
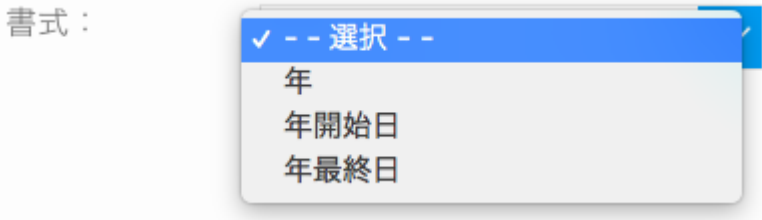





YellowfinInputType

TEXT BOX		<ul style="list-style-type: none">• textClassNameHTMLclass• styleHTMLCSS• numericOnlytrue• numericTypeINTEGER numericOnlytrue• unsignedOnlytrue numericOnlytrue• thousandSeparatorOnnu mericOnlytrue• decimalPlacesnumericOnl ytruenumericType INTEGER• minValue/maxValue• maxLengthnumeric Onlyfalse• readOnlyHTMLreadOnly• disabledtrueHTMLdisabled• nameHTMLname• tooltipHTMLtitle• disableBrowserAutoComplete• placeholder• placeholderCss• placeholderAccountForScrolltrue• suffix• widthcsspx	<pre>ParameterImpl para parameter.setProperty("DEC parameter.setInputType(Ing parameter.setName("Decimal parameter.setMinAllowed(0) parameter.addViewOption("r parameter.addViewOption("r</pre>
SELECT		<ul style="list-style-type: none">• widthcsspx• styleHTMLCSS• size1• multiSelecttrue• selectAllAvailable multiSelecttrue• multiJoinString • disabledtrue• showOptionTitletrue	<pre>ParameterImpl p = p.setProperty("DATE_FORMAT p.setName("Format"); p.setInputType(InputType.5 List<CustomValue<?>> optV optValues.add(new CustomV optValues.add(new CustomV optValues.add(new CustomV Date")); optValues.add(new CustomV Date")); p.setPossibleValues(optVal</pre>

SLIDER AND TEXT		<ul style="list-style-type: none">• <code>maxValue100</code>• <code>textMaxValue</code>• <code>minValue0</code>• <code>textMinValue</code>• <code>sliderWidth</code>	<pre>ParameterImpl p = p.setName("Group Size"); p.setProperty("size"); p.setInputType(InputType.TEXT); p.addViewOption("textMaxValue", 100); p.addViewOption("textMinValue", 0); p.setDefaultValue(5);</pre>
TOGGLE	<p>On/Off</p> <p>最初の行をタイトルに含める</p> 	<ul style="list-style-type: none">• <code>onValuetrue</code>• <code>offValuefalse</code>• <code>disabledtrue</code>	<pre>ParameterImpl p = p.setProperty("FIRST_ROW_TITLE"); p.setName("First row contains title"); p.setDefaultValue(true); p.setInputType(InputType.TEXT);</pre>
FIELD MATCH DROPDOWN	<p>FieldObject2</p> <pre>{ 'from': fieldA.id, 'to': fieldB.id }</pre> 	<ul style="list-style-type: none">• <code>fieldListA</code>• <code>fieldListB</code>• <code>titleAA</code>• <code>titleBB</code>• <code>titleClassAACSS class</code>• <code>titleClassBB</code>• <code>viewOptionsAA</code>• <code>viewOptionsBB</code>	<pre>ParameterImpl p = p.setProperty("fieldMatchField"); p.setList(true); //List of ListOptions lo = new ListOptions(); lo.setAddButtonText("Match"); p.setListOptions(lo); FieldObjectDataType textType; FieldObjectDataType numericType; // LIST A List<FieldObject> fieldListA = new ArrayList<>(); FieldObject field1 = new FieldObject("1", textType); FieldObject field2 = new FieldObject("2", numericType); fieldListA.add(field1); fieldListA.add(field2); // LIST B List<FieldObject> fieldListB = new ArrayList<>(); FieldObject field3 = new FieldObject("3", textType); FieldObject field4 = new FieldObject("4", numericType); FieldObject field5 = new FieldObject("5", textType); FieldObject field6 = new FieldObject("6", numericType); fieldListB.add(field3); fieldListB.add(field4); fieldListB.add(field5); fieldListB.add(field6); // Add the lists to the ParameterImpl p.addViewOption("fieldListA", fieldListA); p.addViewOption("fieldListB", fieldListB); p.addViewOption("titleA", "Title A"); p.addViewOption("titleB", "Title B"); p.setInputType(InputType.TEXT);</pre>
FIELD MATCH DROPDOWN	FIELDMATCHDROPDOWN		


```
{  
  'to': ADDRESSID.id  
}
```

インプット 1

ADDRESSLIN...
ADDRESSTY...
POSTCODE
ADDRESSLIN...
ADDRESSLIN...

ADDRESSID
ADDRESSTY...

インプット 2

ADDRESSLIN...
ADDRESSTY...
POSTCODE
ADDRESSLIN...
ADDRESSLIN...
 ADDRESSID

ADDRESSTY...

- fieldListA
- fieldListB
- titleAA
- titleBB
- titleClassAACSS class
- titleClassBB
- viewOptionsAA
- viewOptionsBB
- includeUnmatchedfalse
- roughMatchfalse
- imageMap
- primary
- dragHandle
- error
- forceAllConnectionstrueif
else

```
ParameterImpl p =  
p.setInputType( InputType.F  
p.setProperty( "UNION_FIEL  
p.addViewOption( "includeUr
```

```
//Match exact types  
p.addViewOption( "roughMatc
```

```
List<FieldObject> foListA  
List<FieldObject> foListB
```

```
FieldObjectDataType textDa  
TEXT;  
FieldObjectDataType numeri  
FieldObjectDataType.NUMERI
```

```
// LIST A  
FieldObject field1 = new F  
"Field 1", textDataType);  
FieldObject field2 = new F  
2", numericDataType);//Thi
```

```
foListA.add(field1);  
foListA.add(field2);
```

```
// LIST A  
FieldObject field3 = new F  
3", textDataType);  
FieldObject field4 = new F  
4", numericDataType);//Thi  
FieldObject field5 = new F  
5", textDataType);  
FieldObject field6 = new F  
6", numericDataType);//Thi
```

```
foListB.add(field3);  
foListB.add(field4);  
foListB.add(field5);  
foListB.add(field6);
```

```
p.addViewOption( "fieldList  
p.addViewOption( "fieldList
```

```
//Input 1  
p.addViewOption( "titleA",
```

```
//Input 2  
p.addViewOption( "titleB",
```

			<div>Input 1</div> <div><div>0000</div><div>Field 1</div><div>Field 2</div></div>
A U T O C O M P L E T E B O X	<div>新しいテーブル名</div> <div><div></div><div>YF_SPREADSHEET_82236</div><div>YFTT75686</div></div> <div>possibleValuesCustomValue</div>	<ul style="list-style-type: none">• initialValueCustomValue value = description =• width• maxDisplayItems4• isDataGrouped• resultListStyleHTMLClass	<pre>ParameterImpl p = p.setProperty("TABLETEXTBC p.setName("New Table Name" p.setInputType(InputType.7 p.addViewOption("width", 2 CustomValue<String> initia CustomValue<String>("all_c p.addViewOption("initialVa List<CustomValue<?>> possi possibleValues.add(new Cus ("all_columns", "all_colum possibleValues.add(new Cus ("all_parameters", "all_pe possibleValues.add(new Cus ("all_sql_modules", "all_s possibleValues.add(new Cus ("all_tables", "all_tables p.setPossibleValues(possik p.addViewOption("width", 2 p.addViewOption("resultLis "outputStepAutoCompleteLis</pre>

HOW ADVANCED

/ParameterDisplayRules

高度な設定 ↓

All Rows

Separator

Less ↑

• showText

• hideText

Boolean value = true;
Boolean negative = false;

// SHOW_ADVANCED_SETTINGS
ParameterDisplayRule advancedRule = new
ParameterDisplayRule("AND",
value, negative);

ParameterImpl p = new ParameterImpl();
p.setProperty("ALL_ROWS");
p.setName("All Rows");
p.setDefaultValue(true);
p.setInputType(InputType.TEXT);
p.addDisplayRule(advancedRule);
paramList.add(p);

p = new ParameterImpl();
p.setProperty("SEPARATOR");
p.setName("Separator");
p.setInputType(InputType.TEXT);
p.addViewOption("width", "100px");
p.addDisplayRule(advancedRule);
paramList.add(p);

p = new ParameterImpl();
p.setProperty("SHOW_ADVANCED_SETTINGS");
p.setName("Show Advanced Settings");
p.setInputType(InputType.CHECKBOX);
p.setDefaultValue(false);
p.addViewOption("showText", true);
p.addViewOption("hideText", false);
p.addViewOption("width", "100px");
paramList.add(p);

PAIR COLUMNS

2

{
column1Key: dataRow1Col1
column2Key: dataRow2Col2
}

ルール

検索

代替テキスト

Farenheit

Celsius

×

miles

km

• column1Header

• column2Header

• column1Key

• column2Key

ParameterImpl p = new ParameterImpl();
p.setProperty("RULES_PARAMETER");
p.setName("Rules");
p.setInputType(InputType.TEXT);
p.addViewOption("column1Header", "検索");
p.addViewOption("column2Header", "代替テキスト");
p.addViewOption("column1Key", "column1Key");
p.addViewOption("column2Key", "column2Key");

List<Map<String, String>> defaults = new ArrayList<>();

Map<String, String> row = new HashMap<>();
row.put("FIND_PARAMETER", "Fahrenheit");
row.put("REPLACE_PARAMETER", "Celsius");
defaults.add(row);

row = new HashMap<>();
row.put("FIND_PARAMETER", "miles");
row.put("REPLACE_PARAMETER", "km");
defaults.add(row);

p.setDefaultValue(defaults);

DELIMITER EXAMPLE

Delimiter: ;


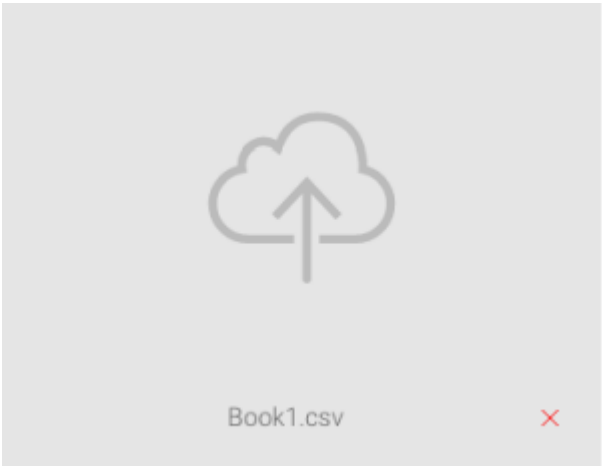
7 Rosella Avenue ; Boronia ; 3155 ; Victoria

• text

ParameterImpl p = new ParameterImpl();
p.setProperty("DELIMITER");
p.setName("Delimiter");
p.addViewOption("text", ";");
p.setInputType(InputType.TEXT);

INPUT LIST		<ul style="list-style-type: none">• inputOptionsInputListView<ul style="list-style-type: none">◦ controlType TEXTBOXTOGGLE CHECKBOX SELECT◦ viewOptions	<pre>Map<String, String> HashMap<>(); viewOptions1.put("property", viewOptions1.get("property")); viewOptions2.put("property", viewOptions2.get("property")); Map<String, Map<String, String>> HashMap<>(); inputOption1.put("controlType", inputOption1.get("controlType")); inputOption1.put("viewOptions", inputOption1.get("viewOptions")); Map<String, Map<String, String>> HashMap<>(); inputOption2.put("controlType", inputOption2.get("controlType")); inputOption2.put("viewOptions", inputOption2.get("viewOptions")); List<Map<String, Map<String, String>>> new ArrayList<>(); ParameterImpl p = new ParameterImpl(); p.setInputType(InputType.TEXTBOXTOGGLE); p.setProperty("FIND_AND_REPLACE", "FIND_AND_REPLACE"); p.addViewOption("inputOptions", inputOptions);</pre>
------------	--	---	---

--	--	--	--	--

FileUploadParameter	<div></div> <div></div>	<div><div>1. <code>public FileUploadParameterString propertyString fileTypeString typeCodeString uploadEvent</code></div><div><code>propertyYellowfin IDID fileTypecsvxml* typeCodeYellowfin uploadEventnull</code></div><div>2. <code>public void setUploadWidgetText(String text)</code></div><div>3. <code>public void setUploadImageBase64 (String base64, String imgType) Base64</code></div><div>4. <code>public void setUploadImageHoverBase64 (String base64, String imgType)</code></div></div> <div><div><div>• <code>inputNameHTML</code></div><div>• <code>showProcessingBarYellowfintrue</code></div><div>blockedURL</div><div>• <code>processingTextshowProcessingBartrue</code></div><div>• <code>width100%</code></div><div>• <code>height100%</code></div><div>• <code>stylesCSSclass</code></div></div><div><div><code>FileUploadParameter p = new FileUploadParameter ("UPLOAD_FILE_KEY", "", "FILE", null); p.setParameterClassName ("left-param"); p.setUploadImageBase64 (getBase64OffImage(), "image /svg+xml"); p.setUploadImageHoverBase64 (getBase64OnImage(), "image /svg+xml"); p.setUploadWidgetText ("Drag File Here");</code></div><div><code>byte[] rawData = this.getFile ("UPLOAD_FILE_KEY");</code></div></div></div>
---------------------	--	--

Passw ordPar ameter	/Yellowfin <div>API Key <input type="password" value="....."/></div>	<ol style="list-style-type: none">1. <code>public PasswordParameter(String property, String value)</code> propertyYellowfin value2. <code>public String</code> getValue3. <code>public void setValue</code> (String value) /4. <code>public String</code> encrypt() throwsExc eption5. <code>public String</code> decrypt() throwsExc eption	<ul style="list-style-type: none">• text Class sName HTML class• style HTML L CSS• disabled HTML True• placeholder• width 190px	<div><pre> PasswordParameter apiKey = new PasswordParameter("API_KEY", "myApiKey"); apiKey.setName("API Key"); apiKey.addViewOption ("width", "190px");</pre></div> <div><pre> List<Parameter> paramsList = getPanelCollection(). getSectionMap().get ("passwordSection"). getParameters(); for (Parameter parameter : paramsList){ if ("API_KEY".equals (parameter.getProperty())){ PasswordParameter pwd = ((PasswordParameter) parameter); pwd.setValue (getStepOption("API_KEY")); try { apiKey = pwd. decrypt(); } catch (Exception e) { throwUnhandledETLException (e); } } }</pre></div>
---------------------------	---	--	--	---