

INTER-Mediator

Developer's Meet-UP#2

2012/7/31

KDDI Seminar Room

Agenda



開発報告と開発予定

ディスカッション

What's IM…は、今回もありません

イベント報告

INTER-Mediator Jelly #1

on June 4, 2012

at Shinjuku Travelers Coworking

Participated 11 people.





Monday, July 16, 2012
Splash 1-4

(A Gold Package ribbon is required to attend the FTS sessions)

7:30am - 8:00am	FTS: An Introduction to the FileMaker Training Series To... (No Fee Required) Cristoffer Ippolite
8:00am - 10:00am	FTS : Data Modeling Mark Richman
10:30am - 12:30pm	FTS: Working With Fields Rick Colcock
1:30pm - 3:30pm	FTS: Connectivity Alexei Folger
4:00pm - 6:00pm	CWP User Group Meeting Joel Shipiro

CWP User Group Meeting in FileMaker DevCon 2012 on July 16, 2012 at Miami, FL, USA



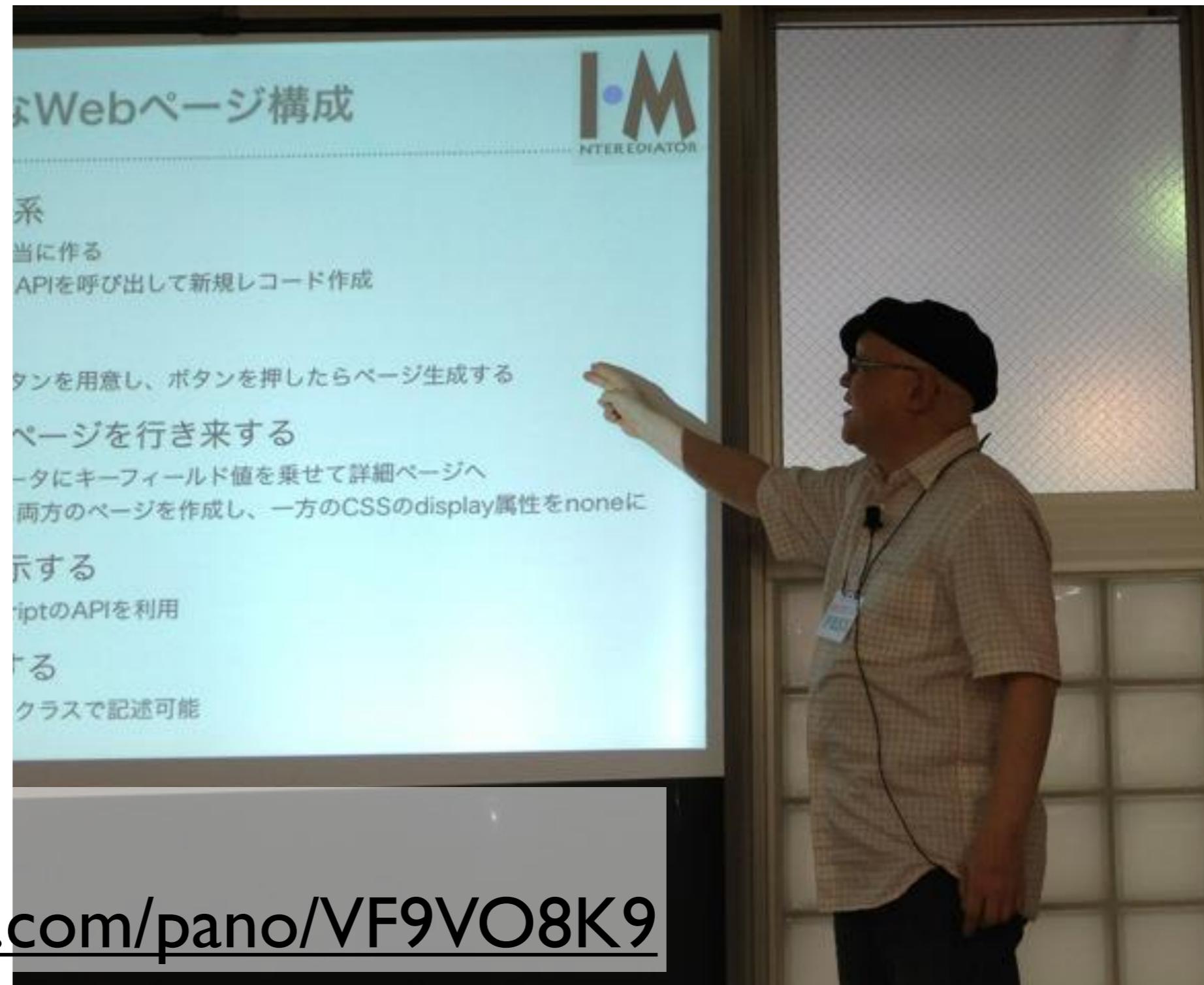


CWP User Group Meeting

Open Source Conference 2012.DB

on July 25, 2012

at Japan Electronics College



Panorama Photo
<http://www.pixeet.com/pano/VF9VO8K9>

INTER-Mediator Jelly #1

on July 27, 2012
at Shinjuku Travelers Coworking

Participated 6 people.



アップデート報告

これまでの経緯

Ver.1.0.10 (April 6, 2012)	FileMaker Server 12対応、DBのアカウントでの認証
Ver.2	バージョン番号を荒くした
Ver.2.1 (April 29, 2012)	バリデーション
Ver.2.3/2.4 (May 25 and 30, 2012)	プログラムの追加方法を改良
Ver.2.5 (June 5, 2012)	ターゲット指定に\$の指定を可能にした。元のデータの一部をデータベースの値で置き換える
Ver.2.6 (June 10, 2012)	ユーザによるパスワードの変更
Ver.2.7 (June 20, 2012)	PostgreSQL、SQLiteへの対応
Ver.2.8 (July 11, 2012)	認証付きでのメディア公開

バリデーション



include_MySQL.php - INTER-Mediator - [~/Documents/INTER-Mediator_develop/INTER-]

Project Demo2

change_log.txt Adapter_DBServer.js php include_MySQL.php

1: Project

INTER-Mediator (~/Documents/INTER-Mediator_develop/INTER-Mediator)

develop-im

- Auth_Support
- Demo-Script
- INTER-Mediator
- INTER-Mediator-UnitTest
- Sample_Auth
- Sample_chat
- Sample_Extensible
- Sample_forDebugging
- Sample_form
- Sample_invoice
 - include_FMS.php
 - include_MySQL.php
 - invoice_FMS.html
 - invoice_MySQL.html
- Sample_products
- Sample_products_auth
- Sample_search
- WebSite
- index.html
- sample.css

```
array(
    'name' => 'item',
    // 'table' => 'item',
    // 'view' => 'item_display',
    'key' => 'id',
    'relation' => array(
        array('foreign-key' => 'invoice_id', 'join-field' => 'id', 'type' => 'one-to-many'),
        'repeat-control' => 'insert delete',
        'default-values' => array(
            array('field' => 'product_id', 'value' => 1),
        ),
        'validation' => array(
            array(
                'field' => 'qty',
                'rule' => 'value>=0 && value<100',
                'message' => 'Quantity should be between 1..99.'
            ),
            array(
                'field' => 'unitprice',
                'rule' => 'value>=0 && value<10000',
                'message' => 'Unit price should be between 1.. 9999.'
            ),
        ),
        'post-repeater' => 'itemsExpanded',
    ),
),
```

サーバサイドプログラミング



definecontexts2.php - INTER-Mediator - [~/Documents/INTER-Mediator_develop/INTER-Mediator]

```
Project Demo2 change_log.txt Adapter_DBServer.js include_MySQL.php definecontexts2.php
```

1: Project

INTER-Mediator (~/Documents/INTER-Mediator)

- develop-im
 - Auth_Support
 - Demo-Script
 - INTER-Mediator
 - INTER-Mediator-UnitTest
 - Sample_Auth
 - Sample_chat
 - Sample_Extensible
 - aggregation1.html
 - aggregation2.html
 - DB_Extended_PDO.php
 - definecontexts1.php
 - definecontexts2.php
 - schema_data.txt
 - SumForCustomers.php
 - SumForItems.php
 - YearMonthGen.php
 - Sample_forDebugging
 - Sample_form
 - Sample_invoice
 - Sample_products
 - Sample_products_auth

IM_Entry(
 array(
 array(
 'name' => 'everymonth',
 'view' => 'item_master',
 'query' => array(array('field' => 'id', 'operator' => '=', 'value' => '1')),
 'records' => 1,
 'extending-class' => "YearMonthGen",
),
 array(
 'name' => 'summary1',
 'view' => 'saleslog',
 'relation' => array(
 array('foreign-key' => 'dt', 'operator' => '>=', 'join-field' => 'startdt'),
 array('foreign-key' => 'dt', 'operator' => '<', 'join-field' => 'enddt'),
),
 'extending-class' => "SumForItems",
),
 array(
 'name' => 'summary2',
 'view' => 'saleslog',
 'relation' => array(
 array('foreign-key' => 'dt', 'operator' => '>=', 'join-field' => 'startdt'),
 array('foreign-key' => 'dt', 'operator' => '<', 'join-field' => 'enddt'),
),
 'extending-class' => "SumForCustomers",
),
),
);

サーバサイドプログラミング

A screenshot of the PhpStorm IDE interface. The left sidebar shows a project structure with a 'Project' view containing several folders and files under 'INTER-Mediator'. One file, 'SumForItems.php', is currently selected and highlighted with a grey bar at the bottom of the list. The main editor window displays the PHP code for this file. The code implements a class 'SumForItems' that extends 'Extending_Interface_AfterGet'. It contains a method 'doAfterGetFromDB' which iterates through a database result set, sums up item prices, sorts them, and then formats the results into an array with 'itemname' and 'totalprice' keys. A note in the code indicates it was created by JetBrains PhpStorm.

```
<?php
/*
 * Created by JetBrains PhpStorm. ...
 */
class SumForItems implements Extending_Interface_AfterGet
{
    function doAfterGetFromDB($dataSourceName, $result)
    {
        $sum = array();
        foreach ($result as $record) {
            $sum[$record["item"]] += $record["total"];
        }
        arsort($sum);
        $result = array();
        $counter = 10;
        foreach ($sum as $product => $totalprice) {
            $result[] = array(
                "itemname"=>$product,
                "totalprice"=>number_format($totalprice)
            );
            $counter--;
            if ( $counter <= 0 ) {
                break;
            }
        }
        $this->resultCode = count($result);
        return $result;
    }
}
```

クライアントサイドプログラミング



The screenshot shows an IDE interface with the following details:

- Title Bar:** include_SQL.php - INTER-Mediator - [~/Documents/INTER-Mediator_develop...]
- Toolbar:** Standard file operations (New, Open, Save, Print, Find, Replace, etc.) and project-related icons.
- Project Explorer (Left):** Shows the project structure under "1: Project".
 - INTER-Mediator (~/Documents/INTER-Mediator)
 - develop-im
 - Auth_Support
 - Demo-Script
 - INTER-Mediator
 - INTER-Mediator-UnitTest
 - Sample_Auth
 - Sample_chat
 - Sample_Extensible
 - Sample_forDebugging
 - Sample_form
 - Sample_invoice
 - php include_FMS.php
 - php include_SQL.php
 - invoice_FMS.html
 - invoice_SQL.html
 - Sample_products
 - Sample_products_auth
 - Sample_search
 - WebSite
 - index.html
 - sample.css
 - sample_memo_html5.html
 - dist-docs
 - .gitignore
 - buildup.sh
 - README.txt
 - External Libraries
- Code Editor (Right):** Displays PHP code for "include_SQL.php".

```
require_once('../INTER-Mediator/INTER-Mediator.php');
$TableDefinitions = array(
    array(
        'name' => 'invoice',
        'records' => '1',
        'paging' => true,
        'key' => 'id',
        'query' => array(
            //      array('field' => 'issued', 'value' => '2012-01-01'),
        ),
        'sort' => array(
            array('field' => 'id', 'direction' => 'ASC'),
        ),
        'repeat-control' => 'insert delete',
        'post-enclosure' => 'invoiceExpanded',
    ),
    array(
        'name' => 'item',
        // 'table' => 'item',
        // 'view' => 'item_display',
        'key' => 'id',
        'relation' => array(
            array('foreign-key' => 'invoice_id', 'join-field' => 'invoice'),
        ),
        'repeat-control' => 'insert delete',
        'default-values' => array(
            array('field' => 'product_id', 'value' => 1),
        ),
        'validation' => array(
            array(
                'field' => 'qty',
                'rule' => 'value>=0 && value<100',
                'message' => 'Quantity should be between 1..99.'
            ),
            array(
                'field' => 'unitprice',
                'rule' => 'value>=0 && value<10000',
                'message' => 'Unit price should be between 1..9999'
            ),
        ),
        'post-repeater' => 'itemsExpanded',
    )
);
```

Two red circles highlight specific sections of the code in the editor:

- A red circle highlights the section of code that defines the 'repeat-control' for the 'invoice' table, specifically the line: `'repeat-control' => 'insert delete',`
- A red circle highlights the section of code that defines the 'repeat-control' for the 'item' table, specifically the line: `'repeat-control' => 'insert delete',`

クライアントサイドプログラミング



invoice_MySQL.html - INTER-Mediator - [~/Documents/INTER-Mediator_develop/INTER-Mediator]

Project: Demo2

File: definecontexts2.php

change_log.txt Adapter_DBServer.js SumForItems.php invoice_MySQL.html

Sample_invoice

include_FMS.php
include_MySQL.php
invoice_FMS.html
invoice_MySQL.html

INTERMediatorOnPage.invoiceExpanded = function (target) {
 calcTotal(target);
};

INTERMediatorOnPage.itemsExpanded = function (repeaters) {
 modLine(repeaters[0]);
};

function modLine(target) {
 if (!target) {
 return;
 }
 var qtyId = INTERMediatorOnPage.getNodeIdFromIMDefinition("item@qty", target);
 var unitPriceId = INTERMediatorOnPage.getNodeIdFromIMDefinition("item@unitprice", target);
 var productPriceId = INTERMediatorOnPage.getNodeIdFromIMDefinition("product@unitprice", target);
 var amountId = INTERMediatorOnPage.getNodeIdFromIMDefinition("item@amount", target);
 if (qtyId && unitPriceId && productPriceId && amountId) {
 var unitPrice = new Number(document.getElementById(unitPriceId).value > 0 ?
 document.getElementById(unitPriceId).value :
 document.getElementById(productPriceId).innerHTML);
 var amountPrice = unitPrice * document.getElementById(qtyId).value;
 document.getElementById(amountId).innerHTML = INTERMediatorLib.numberFormat(amountPrice);
 }
}

function calcTotal(target) {
 var nodes = INTERMediatorOnPage.getNodeIdsFromIMDefinition('item@amount', target);
 if (nodes) {
 var s = 0;
 for (var i = 0; i < nodes.length; i++) {
 s += INTERMediatorLib.toNumber(document.getElementById(nodes[i]).innerHTML);
 }
 if (document.getElementById('total')) {
 document.getElementById('total').innerHTML = INTERMediatorLib.numberFormat(s, 2);
 }
 }
}

</script>

パスワード変更とユーザ管理



The screenshot shows a web browser window titled "INTER-Mediator - Sample" displaying the URL "localhost/im/Auth_Support/MySQL_accountmanager.html". The page content is a "User Management System". A sidebar on the left lists "User Accounts" and "Group Accounts". The main area contains a message: "Set the new password and click "Set" button on each user." Below this is a login form with fields for "User:" and "Password:", and a "Log In" button. Overlaid on the page is a modal dialog box with a dark gray background. It contains a "User:" input field, a "Password:" input field, and a "Log In" button. At the bottom of the dialog is a "New Password:" input field and a "Change Password" button.

パスワード変更とユーザ管理



メディアプロキシ

動作要件

- あるPNGファイルにアクセスした場合
- 定義ファイルにリダイレクトする
- パラメータとして、?media=ファイル名

コンテキスト指定

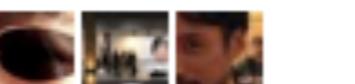
- 'authentication'=> array('media-handling' => true),

オプション指定

- 'media-root-dir'=>'/Library/WebServer/Documents/im/
Sample_products/images',

機能追加アンケート結果（前回）

先々の開発計画について、アンケートにしてみました。現在、認証をがんばっているところで、さらに次は、アスペクト指向によるコンテキストの分離を行います。その先、どの機能の組み込みに移るか、みなさんの要望を聞きたいと思います。複数選択もOKですので、よろしくお願ひします。 ×

- バリデーション（値のチェックと警告） 
 - COMMIT/ROLLBACKを使った書き込みデータの確定 
 - ユーザによるパスワードの変更 
 - 検索結果のローカルキャッシュ 
 - FileMakerスタイルの検索機能 
 - クラウドデータベースのサポート 
 - ユーザやグループを管理するためのページ 
 - 画像やファイルのハンドリング 
 - ローカルデータベースサポート 
 - FileMakerの「チェックボックスセット」的な動作をする選択肢 
- + オプションを追加...

機能追加アンケート結果！

7/31のMeet-up #2に向けて、近々のフィーチャリクエストを募集したいと思います。どんな機能を組み込んでほしいか、アンケートの回答をよろしくお願ひします。

- HTML5のローカルデータベース 
- 画像などのファイルのアップロード 
- クラウドのサポート 
- サーバサイドスクリプトのコンテキスト拡張 
- Oracle, SQL Serverサポート
- 有償のコースの充実
- HTMLエディタとの連携
- 英語のマニュアル
- BLOB(Binary Large Object)のサポート
- HTML5特有のユーザインタフェース
(Datalistなど)
- FileMakerのチェックボックスセットのようなマルチバリューフィールド
- データベースアクセス結果のローカルキャッシュ
- エラーのアラート表示
- サーバサイドスクリプトのO/Rマッピング対応

今後の開発に関する予定

画像やメディアのハンドリング

- BLOB（オブジェクト）への対応

データベース対応

- ローカルデータベース、クラウド
- ローカルキャッシュ

サーバ再度の高度化

- あるコンテキストの処理中に別のコンテキストの処理ができる

セキュリティ対策

テスト

- きちんとテストケースを作ってバグを減らす

メディアハンドリング

		ファイル	データベース
表示	認証なし	フィールドにファイル名を記録し、imgのsrcに埋め込む	プロキシ?
	認証あり	メディアプロキシ	プロキシ?
入力	認証なし	専用CGI?	専用CGI?
	認証あり	専用CGI?	専用CGI?

セキュリティ対策

フィールドの更新をブロック

- フィールド名が分かれれば、書き直されることになる
- コンテキストごとに更新できないフィールドを指定
- 主キーとアクセス権設定に使うフィールドもブロック

表示しないフラグ～の場合の検討

- フラグが立っていないレコードは見えないようにした
- 権限があればフラグに関係なく見えるモデル
- ブロックするだけでなく、0→1は許容するが、他の書き込みを許容しないような設定も必要かも？

IMのソースをベースに
開発する

INTER-Mediatorのソース



GitHubを使って公開

専用アプリケーションを使えばかなり楽になる

ハンズオン



アカウントの作成（当然、無料）

アプリケーションをダウンロード

レポジトリのダウンロード

変更結果のアップロード

GitHubアプリケーション



Repositories > msyk/INTER-Mediator > Commit >

msyk 2012年5月25日 1:54
Ver.2.3

Branch in Sync

[View on GitHub](#)

[Revert commit](#) [Rollback to this commit](#) [Collapse all files](#)

buildup.sh

OLD	NEW
5	5
6	6 YUICOMP="../yuicompressor-2.4.7.jar"
7	7
8	-version="2.2"
8	+version="2.3"
9	9
10	10 dt=`date "+%Y-%m-%d"`
11	11 versionInFilename=`echo "\${version}" tr '.' '_'`;

demo-script/demo1.html

OLD	NEW
1	1 <html>
2	2 <head>
3	- <link rel="stylesheet" type="text/css" href="../develop-im/sample.css"/>
3	+ <link rel="stylesheet" type="text/css" href="../develop-im/sample.css" />
4	<script type="text/javascript" src="demo1.php"></script>
5	</head>
6	<body>
7	<table border="1">
8	- <tr>
10	- <td>id</td>
11	- <td><input type="text" class="IM[person@id]" /></td>
12	- </tr>
13	- <tr>
14	- <td>name</td>
15	- <td><input type="text" class="IM[person@name]" value="" /></td>
16	- </tr>
17	- <tr>
18	- <td>address</td>
19	- <td><input type="text" class="IM[person@address]" value="" /></td>

↑ master

ディスカッション