946830-c0a80070-48cad9b7-8b971682/CHANGES.txt.?attachment=true
```

1. ID"
2. " ?attachment = true"

4.

Datalist

5./ API

6.

Joget Workflow Source Code

Macbook ProJoget5.0.0

```
- Home
  - joget
    - plugins
      - jw-community
        -5.0.0
```

"plugins" "jw-community" Joget Workflow

"plugins" maven

```
cd joget/plugins/
~/joget/jw-community/5.0.0/wflow-plugin-archetype/create-plugin.sh org.joget.tutorial
file_link_datalist_formatter 5.0.0
```

shellmaven

```
Define value for property 'version': 1.0-SNAPSHOT: : 5.0.0
[INFO] Using property: package = org.joget.tutorial
Confirm properties configuration:
groupId: org.joget.tutorial
artifactId: file_link_datalist_formatter
version: 5.0.0
package: org.joget.tutorial
Y: : y
```

"BUILD SUCCESS" "plugins" "file_link_datalist_formatter"

IDE maven [NetBeans](#)

7. Just code it!

a.

"org.joget.tutorial" "FileLinkDatalistFormatter" **org.joget.apps.datalist.model.DataListColumnFormatDefault** [Datalist Column Formatter](#)

b.

AppPluginUtil.getMessage(18nMESSAGE_PATH

Implementation of all basic abstract methods

```
package org.joget.tutorial;
import org.joget.apps.app.service.AppPluginUtil;
import org.joget.apps.app.service.AppUtil;
import org.joget.apps.datalist.model.DataList;
import org.joget.apps.datalist.model.DataListColumn;
import org.joget.apps.datalist.model.DataListColumnFormatDefault;

public class FileLinkDatalistFormatter extends DataListColumnFormatDefault {

    private final static String MESSAGE_PATH = "messages/FileLinkDatalistFormatter";

    public String getName() {
        return "File Link Datalist Formatter";
    }

    public String getVersion() {
        return "5.0.0";
    }

    public String getClassName() {
        return getClass().getName();
    }

    public String getLabel() {
        //support i18n
        return AppPluginUtil.getMessage("org.joget.tutorial.FileLinkDatalistFormatter.pluginLabel",
getClassName(), MESSAGE_PATH);
    }

    public String getDescription() {
        //support i18n
        return AppPluginUtil.getMessage("org.joget.tutorial.FileLinkDatalistFormatter.pluginDesc",
getClassName(), MESSAGE_PATH);
    }

    public String getPropertyOptions() {
        return AppUtil.readPluginResource(getClassName(), "/properties/fileLinkDatalistFormatter.json", null,
true, MESSAGE_PATH);
    }

    public String format(DataList dataList, DataListColumn column, Object row, Object value) {
        throw new UnsupportedOperationException("Not supported yet.");
    }
}
```

UlgetPropertyOptions "/properties/fileLinkDatalistFormatter.json""file_link_datalist_formatter / src / main""resources / properties""properties""fileLinkDatalistFormatter.json"

"" message.key ""i18n

```
[{
  title : '@@datalist.fileLinkFormatter.config@@',
  properties : [{
    name : 'formDefId',
    label : '@@datalist.fileLinkFormatter.form@@',
    type : 'selectbox',
    options_ajax : '[CONTEXT_PATH]/web/json/console/app[APP_PATH]/forms/options',
    required : 'True'
  },
  {
    name : 'attachment',
    label : '@@datalist.fileLinkFormatter.attachment@@',
    type : 'checkbox',
    options : [{
      value : 'true',
      label : ''
    }]
  }
]}]
```

```

public String format(DataList dataList, DataListColumn column, Object row, Object value) {
    String result = (String) value;

    if (result != null && !result.isEmpty()) {
        try {
            String formDefId = getPropertyString("formDefId");
            AppDefinition appDef = AppUtil.getCurrentAppDefinition();
            result = "";

            String attachment = "";
            if ("true".equals(getPropertyString("attachment"))) {
                attachment = "?attachment=true";
            }

            //get the id of this record
            String primaryKeyValue = (String) LookupUtil.getBeanProperty(row, dataList.getBinder().
getPrimaryKeyColumnName());

            HttpServletRequest request = WorkflowUtil.getHttpServletRequest();

            //suport for multi values
            for (String v : value.toString().split(";")) {
                if (!v.isEmpty()) {
                    // determine actual path for the file uploads
                    String fileName = v;
                    String encodedFileName = fileName;

                    try {
                        encodedFileName = URLEncoder.encode(fileName, "UTF8").replaceAll("\\+", "%20");
                    } catch (UnsupportedEncodingException ex) {
                        // ignore
                    }

                    String filePath = request.getContextPath() + "/web/client/app/" + appDef.getAppId() + "
/" + appDef.getVersion().toString() + "/form/download/" + formDefId + "/" + primaryKeyValue + "/" +
encodedFileName + "." + attachment;

                    if (!result.isEmpty()) {
                        result += ", ";
                    }
                    result += "<a href=\"" + filePath + "\" target=\"_blank\">" + StringUtil.stripAllHtmlTag
(fileName) + "</a>";
                }
            }
        } catch (Exception e) {
            LogUtil.error(getClassName(), e, "");
        }
    }
    return result;
}

```

c.

org.displaytag.util.LookupUtildisplaytagjsp-apiPOM

```

<!-- Change plugin specific dependencies here -->
    <dependency>
        <groupId>javax.servlet</groupId>
        <artifactId>jsp-api</artifactId>
        <version>2.0</version>
    </dependency>
    <dependency>
        <groupId>displaytag</groupId>
        <artifactId>displaytag</artifactId>
        <version>1.2</version>
        <exclusions>
            <exclusion>
                <artifactId>slf4j-api</artifactId>
                <groupId>org.slf4j</groupId>
            </exclusion>
            <exclusion>
                <artifactId>jcl104-over-slf4j</artifactId>
                <groupId>org.slf4j</groupId>
            </exclusion>
            <exclusion>
                <groupId>org.slf4j</groupId>
                <artifactId>slf4j-log4j12</artifactId>
            </exclusion>
        </exclusions>
    </dependency>
<!-- End change plugin specific dependencies here -->

```

d. (18n)

```
getLabelgetDescription18n18n
```

```
"file_link_datalist_formatter / src / main" "resources / messages" "FileLinkDatalistFormatter.properties"
```

```

org.joget.tutorial.FileLinkDatalistFormatter.pluginLabel=File Link Datalist Formatter
org.joget.tutorial.FileLinkDatalistFormatter.pluginDesc=To format the column value as attachment download link.
datalist.fileLinkFormatter.config=Configure File Link Formatter
datalist.fileLinkFormatter.form=Form
datalist.fileLinkFormatter.attachment=Download as Attachment?

```

e. Felix

We will have to register our plugin class in Activator class (Auto generated in the same class package) to tell Felix Framework that this is a plugin.

```

public void start(BundleContext context) {
    registrationList = new ArrayList<ServiceRegistration>();
    //Register plugin here
    registrationList.add(context.registerService(FileLinkDatalistFormatter.class.getName(), new
FileLinkDatalistFormatter(), null));
}

```

f.

```
"file_link_datalist_formatter / target" "file_link_datalist_formatter-5.0.0.jar"
```

```
jar jar
```

Filter by Type **Datalist Column Formatter**

<input type="checkbox"/>	Plugin Name	Plugin Description	Plugin Version
<input type="checkbox"/>	Date Formatter	Format data data with various date format.	5.0.0
<input type="checkbox"/>	Default Formatter	Default Formatter	5.0.0
<input type="checkbox"/>	File Link Datalist Formatter	To format the column value as attachment download link.	5.0.0
<input type="checkbox"/>	Options Value Formatter	Format datalist column with options label.	5.0.0

userviewCRUD

Formatter

General > Action Mapping > **Formatter** > Formatter (File Link Datalist Formatter)

Formatter

File Link Datalist Formatter

Configure File Link Formatter

General > Action Mapping > Formatter > **Configure File Link Formatter**

Form *

testFile

Download as Attachment?

☒

CRUD

10

Show

<input type="checkbox"/>	TextField	FileUpload	
<input type="checkbox"/>	test1	CHANGES.txt	Edit

One item found.

1

CSV

Excel

XML

PDF

8.

[file_link_datalist_formatter.zip](#)

jar<http://marketplace.joget.org/>