

## 變更履歷

[illegible]

# 属性一覧

2014/04/16

#	属性名	概要	対応要素	html	xml	パラメータ形式	備考
1	data-load	画面表示時に初期表示データを取得する場合に記述する	指定なし(*1)	○	○	JSON(Object/Array)	・複数のタグに指定されていた場合、最初の設定のみ有効となる
2	data-func	formのsubmit時にcrudまたはメール機能を実行する場合に記述する	form	○		JSON(Object/Array)	・入力に依存しない条件は、data-condではなくdata-funcのパラメータとして指定する
3	data-cond	data-funcとセットで使用 crudの条件設定が可能なメソッドを実行する際の条件を付与する為に記述する data-funcのCOND/パラメータは、クライアントからの変更が不可だが、data-condは入力に応じて条件を変更できる	input	○		JSON(Object)	・data-funcが記述されたの子要素で利用する
4	data-iterator	crudの実行結果が複数行の場合、任意の場所で要素を繰り返す際に記述する	指定なし	○	○	文字列	・data-load、data-funcのmethodにて“select”または“selectAll”を指定した場合のbindは、data-iteratorを指定した要素配下に指定する必要がある
5	data-keep	data-iteratorを指定した要素の兄弟要素はデフォルトでは削除されるが、本属性が指定されていた場合は、削除しない	指定なし	○	○	なし	
6	data-bind-in	formの情報をDBに登録、更新する際に記述する	input textarea	○		JSON(Object)	
7	data-bind-out	抽出したデータやリクエストパラメータ等を任意の場所に抽出結果をbindする際に記述する	指定なし	○	○	JSON(Object/Array)	・data-load、data-funcのmethodにて“select”または“selectAll”を指定した場合のbindは、data-iteratorを指定した要素配下に指定する必要がある
8	data-case	任意の条件によって画面の出しわけを行いたい場合に記述する	指定なし	○	○	JSON(Object/Array)	・複数指定はand条件となる
9	data-erase	本属性が指定された要素または子要素を画面に出力しないように設定で	指定なし	○	○	文字列	
10	data-url	リクエストURLパターンを設定する。本属性を省略した場合、所定のルールによりURLが決定される。(詳細は、[data-uri]シート参照)	指定なし(*1)	○	○	JSON(Array)	・複数のタグに指定されていた場合、最初の設定のみ有効となる
11	data-auth	当該画面へのアクセスに認証が必要な場合、本属性を指定する	指定なし(*1)	○	○	なし	
12	data-image	DBに格納された画像パス(非Web領域)の画像を取得する場合に記述する	指定なし	○		JSON(Array)	

(\*1)HTMLの場合、HEAD要素に指定することを推奨する。

data-load

指定の設定値から選択

検索機能				
JSON(Object)	パラメータ名	必須	設定値	説明
{"TYPE": "crud", "METHOD": "select",  "TARGET": "shop", "RESULT": "result1",  "OFFSET": "10",  "LIMIT": "50",  "PAGER": {  "PARAM_NAME": "page", "PERPAGE": "10", "PAGERCOUNT": "3" }, "COND": [ { "COL": "shop.type", "OPE": "eq",  "VALUE": "_REQ.type" } ], "SORT": [ "shop.id desc",  ... ], "NOT_FOUND_ERR": "true"	TYPE		crud	CRUD機能を利用する場合「crud」と指定する。(デフォルト)
	METHOD	○	select selectAll selectByPk	任意の検索条件を指定してtargetに設定したテーブルからデータを抽出する際に指定する 関連テーブル(FK)を含めた全てのデータを任意の検索条件を指定して抽出する際に指定する 対象テーブルの1レコードを抽出する。 data-iteratorを指定せずにdata-bind-outを記述できる。 本メソッドを指定した場合、抽出結果が1件でない場合、エラーとなる。
	TARGET	○	テーブル名	処理対象のテーブル名を指定する
	RESULT	○	任意の文字列	検索結果へのアクセスで使用する識別子 ※ページ内で一意となる名称とする
	OFFSET		任意の数値(int)	取得を開始する位置を指定する ※"METHOD"が"select"または"selectAll"の場合、有効 ※"SORT"を指定しない場合、返却される行は保証されない
	LIMIT		任意の数値(int)	取得件数を指定する ※"METHOD"が"select"または"selectAll"の場合、有効 ※"SORT"を指定しない場合、返却される行は保証されない
	PAGER		Object	ページ情報オブジェクト ページを表示する場合に指定する ※ページ情報の使用方法は、【補足】pager仕様参照。 ※"METHOD"が"select"または"selectAll"の場合、有効
	PARAM_NAME		任意の文字列	ページ番号のパラメータ名を指定する 未指定の場合は、「page」を採用する
	PERPAGE		任意の数値(int)	1ページあたりの最大行数(デフォルトは、20)
	PAGERCOUNT		任意の数値(int)	ページ最大表示数(当該ページの前後の表示数)(デフォルトは、5)
	COND		Array Object	検索条件配列 検索条件オブジェクト
	COL	△	テーブル名.カラム名	条件として付与するテーブル名.カラム名を指定する(デリミタは、ドット)
	OPE	△	isnull	is null
			isnotnull	is not null
			eq	=
			nq	<>
			gt	>
			ge	>=
			lt	<
			le	<=
			%like	like 'xxxx'(前方一致)
			%like%	like 'xxx%(中間一致)
			like%	like 'xxxx'(後方一致)
			%notlike	not like 'xxxx'
			%notlike%	not like 'xxx%'
			notlike%	not like 'xxx%'
			in	in (1,2,3,4)
			notin	not in (1,2,3,4)
	VALUE	△	_REQ.リクエストパラメータ名 または _SES.LOGIN_USER.認証テーブルのカラム名 または _CONST.定数Key	検索条件の設定元を指定する。 リクエストパラメータを指定する場合: "_REQ."パラメータ名 認証情報を指定する場合: _SES.LOGIN_USER.認証テーブルのカラム名 定数を指定する場合: "_CONST."定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	SORT		Array	ソート条件配列 ※"METHOD"が"select"または"selectAll"の場合、有効
	-		テーブル名.カラム名 (asc/desc) または _REQ.リクエストパラメータ名 または _CONST.定数Key	ソート対象のカラム名および、昇順or降順を指定する(デフォルトは、昇順) リクエストパラメータを指定する場合: "_REQ."パラメータ名 定数を指定する場合: "_CONST."定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	NOT_FOUND_ERR		true/false	データが抽出できなかった場合、404ページに遷移するか否かを指定する (デフォルトは、false。404エラーとならない)

insert機能				
JSON(Object)	パラメータ名	必須	設定値	説明
{"TYPE": "crud", "METHOD": "insert", "TARGET": "shop", "RESULT": "result1",  "COL": [ { "COL": "shop.type", "VALUE": "_REQ.type" } ], "TOKEN": "true"	TYPE		crud	デフォルト
	METHOD	○	insert	データを登録する場合、「insert」と指定する
	TARGET	○	テーブル名	処理対象のテーブル名を指定する
	RESULT	○	任意の文字列	data-bind-inで指定する識別子 (data-loadでinsertを指定した場合は、data-bind-inは指定不可の為、左記の値は参照されない) ※ページ内で一意となる名称とする
	COL		Array Object	登録値配列 登録オブジェクト
	COL	△	テーブル名.カラム名	テーブル名.カラム名を指定する(デリミタは、ドット)
	VALUE	△	_REQ.リクエストパラメータ名 または _SES.LOGIN_USER.認証テーブルのカラム名 または _CONST.定数Key	設定元を指定する リクエストパラメータを指定する場合: "_REQ."パラメータ名 認証情報を指定する場合: _SES.LOGIN_USER.認証テーブルのカラム名 定数を指定する場合: "_CONST."定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	TOKEN		true/false	TOKENチェックを行うか否か(デフォルトは、true)

update機能				
JSON(Object)	パラメータ名	必須	設定値	説明
<pre>{   "TYPE": "crud",   "METHOD": "update",   "TARGET": "shop",   "RESULT": "result1",    "COL": [     {       "COL": "shop.type",       "VALUE": "_REQ.type"     }   ],   "COND": [     {       "COL": "shop.type",       "OPE": "eq",        "VALUE": "_REQ.type"     }   ],   "TOKEN": "true" }</pre>	TYPE		crud	デフォルト
	METHOD	○	update	データを更新する場合、「update」と指定する
	TARGET	○	テーブル名	処理対象のテーブル名を指定する
	RESULT	○	任意の文字列	data-bind-in、data-condで指定する識別子 (data-loadでupdateを指定した場合は、data-bind-in、data-condは指定不可の為、左記の値は参照されない) ※ページ内で一意となる名称とする
	COL		Array Object	更新値配列 更新オブジェクト
	COL	△	テーブル名.カラム名	テーブル名.カラム名を指定する(デリミタは、ドット)
	VALUE	△	_REQ.リクエストパラメータ名 または _SES.LOGIN_USER.認証テーブルのカラム名 または CONST.定数Key	設定元を指定する リクエストパラメータを指定する場合: "_REQ."パラメータ名 認証情報を指定する場合: _SES.LOGIN_USER.認証テーブルのカラム名 定数を指定する場合: "_CONST."定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	COND		Array Object	更新条件配列 更新条件オブジェクト
	COL	△	テーブル名.カラム名	条件として付与するテーブル名.カラム名を指定する(デリミタは、ドット)
	OPE	△	isnull isnotnull eq nq gt ge lt le %like %like% like% %notlike %notlike% notlike% in notin	is null is not null = <> > >= < <= like 'xxxx'(前方一致) like 'xxxx%(中間一致) like 'xxxx%(後方一致) not like 'xxxx' not like 'xxx%' not like 'xxx%' in (1,2,3,4) not in (1,2,3,4)
	VALUE	△	_REQ.リクエストパラメータ名 または _SES.LOGIN_USER.認証テーブルのカラム名 または CONST.定数Key	設定元を指定する リクエストパラメータを指定する場合: "_REQ."パラメータ名 認証情報を指定する場合: _SES.LOGIN_USER.認証テーブルのカラム名 定数を指定する場合: "_CONST."定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	TOKEN		true/false	TOKENチェックを行うか否か(デフォルトは、true)

delete機能				
JSON(Object)	パラメータ名	必須	設定値	説明
<pre>{   "TYPE": "crud",   "METHOD": "delete",   "TARGET": "shop",   "RESULT": "result1",    "COND": [     {       "COL": "shop.type",       "OPE": "eq",        "VALUE": "_REQ.type"     }   ],   "TOKEN": "true" }</pre>	TYPE		crud	デフォルト
	METHOD	○	delete	データを物理削除する場合、「delete」と指定する
	TARGET	○	テーブル名	処理対象のテーブル名を指定する
	RESULT	○	任意の文字列	data-condで指定する識別子 (data-loadでdeleteを指定した場合は、data-condは指定不可の為、左記の値は参照されない) ※ページ内で一意となる名称とする
	COND		Array Object	削除条件配列 削除条件オブジェクト
	COL	△	テーブル名.カラム名	条件として付与するテーブル名.カラム名を指定する(デリミタは、ドット)
	OPE	△	isnull isnotnull eq nq gt ge lt le %like %like% like% %notlike %notlike% notlike% in notin	is null is not null = <> > >= < <= like 'xxxx'(前方一致) like 'xxxx%(中間一致) like 'xxxx%(後方一致) not like 'xxxx' not like 'xxx%' not like 'xxx%' in (1,2,3,4) not in (1,2,3,4)
	VALUE	△	_REQ.リクエストパラメータ名 または _SES.LOGIN_USER.認証テーブルのカラム名 または CONST.定数Key	設定元を指定する リクエストパラメータを指定する場合: "_REQ."パラメータ名 認証情報を指定する場合: _SES.LOGIN_USER.認証テーブルのカラム名 定数を指定する場合: "_CONST."定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	TOKEN		true/false	TOKENチェックを行うか否か(デフォルトは、true)

ストアドプロシージャ実行				
JSON(Object)	パラメータ名	必須	設定値	説明
<pre>{   "TYPE": "procedure",   "TARGET": "shop",   "RESULT": "result1",    "OFFSET": "10",   "LIMIT": "50",   "PAGER": {      "PARAM_NAME": "page",      "PERPAGE": "10",     "PAGERCOUNT": "3"   },   "ARGS": [     {       "NAME": "type",        "VALUE": "_REQ.type"     }   ],   "NOT_FOUND_ERR": "true" }</pre>	TYPE		procedure	プロシージャを利用する場合「procedure」と指定する
	TARGET	○	ストアドプロシージャ名	実行対象のストアドプロシージャ名を指定する
	RESULT	○	任意の文字列	検索結果へのアクセスで使用する識別子 ※ページ内で一意となる名称とする
	OFFSET		任意の数値(int)	取得を開始する位置を指定する
	LIMIT		任意の数値(int)	取得件数を指定する
	PAGER		Object	ページ情報オブジェクト ページを表示する場合に指定する ※ページ情報の使用方法は、 <a href="#">【{補足} pager仕様】参照</a> 。
	PARAM_NAME		任意の文字列	ページ番号のパラメータ名を指定する 未指定の場合は、「page」を採用する
	PERPAGE		任意の数値(int)	1ページあたりの最大行数(デフォルトは、20)
	PAGERCOUNT		任意の数値(int)	ページ最大表示数(当該ページの前後の表示数)(デフォルトは、5)
	ARGS		Array	ストアドプロシージャ引数配列
	COL	△	Object	ストアドプロシージャ引数オブジェクト ストアドプロシージャ引数の名称 ※ストアドプロシージャ宣言時の名称と一致している必要がある
	VALUE	△	REQ.リクエストパラメータ名 または SES.LOGIN_USER.認証テーブルのカラム名 または CONST.定数Key	ストアドプロシージャ引数の設定元を指定する。 リクエストパラメータを指定する場合: "_REQ."パラメータ名 認証情報を指定する場合: SES.LOGIN_USER.認証テーブルのカラム名 定数を指定する場合: "_CONST."定数Key(全カテコリが指定可能。category未指定の場合は、keyのみを指定する)
	NOT_FOUND_ERR		true/false	データが抽出できなかった場合、404ページに遷移するか否かを指定する ※ストアドプロシージャの返却がvoidでない場合に有効 (デフォルトは、false。404エラーとならない)

JSON(Array)	パラメータ名	必須	設定値	説明
<pre>[ {   "METHOD": "select",   "TARGET": "company",   "RESULT": "companyList" }, {   "METHOD": "select",   "TARGET": "shop",   "RESULT": "shopList" } ... ]</pre>				「JSON(Object)」参照

Sample				
◆画面表示時にcompanyテーブルから全データを抽出し、表示する				
<pre>&lt;head data-load="{\"METHOD\":\"select\",\"TARGET\":\"shop\",\"RESULT\":\"result1\"}"&gt;</pre> <pre>&lt;ul&gt;</pre> <pre>&lt;li data-iterator="result1"&gt;&lt;b data-bind-out="{\"RESULT\":\"result1\",\"TARGET\":\"text\",\"ARG\":\"shop.name\"}"&gt;サンプル&lt;/b&gt;&lt;/li&gt;</pre> <pre>&lt;/ul&gt;</pre>				
◆画面表示時にリクエストパラメータ、定数からそれぞれ条件を指定し、shopテーブルおよびFKが設定されたテーブルのデータを抽出する				
<pre>&lt;head data-load="{\"METHOD\":\"selectAll\",\"TARGET\":\"shop\",\"RESULT\":\"result1\",\"COND\": [{\"COL\":\"shop.type\",\"OPE\":\"eq\",\"VALUE\":\"_REQ.type\"},{\"COL\":\"shop.del_flag\",\"OPE\":\"ne\",\"VALUE\":\"_CONST.del\"}]}"&gt;</pre> <pre>&lt;li data-iterator="result1" data-bind-out="{\"RESULT\":\"result1\",\"TARGET\":\"text\",\"ARG\":\"shop.name\"}"&gt;ショップ名</pre> <pre>&lt;ul&gt;</pre> <pre>&lt;li data-iterator="result1.shop.review" data-bind-out="{\"RESULT\":\"result1\",\"TARGET\":\"text\",\"ARG\":\"review.content\"}"&gt;レビュー&lt;/li&gt;</pre> <pre>&lt;/ul&gt;</pre> <pre>&lt;/li&gt;</pre>				

data-func

指定の設定値から選択

検索機能				
JSON(Object)	パラメータ名	必須	設定値	説明
{"TYPE": "crud", "METHOD": "select",  "TARGET": "shop", "RESULT": "result1",  "OFFSET": "10",  "LIMIT": "50",  "PAGER": {  "PARAM_NAME": "page", "PERPAGE": "10", "PAGERCOUNT": "3" }, "COND": [ {"COL": "shop.type", "OPE": "eq",  "VALUE": "._REQ.type" }], "SORT": [ "shop.id desc",  ... ], "TOKEN": "true"	TYPE		crud	CRUD機能を利用する場合「crud」と指定する。(デフォルト)
	METHOD	○	select selectAll selectByPk	任意の検索条件を指定してtargetに設定したテーブルからデータを抽出する際に指定する 関連テーブル(FK)を含めた全てのデータを任意の検索条件を指定して抽出する際に指定する 対象テーブルの1レコードを抽出する。 data-iteratorを指定せずにdata-bind-outを記述できる。 本メソッドを指定した場合、抽出結果が1件でない場合、エラーとなる。
	TARGET	○	テーブル名	処理対象のテーブル名を指定する
	RESULT	○	任意の文字列	検索結果へのアクセス、条件指定で使用する識別子 ※ページ内で一意となる名称とする
	OFFSET		任意の数値(int)	取得を開始する位置を指定する ※「METHOD」が「select」または「selectAll」の場合、有効 ※「SORT」を指定しない場合、返却される行は保証されない
	LIMIT		任意の数値(int)	取得件数を指定する ※「METHOD」が「select」または「selectAll」の場合、有効 ※「SORT」を指定しない場合、返却される行は保証されない
	PAGER		Object	ページ情報オブジェクト ページを表示する場合に指定する ※ページ情報の使用方法は、【補足】pager仕様参照。 ※「METHOD」が「select」または「selectAll」の場合、有効
	PARAM_NAME		任意の文字列	ページ番号のパラメータ名を指定する 未指定の場合は、「page」を採用する
	PERPAGE		任意の数値(int)	1ページあたりの最大行数(デフォルトは、20)
	PAGERCOUNT		任意の数値(int)	ページ最大表示数(当該ページの前後の表示数)(デフォルトは、5)
	COND		Array	検索条件配列
			Object	検索条件オブジェクト
	COL	△	テーブル名.カラム名	条件として付与するテーブル名.カラム名を指定する(デリミタは、ドット)
	OPE	△	isnull	is null
			isnotnull	is not null
			eq	=
			nq	<>
			gt	>
			ge	>=
			lt	<
			le	<=
			%like	like 'xxxx'(前方一致)
			%like%	like 'xxx%'(中間一致)
			like%	like 'xxxx'(後方一致)
			%notlike	not like 'xxxx'
			%notlike%	not like 'xxx%'
			notlike%	not like 'xxx%'
			in	in (1,2,3,4)
			notin	not in (1,2,3,4)
	VALUE	△	._REQ.リクエストパラメータ名 または ._SES.LOGIN_USER.認証テーブルのカラム名 または ._CONST.定数Key	検索条件の設定元を指定する。 リクエストパラメータを指定する場合:「._REQ.」パラメータ名 認証情報を指定する場合: _SES.LOGIN_USER.認証テーブルのカラム名 定数を指定する場合:「._CONST.」定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	SORT		Array	ソート条件配列
				※「METHOD」が「select」または「selectAll」の場合、有効
	-		テーブル名.カラム名 (asc/desc) または ._REQ.リクエストパラメータ名 または ._CONST.定数Key	ソート対象のカラム名および、昇順or降順を指定する(デフォルトは、昇順) リクエストパラメータを指定する場合:「._REQ.」パラメータ名 定数を指定する場合:「._CONST.」定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	TOKEN		true/false	TOKENチェックを行うか否か(デフォルトは、true)

insert機能				
JSON(Object)	パラメータ名	必須	設定値	説明
{"TYPE": "crud", "METHOD": "insert", "TARGET": "shop", "RESULT": "result1",  "COL": [ {"COL": "shop.type", "VALUE": "._REQ.type" }], "TOKEN": "true"	TYPE		crud	デフォルト
	METHOD	○	insert	データを登録する場合、「insert」と指定する
	TARGET	○	テーブル名	処理対象のテーブル名を指定する
	RESULT	○	任意の文字列	data-bind-inで指定する識別子 ※ページ内で一意となる名称とする
	COL		Array	登録値配列
			Object	登録オブジェクト
	COL	△	テーブル名.カラム名	テーブル名.カラム名を指定する(デリミタは、ドット)
	VALUE	△	._REQ.リクエストパラメータ名 または ._SES.LOGIN_USER.認証テーブルのカラム名 または ._CONST.定数Key	設定元を指定する リクエストパラメータを指定する場合:「._REQ.」パラメータ名 認証情報を指定する場合: _SES.LOGIN_USER.認証テーブルのカラム名 定数を指定する場合:「._CONST.」定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	TOKEN		true/false	TOKENチェックを行うか否か(デフォルトは、true)

update機能				
JSON(Object)	パラメータ名	必須	設定値	説明
<pre>{   "TYPE": "crud",   "METHOD": "update",   "TARGET": "shop",   "RESULT": "result1",    "COL": [     {       "COL": "shop.type",       "VALUE": "_REQ.type"     }   ],   "COND": [     {       "COL": "shop.type",       "OPE": "eq"     }   ],    "VALUE": "_REQ.type" }, "TOKEN": "true"</pre>	TYPE		crud	デフォルト
	METHOD	<input type="radio"/>	update	データを更新する場合、「update」と指定する
	TARGET	<input type="radio"/>	テーブル名	処理対象のテーブル名を指定する
	RESULT	<input type="radio"/>	任意の文字列	data-bind-in、data-condで指定する識別子 ※ページ内で一意となる名称とする
	COL		Array Object	更新値配列 更新オブジェクト
	COL	<input checked="" type="checkbox"/>	テーブル名.カラム名	テーブル名.カラム名を指定する(デリミタは、ドット)
	VALUE	<input checked="" type="checkbox"/>	REQ.リクエストパラメータ名 または SES.LOGIN_USER.認証テーブルのカラム名 または CONST.定数Key	設定元を指定する リクエストパラメータを指定する場合: "_REQ."パラメータ名 認証情報を指定する場合: SES.LOGIN_USER.認証テーブルのカラム名 定数を指定する場合: "_CONST."定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	COND		Array Object	更新条件配列 更新条件オブジェクト
	COL	<input checked="" type="checkbox"/>	テーブル名.カラム名	条件として付与するテーブル名.カラム名を指定する(デリミタは、ドット)
	OPE	<input checked="" type="checkbox"/>	isnull	is null
			isnotnull	is not null
			eq	=
			nq	<>
			gt	>
			ge	>=
			lt	<
			le	<=
			%like	like 'xxxx'(前方一致)
			%like%	like 'xxx%(中間一致)
			like%	like 'xxxx%(後方一致)
			%notlike	not like 'xxxx'
			%notlike%	not like 'xxx%
			notlike%	not like 'xxx%
			in	in (1,2,3,4)
			notin	not in (1,2,3,4)
	VALUE	<input checked="" type="checkbox"/>	REQ.リクエストパラメータ名 または SES.LOGIN_USER.認証テーブルのカラム名 または CONST.定数Key	設定元を指定する リクエストパラメータを指定する場合: "_REQ."パラメータ名 認証情報を指定する場合: SES.LOGIN_USER.認証テーブルのカラム名 定数を指定する場合: "_CONST."定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	TOKEN		true/false	TOKENチェックを行うか否か(デフォルトは、true)

delete機能				
JSON(Object)	パラメータ名	必須	設定値	説明
<pre>{   "TYPE": "crud",   "METHOD": "delete",   "TARGET": "shop",   "RESULT": "result1",    "COND": [     {       "COL": "shop.type",       "OPE": "eq"     }   ],    "VALUE": "_REQ.type" }, "TOKEN": "true"</pre>	TYPE		crud	デフォルト
	METHOD	<input type="radio"/>	delete	データを物理削除する場合、「delete」と指定する
	TARGET	<input type="radio"/>	テーブル名	処理対象のテーブル名を指定する
	RESULT	<input type="radio"/>	任意の文字列	data-condで指定する識別子 ※ページ内で一意となる名称とする
	COND		Array Object	削除条件配列 削除条件オブジェクト
	COL	<input checked="" type="checkbox"/>	テーブル名.カラム名	条件として付与するテーブル名.カラム名を指定する(デリミタは、ドット)
	OPE	<input checked="" type="checkbox"/>	isnull	is null
			isnotnull	is not null
			eq	=
			nq	<>
			gt	>
			ge	>=
			lt	<
			le	<=
			%like	like 'xxxx'(前方一致)
			%like%	like 'xxxx%(中間一致)
			like%	like 'xxxx%(後方一致)
			%notlike	not like 'xxxx'
			%notlike%	not like 'xxx%
			notlike%	not like 'xxx%
			in	in (1,2,3,4)
			notin	not in (1,2,3,4)
	VALUE	<input checked="" type="checkbox"/>	REQ.リクエストパラメータ名 または SES.LOGIN_USER.認証テーブルのカラム名 または CONST.定数Key	設定元を指定する リクエストパラメータを指定する場合: "_REQ."パラメータ名 認証情報を指定する場合: SES.LOGIN_USER.認証テーブルのカラム名 定数を指定する場合: "_CONST."定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	TOKEN		true/false	TOKENチェックを行うか否か(デフォルトは、true)

ストアドロシー ज्या実行				
JSON(Object)	パラメータ名	必須	設定値	説明
[ "TYPE": "procedure", "TARGET": "shop", "RESULT": "result1",  "OFFSET": "10", "LIMIT": "50", "PAGER": {  "PARAM_NAME": "page",  "PERPAGE": "10", "PAGERCOUNT": "3" }, "ARGS": [ { "NAME": "type",  "VALUE": "_REQ.type" } ] ]	TYPE	<input type="radio"/>	procedure	プロシー ज्याの場合「procedure」と指定する
	TARGET	<input type="radio"/>	ストアドロシー ज्या名	実行対象のストアドロシー ज्या名を指定する
	RESULT	<input type="radio"/>	任意の文字列	検索結果へのアクセスで使用する識別子 ※ページ内で一意となる名称とする
	OFFSET		任意の数値(int)	取得を開始する位置を指定する
	LIMIT		任意の数値(int)	取得件数を指定する
	PAGER		Object	ページ情報オブジェクト
				ページを表示する場合に指定する ※ページ情報の使用方法は、 <a href="#">[(補足) pager仕様]</a> 参照。
	PARAM_NAME		任意の文字列	ページ番号のパラメータ名を指定する 未指定の場合は、「page」を採用する
	PERPAGE		任意の数値(int)	1ページあたりの最大行数(デフォルトは、20)
	PAGERCOUNT		任意の数値(int)	ページ最大表示数(当該ページの前後の表示数)(デフォルトは、5)
	ARGS		Array	ストアドロシー ज्या引数配列
			Object	ストアドロシー ज्या引数オブジェクト
	COL	<input checked="" type="checkbox"/>	任意の文字列	ストアドロシー ज्या引数の名称 ※ストアドロシー ज्या宣言時の名称と一致している必要がある
	VALUE	<input checked="" type="checkbox"/>	REQ.リクエストパラメータ名 または SES.LOGIN_USER 認証テーブルのカラム名 または CONST.定数Key	ストアドロシー ज्या引数の設定元を指定する。 リクエストパラメータを指定する場合: "_REQ."/パラメータ名 認証情報を指定する場合: SES.LOGIN_USER 認証テーブルのカラム名 定数を指定する場合: "_CONST."定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)

メール送信機能				
JSON(Object)	パラメータ名	必須	設定値	説明
[ "TYPE": "sendmail", "METHOD": "inquiry mail", "TARGET": "inquiry", "RESULT": "result1",  "ADDRESS_COL": "shop.address", "ID_COL": "ids", "TO": [ "admin_to", ... ], "CC": [ "admin_cc", ... ], "BCC": [ "admin_bcc", ... ], ]	TYPE	<input type="radio"/>	sendmail	メール送信を行う場合「sendmail」と指定する
	METHOD	<input type="radio"/>	メールテンプレート名	templatesフォルダに格納したメールテンプレートファイル名(*.ftl)
	TARGET		テーブル名	送信内容を保存するテーブル名を指定する。送信内容を保存しない場合は、不要
	RESULT	<input type="radio"/>	任意の文字列	送信内容を特定する際に使用する識別子 ※ページ内で一意となる名称とする
	ADDR_COL	<input checked="" type="checkbox"/>	送信先アドレスカラム	送信先のアドレスのテーブルカラム名を指定する
	ID_COL	<input checked="" type="checkbox"/>	送信先id保存カラム	送信先アドレスが保存されたテーブルのPKを保存するカラム名を指定する
	TO	<input checked="" type="checkbox"/>	toアドレスのkey配列	id_colを指定しない場合、必須 toメールアドレスをシステム定数から取得し設定する場合、system_constanテーブルのCategory.Keyを指定する "CONST."定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	CC		ccアドレスKey配列	ccメールアドレスをシステム定数から取得し設定する場合、system_constanテーブルのCategory.Keyを指定する CONST.定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	BCC		bccアドレスKey配列	bccメールアドレスをシステム定数から取得し設定する場合、system_constanテーブルのCategory.Keyを指定する CONST.定数Key(全カテゴリが指定可能。category未指定の場合は、keyのみを指定する)
	-			

JSON(Array)	必須	設定値	説明
[ { "TYPE": "crud", "METHOD": "select", "TARGET": "shop", "RESULT": "result1" }, { "TYPE": "sendmail", "METHOD": "inquiry mail", "RESULT": "result2", "TO": [ "admin_to" ] }, ... ]			「JSON(Object)」参照

Sample



data-cond

指定の設定値から選択

JSON(Object)	パラメータ名	必須	設定値	説明
{"RESULT": "result1", "COL": "shop.type", "OPE": "eq", "VALUE": "_CONST.type"}	RESULT	○	data-funcで指定したresult識別	どのdata-funcに対する条件なのかを識別する為に指定する
	COL	○	テーブル名.カラム名	条件として付与するテーブル名.カラム名を指定する(デリミタは、ドット)
	OPE	○	isnull	is null
			isnotnull	is not null
			eq	=
			nq	<>
			gt	>
			ge	>=
			lt	<
			le	<=
			%like	like 'xxxx'(前方一致)
			%like%	like 'xxxx%(中間一致)
			like%	like 'xxxx%(後方一致)
			%notlike	not like 'xxxx'
			%notlike%	not like 'xxx%
			notlike%	not like 'xxx%
			in	in (1,2,3,4)
			notin	not in (1,2,3,4)
	VALUE		REQ.リクエストパラメータ名 または SES.LOGIN.USER.認証テーブル のカラム名 または CONST.定数Key	検索条件に初期値を設定する場合に指定する 常にinputタグの入力を条件とする場合は、指定しない リクエストパラメータを指定する場合: " REQ."パラメータ名 認証情報を指定する場合: SES.LOGIN.USER.認証テーブルのカラム名 定数を指定する場合: "_CONST."定数Key(viewカテゴリのKeyのみ指定可能)

Sample
<p>◆入力値を検索条件とする場合</p> <pre>&lt;form action="/../html/shop/list.html" method="POST" id="listForm" data-func="{\"TYPE\":\"crud\",\"TARGET\":\"shop\",\"METHOD\":\"select\",\"RESULT\":\"result1\"}&gt;   &lt;input type="text" data-cond="{\"RESULT\":\"result1\",\"COL\":\"shop.name\",\"OPE\":\"%like%\"}&gt;</pre>
<p>◆リクエストパラメータを初期の検索条件とする場合</p> <pre>&lt;input type="hidden" data-cond="{\"RESULT\":\"result1\",\"COL\":\"shop.id\",\"OPE\":\"in\",\"VALUE\":\"_REQ.ids\"} /&gt;</pre>
<p>◆認証情報を初期の検索条件とする場合</p> <pre>&lt;input type="hidden" data-cond="{\"RESULT\":\"result1\",\"COL\":\"shop.id\",\"OPE\":\"eq\",\"VALUE\":\"_SES.LOGIN.USER.id\"} /&gt;</pre>
<p>◆CONSTを初期の検索条件とする場合</p> <pre>&lt;input type="hidden" data-cond="{\"RESULT\":\"result1\",\"COL\":\"shop.flag\",\"OPE\":\"eq\",\"VALUE\":\"_CONST.flag\"} /&gt;</pre>

# data-iterator

パラメータ	必須	設定値	説明
result識別子(.FKテーブル名)	○	result識別子 または result識別 子.PAGE_NO_LI ST	data-loadまたはdata-funcで指定したresult識別子 METHODが“selectAll”でtarget に対しFKが指定されてた子テーブルが存在する場合、子テーブルの テーブル名を付与することで多重繰り返しが可能。 ページャ情報のページング番号リストを指定する場合は、「result識別子.PAGE_NO_LIST」を指定する。

Sample
◆多重iterator例 <pre>&lt;ul class="TMP_L_0001" data-case="{\"TYPE\":\"size\",\"ARG\":[\"result1\",\"gt\",\"0\"]}&gt;   &lt;li data-iterator="result1" data-bind="{\"TARGET\":\"text\",\"ARG\":\"result1.shop.name\"}&gt;ショップ名     &lt;ul&gt;       &lt;li data-iterator="result1.shop.review" data-bind="{\"TARGET\":\"text\",\"ARG\":\"result1.shop.review.content\"}&gt;レビュー&lt;/li&gt;     &lt;/ul&gt;   &lt;/li&gt; &lt;/ul&gt;</pre>

特筆事項
・data-iteratorが指定された要素以外の兄弟要素は削除される(data-keepが設定されている場合を除く)

# data-keep

パラメータ	必須	設定値	説明
"true" 固定			

Sample
<pre>&lt;ul class="TMP_L_0001" data-case="{\"TYPE\":\"size\",\"ARG\":[\"result1\",\"gt\",\"0\"]}&gt;   &lt;li data-keep="true"&gt;ショップ名&lt;/li&gt;   &lt;li data-iterator="result1" data-bind="{\"TARGET\":\"text\",\"ARG\":\"result1.shop.name\"}&gt;ショップ名&lt;/li&gt; &lt;/ul&gt;</pre>

data-bind-in

JSON(Object)	パラメータ名	必須	設定値	説明
{"RESULT": "result1", "ARG": "shop.type", "VALUE": "_CONST.type"}	RESULT	○	data-funcで指定したresult識別	どのdata-funcの登録、更新情報とするかを識別する為に指定する
	ARG	○	テーブル名.カラム名 または PAGER.ページャー情報Key	登録、更新対象のカラムを指定する(デリミタは、ドット) ページャーの情報を指定する場合: "_PAGER".ページャー情報Key(【補足】pager]シート参照)
	VALUE		REQ.リクエストパラメータ名 または SES.LOGIN.USER.認証テーブル のカラム名 または CONST.定数Key	初期値を設定する場合に指定する 常にinputタグの入力を条件とする場合は、指定しない リクエストパラメータを指定する場合: "_REQ."パラメータ名 認証情報を指定する場合: SES.LOGIN.USER.認証テーブルのカラム名 定数を指定する場合: "_CONST."定数Key(viewカテゴリのKeyのみ指定可能)
Sample				
◆input text				
<form action="../../html/shop/list.html" method="POST" id="listForm" data-func={"TYPE": "crud", "TARGET": "shop", "METHOD": "insert", "RESULT": "result1"}>				
<input type="text" data-bind-in={"RESULT": "result1", "ARG": "shop.name"}>ショップ名				
<input type="hidden" data-bind-in={"RESULT": "result1", "ARG": "shop.type", "DEFAULT": "_CONST.type"}>				
◆textarea				
<form action="../../html/shop/list.html" method="POST" id="listForm" data-func={"TYPE": "crud", "TARGET": "shop", "METHOD": "update", "RESULT": "result1"}>				
<textarea rows="4" cols="40" data-bind-in={"RESULT": "result1", "ARG": "shop.summary"}>ショップ概要</textarea>				

data-bind-out

指定の設定値から選択				
JSON(Object)	パラメータ名	必須	設定値	説明
<pre>[   {     "RESULT": "result1",     "TARGET": "text",     "FORMAT": "/[0]/[1]/",     "ARG": "shop.name",     "ARGS": [       "shop.id",     ],     "MAP_KEY": "_CONST.view.stars",     "ESCAPE": "false",     "ERASE_TAG": "false",   }, ]</pre>	RESULT		data-loadまたはdata-funcで指定したresult識別子	どのdata-load、data-funcの結果を出力するかを識別する為に指定する bindする内容がdata-load、data-funcの結果でない場合、指定しない
	TARGET	○	bind対象の属性名 または "text" または "inner"	src、hrefなどbind対象の属性名を指定する "text"を指定した場合、data-bind属性が設定された要素のtextが置き換わる "inner"を指定した場合、配下の属性が全て置き換わる
	FORMAT		任意の文字列 または DATE[フォーマット] または NUMBER[フォーマット]	未指定の場合、抽出したデータをそのままbindする ただし、bind対象先に置換文字列[0]が含まれていた場合、bind対象先をformatとする また、[ ORIGINAL]が含まれていた場合、置換対象の属性の元データを付与する 日付、数値フォーマットについては、後述。
	ARG	△	テーブル名.カラム名 または .REQ.リクエストパラメータ名 または .SES.LOGIN_USER 認証テーブル の または .CONST.定数Key または -	bind対象のカラムが1つであった場合、argsではなく、argを指定する DBの値をbindする場合：取得元テーブル名.カラム名を指定する リクエストパラメータをbindする場合：".REQ."パラメータ名を指定する 認証情報を指定する場合：.SES.LOGIN_USER 認証テーブルのカラム名 定数をbindする場合：".CONST."定数Keyを指定する(viewカテゴリのKeyのみ指定可能) ページャーの情報を指定する場合：".PAGER"ページャー情報Key([[補足]pager]シート参照)
	ARGS	△	- テーブル名.カラム名 または .REQ.リクエストパラメータ名 または .SES.LOGIN_USER 認証テーブル の または .CONST.定数Key または -	bindするパラメータ配列を指定する(formatに指定された置換文字列数分指定する必要がある) DBの値をbindする場合：取得元テーブル名.カラム名を指定する リクエストパラメータをbindする場合：".REQ."パラメータ名を指定する 認証情報を指定する場合：.SES.LOGIN_USER 認証テーブルのカラム名 定数をbindする場合：".CONST."定数Keyを指定する(viewカテゴリのKeyのみ指定可能) ページャーの情報を指定する場合：".PAGER"ページャー情報Key([[補足]pager]シート参照)
	VALUE		.REQ.リクエストパラメータ名 または .CONST.定数Key	初期値を設定する場合に指定する ※ARGを指定した場合にのみ、設定可能 リクエストパラメータを指定する場合：".REQ."パラメータ名 定数を指定する場合：".CONST."定数Key(viewカテゴリのKeyのみ指定可能)
	MAP_KEY		.CONST.定数Key	ARGで指定された内容を元に取得した値をKeyとして、webappos_system_constantに定義した情報を取得する。 webappos_system_constantのdata_typeを[map]のCONSTを指定する
	ESCAPE		true false	DBにHTMLタグが登録されていた場合、HTMLタグをエスケープする(デフォルト) DBにHTMLタグが登録されていた場合、HTMLタグをエスケープしない ※DBの値をbindした場合のみ有効
	ESCAPE		true false	HTMLタグが含まれた文字列をBindする場合、HTMLタグを削除する HTMLタグが含まれた文字列をBindする場合、HTMLタグを削除しない(デフォルト)

JSON(Array)	パラメータ名	必須	設定値	説明
<pre>[   {     "RESULT": "result1",     "TARGET": "href",     "FORMAT": "/[0]/[1]",     "ARGS": [       "id",       "category_code"     ],   },   {     "RESULT": "result1",     "TARGET": "text",     "ARG": "shop.name"   }, ]</pre>				「JSON(Object)」参照

Sample				
◆text(上書き)	<span data-bind-out=["RESULT":"result1","TARGET":"text","ARG":"shop.name"]>ショップ名</span>			
◆text(部分置換)	<span data-bind-out=["RESULT":"result1","TARGET":"text","ARG":"shop.name"]>[0]のおすすめ情報</span>			
◆text(定数)	<span data-bind-out=["TARGET":"text","ARG": "_CONST.title"]>NEWS</span>			
◆text(日付フォーマット)	<span data-bind-out=["RESULT":"result1","TARGET":"text","FORMAT": "DATE[yyyy/MM/dd]","ARG": "shop.publish_date"]>2013/12/03</span>			
◆text(数値フォーマット、カンマ区切り)	<span data-bind-out=["RESULT":"result1","TARGET":"text","FORMAT": "NUMBER[###0]","ARG": "shop.employees"]>123,456</span>			
◆a[href]	<a href="/../html/shop/list.html" data-bind-out=[["RESULT":"result1","TARGET":"href","FORMAT": "[ ORIGINAL]type/[0]/","ARG": "type.name"]]>Android</a>			
◆img[src]				

日付、数値フォーマット	
日付、数値フォーマットは、ARGが指定されている場合にのみ有効。(“FORMAT”+“ARGS”を指定した場合、“FORMAT”に日付、数値フォーマットは指定できない。)	
◆日付フォーマットパターン DATE[フォーマット] フォーマットに、[[補足]format]シートに記載されたパターンを記述する。 例) yyyy年MM月dd日 E曜日 yyyy/M/d	→ 2011年08月03日 火曜日 → 2011/8/3
◆数値フォーマットパターン NUMBER[フォーマット] フォーマットに、[[補足]format]シートに記載されたパターンを記述する。 例) ####.## 00,000.00	→ 123,456,789.57 → 01,234.50

data-case

指定の設定値から選択				
JSON(Object)	パラメータ名	必須	設定値	説明
<pre>{   "TYPE": "compare",   "OPE": "or",   "ARG": [     "companyList.company.name",     "eq",     "1"   ] }</pre>				
	TYPE	○	compare size	任意の値と比較する際に指定する 返却リストのサイズを比較する場合に指定する data-iteratorと併用する
	OPE		and or	data-caseを配列で指定した場合の条件演算子を指定する デフォルトは、「and」
	ARG	○	-	比較条件を指定する配列
	-	○	比較対象	compareの場合 DBの値を比較する場合:result識別子.テーブル名.カラム名を指定する リクエストパラメータを比較する場合:"REQ."パラメータ名を指定する 定数を比較する場合:"CONST."定数Keyを指定する(viewカテゴリのKeyのみ指定可能) セッションの値を比較する場合:"SES."Keyを指定する ページの値を比較する場合:result識別子."_PAGER".ページャー情報Key sizeの場合 data-loadまたは、data-funcの結果識別子を指定する
			result識別子._VALUE	Listの値を比較対象とする場合 result識別子._VALUE
			result識別子._STATUS.index	Listのstatus(0からのカウント)を比較対象とする場合
			result識別子._STATUS.count	Listのstatus(1からのカウント)を比較対象とする場合
			result識別子._STATUS.odd	Listのstatus(奇数行であるか)を比較対象とする場合
			result識別子._STATUS.even	Listのstatus(偶数行であるか)を比較対象とする場合
			result識別子._STATUS.first	Listのstatus(最初の行であるか)を比較対象とする場合
			result識別子._STATUS.last	Listのstatus(最後の行であるか)を比較対象とする場合
			result識別子._STATUS.modulus	Listのstatus(nで割った余り)を比較対象とする場合
	-	○	eq ne gt ge lt le	= <> > >= < <=
	-		任意の値	比較値

JSON(Array)	パラメータ名	必須	設定値	説明
<pre>[   {     "TYPE": "compare",     "ARG": [       "result1.shop.type",       "eq",       "1"     ]   },   {     "TYPE": "compare",     "ARG": [       "result1.shop.name",       "ne",       "null"     ]   },   ... ]</pre>				「JSON(Object)」参照

Sample

# data-erase

指定の設定値から選択

パラメータ	必須	設定値	説明
削除範囲指定	○	own	data-erasureが指定された要素およびの配下の要素を削除する
		child	data-erasureが指定された要素の配下の要素を削除する

data-url

JSON(Array)	パラメータ名	必須	設定値	説明
[ "/shop/detail/id/{0}", .... ]	-	○	リクエストURLパターン	コンテキスト以降のパスを指定する 可変のパラメータは、{0}とする。パラメータのKeyは、直前に指定されたパスとする。 左記の例では、パラメータのKeyは、idとなる。

Sample
<p>◆urlパターン単一指定、パラメータなし</p> <p>&lt;head data-url=["/shop/search/list/"]&gt;</p> <p>URL: http://xxxxxxx.com/shop/search/list/</p> <p>◆urlパターン複数指定、パラメータなし</p> <p>&lt;head data-url=["/shop/search/list/", "/shop/search/list/a/"]&gt;</p> <p>URL: http://xxxxxxx.com/shop/search/list/ または http://xxxxxxx.com/shop/search/list/a/</p> <p>◆urlパターン単一指定、パラメータあり</p> <p>&lt;head data-url=["/shop/detail/id/{0}/"]&gt;</p> <p>URL: http://xxxxxxx.com/shop/detail/id/47/</p> <p>◆urlパターン単一指定、パラメータ複数</p> <p>&lt;head data-url=["/shop/detail/id/{0}/type/{0}/"]&gt;</p> <p>URL: http://xxxxxxx.com/shop/detail/id/47/type/1/</p> <p>◆urlパターン複数指定、パラメータあり</p> <p>&lt;head data-url=["{ORIGINAL}id/{0}/", "{ORIGINAL}year/{0}/"]&gt;</p> <p>URL: http://xxxxxxx.com/shop/detail/id/47/ または URL: http://xxxxxxx.com/shop/detail/year/2003/</p> <p>※{ORIGINAL}に下記、htmlファイルパスのルール適用し、パラメータを付与した文字列をURLとする</p> <p>◆data-urlを指定しなかった場合のURLは、htmlファイルが配置されたフォルダ構成によって決定される。</p> <p>/[フォルダ名]-[サブフォルダ名]/[htmlファイル名]/</p> <p>・¥shop¥list.htmlの場合</p> <p>http://xxxxxxx.com/shop/list/</p> <p>・¥shop¥search¥list.htmlの場合</p> <p>http://xxxxxxx.com/shop-search/list/</p> <p>・¥shop¥search¥type¥list.htmlの場合</p> <p>http://xxxxxxx.com/shop-search-type/list/</p>



# data-auth

パラメータ	必須	設定値	説明
"true" 固定			

Sample
<pre>&lt;head data-auth='true'&gt; &lt;/head&gt;</pre>

data-image

JSON(Object)	パラメータ名	必須	設定値	説明
{"RESULT": "result1", "TARGET": "data-orignal", "ARG": "event.path",  "W": "100", "H": "150",}	RESULT		data-loadまたはdata-funcで指定したresult識別子	どのdata-load、data-funcの結果を出力するかを識別する為に指定する bindする内容がdata-load、data-funcの結果でない場合、指定しない
	TARGET		bind対象の属性名	未指定の場合は、srcにbindする。
	ARG	○	テーブル名.カラム名 または REQ.リクエストパラメータ名 または CONST.定数Key	DBの値をbindする場合、取得元テーブル名.カラム名を指定する リクエストパラメータをbindする場合: "_REQ."パラメータ名を指定する 定数をbindする場合: "_CONST."定数Keyを指定する(viewカテゴリのKeyのみ指定可能)
	W		100	アップロードした画像をサイズ変更する場合、横サイズをピクセルで指定する
	H		150	アップロードした画像をサイズ変更する場合、縦サイズをピクセルで指定する

Sample

# pager仕様

仕様
<div>◆PAGER仕様</div> <div>data-loadまたは、data-funcのselect系メソッドでPAGERオブジェクトを指定した場合、ページャ情報が返却される。</div> <div>参照方法:result識別子_PAGER</div> <div>参照可能な情報:</div> <div>RECORD_COUNT:取得した全件数</div> <div>PER_PAGE:1ページあたりの最大行数(=PERPAGE)</div> <div>RECORD_BEGIN_NO:表示開始レコード番号(OFFSET)</div> <div>RECORD_END_NO:表示終了レコード番号</div> <div>PAGE_NO:現在のページ番号</div> <div>PREV_PAGE_NO:前のページ番号。前ページが存在しない場合、0を返却。</div> <div>NEXT_PAGE_NO:次のページ番号。次ページが存在しない場合、0を返却。</div> <div>PAGING_BEGIN_NO:表示開始ページング番号</div> <div>PAGING_END_NO:表示終了ページング番号</div> <div>MAX_PAGE_NO:最終ページ番号</div> <div>PAGE_NO_LIST:ページング番号リスト</div> <div>◆サンプル</div> <div>前のページ <u>1</u> <u>2</u> <u>3</u> <u>4</u> <u>5</u> <u>6</u> <u>7</u> 次のページ</div> <div>・HTML</div> <div><pre>&lt;nav&gt; &lt;ul id="pager"&gt; &lt;li data-keep="true" data-case="{ 'TYPE': 'compare', 'ARG': ['result1._PAGER.PREV_PAGE_NO', 'ne', '0'] }"&gt;   &lt;a href="../../html/list.html" data-bind-out="{ 'RESULT': 'result1', 'TARGET': 'href', 'ARG': '_PAGER.PREV_PAGE_NO', 'FOR MAT': '{ _ORIGINAL }/page/{0}/' }"&gt;前のページ&lt;/a&gt; &lt;/li&gt; &lt;li data-iterator='result1.PAGE_NO_LIST'&gt;   &lt;a href="../../html/list.html" data-bind-out="{ { 'TARGET': 'text' }, { 'TARGET': 'href', 'ARG': '', 'FORMAT': '{ _ORIGINAL }/page/{0}/' } }"&gt;1&lt;/a&gt; &lt;/li&gt; &lt;li data-keep="true" data-case="{ 'TYPE': 'compare', 'ARG': ['result1._PAGER.NEXT_PAGE_NO', 'ne', '0'] }"&gt;   &lt;a href="../../html/list.html" data-bind-out="{ 'RESULT': 'result1', 'TARGET': 'href', 'ARG': '_PAGER.NEXT_PAGE_NO', 'FOR MAT': '{ _ORIGINAL }/page/{0}/' }"&gt;次のページ&lt;/a&gt; &lt;/li&gt; &lt;/ul&gt; &lt;/nav&gt;</pre></div>

## ◆日付フォーマット

文字	意味	例
yy	西暦年（2桁）	2012年→2012
yyyy	西暦年（4桁）	2012年→12
M	月	8月→8
MM	月（ゼロ埋め）	8月→08
D	年に対する日	1月9日→9
DDD	年に対する日（ゼロ埋め）	1月9日→009
d	月に対する日	3日→3
dd	月に対する日（ゼロ埋め）	3日→03
w	年に対する週	2011年8月30日→36（2011年の36週目）
W	月に対する週	2011年8月30日→5（2011年8月の5週
E	曜日	2011年8月30日→火
F	月に対する曜日番号	2011年8月30日（火） →5（2011年8月の5回目の火曜日）
a	午前午後	13:00→午後
h	時（12時間制）	13時→1
hh	時（12時間制ゼロ埋め）	13時→01
H	時（24時間制）	3時→3
HH	時（24時間制ゼロ埋め）	3時→03
m	分	3分→3
mm	分（ゼロ埋め）	3分→03
s	秒	3秒→3
ss	秒（ゼロ埋め）	3秒→03
S	ミリ秒	3ミリ秒→3
SSS	ミリ秒（ゼロ埋め）	3ミリ秒→003

## ◆数値フォーマット

文字	メモ
0	数値 1 桁を表す。その桁に数値が無い場合は"0"を表示する
#	数値 1 桁を表す。その桁に数値が無い場合はブランクとなる
.	小数点を表す。
,	カンマ区切りを表す。
-	マイナスを表す。
%	数値を100倍してパーセント表示にする。

# 認証情報

仕様
<div>・フォーム認証またはOAuth認証に成功すると、認証情報がセッションに格納される</div> <div>■ 認証情報へのアクセス</div> <div><div><div>_SES.LOGIN.USER.認証テーブルのカラム名</div><div>認証テーブル: application.propertiesのauth.table.nameで指定したテーブル</div></div></div>