

【B6】 PHPテクニカルセッション

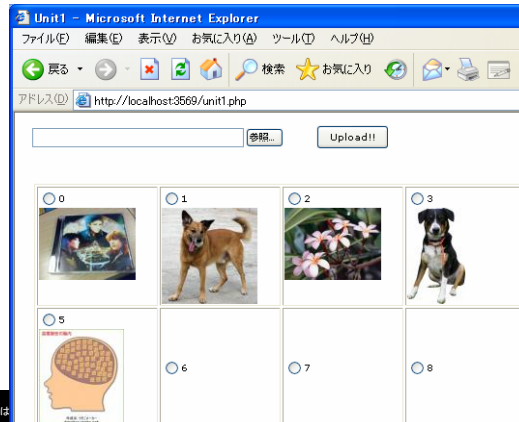
「Delphi for PHPによる フォトギャラリーの作成と運用」

CodeGear
エヴァンジェリスト
高橋智宏

アジェンダ

- 作成するWebアプリケーションの概要
- ファイルのアップロードを行うには？
- サムネイル(縮小)画像を生成するには？
- カスタムコンポーネントを作成するには？
- 作成したWebアプリケーションを配布するには？

作成するWebアプリケーションの概要



必要な機能

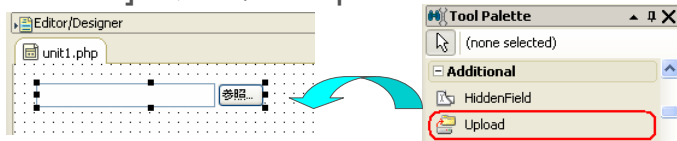
- JPEGファイルをアップロードする
- サムネイルを生成する
- サムネイルをテーブル形式でリスト表示する
- サムネイルをクリックすると、オリジナル画像が表示される

ファイルのアップロードを行うには？

Uploadコンポーネント
アップロードされたファイルを取得する
php.ini の設定項目

Uploadコンポーネント

- [Additional]カテゴリの Upload



- ファイルのフルパス
 - 一時的なもの。ページの処理後は自動的に消える
 - `$this->Upload1->FileTmpName` プロパティ
- 別ファイルとして残すには
 - `move_uploaded_file(移動元パス, 移動先パス);`

php.ini の設定項目

- file_uploads = On
- ;upload_tmp_dir =
 - 指定しない場合はシステムデフォルト
- upload_max_filesize = 2M
- post_max_size = 8M
- memory_limit = 32M

memory_limit > post_max_size > upload_max_filesize

サムネイル(縮小)画像を生成するには？

アップロードされたJPEG画像の読み込み
サムネイル用(最大100x100)の画像リソース生成
サンプリングによる縮小コピー
JPEGリソースをファイルへの書き出す

アップロードされたJPEG画像の読み込み

- \$imgsize = getimagesize(ファイル名);
 - \$imgsize[0]は幅, \$imgsize[1]は高さ
- \$imgdata = imagecreatefromjpeg(ファイル名);
 - \$imgdataはresource型

- 元の画像サイズが、サムネイルのサイズ以下(縮小が不要)の場合は、サムネイルの生成処理は行わない

```
function Button1Click($sender, $params)
{
    ...
    $imgfile = $this->GetImageFileName($imgindex);
    move_uploaded_file($this->Upload1->FileTmpName, $imgfile);
    $imgsize = getimagesize($imgfile);
    $imgdata = imagecreatefromjpeg($imgfile);

    $thumbfile = $this->GetThumbFileName($imgindex);
    $thumb_w = 100;
    $thumb_h = 100;
    if( $imgsize[0] <= $thumb_w || $imgsize[1] <= $thumb_h )
    {
        imagejpeg($imgdata, $thumbfile);
        return;
    }
    ...
}
```

サムネイル画像(最大100x100)の生成

- \$thumbdata = imagecreatetruecolor(幅, 高さ);
 - 適切な幅と高さを、元画像の幅と高さから算出
 - \$thumbdataはresource型
- imagecopyresampled(コピー先resource, コピー元resource, 0, 0, 0, 0, コピー先幅, コピー先高さ, コピー元幅, コピー元高さ);
 - 今回はコピー元からコピー先へと縮小を行う
- imagejpeg(画像resource, 出力先ファイル名);
 - JPEG形式でサムネイルを生成

```
function Button1Click($sender, $params)
{
    ...
    $thumb_w = 100;
    $thumb_h = 100;
    ...
    ...
    if( $imgsize[0] >= $imgsize[1] ) // w >= h
        $thumb_h = $thumb_h * ($imgsize[1] / $imgsize[0]);
    else
        $thumb_w = $thumb_w * ($imgsize[0] / $imgsize[1]);
    $thumbdata = imagecreatetruecolor($thumb_w, $thumb_h);
    imagecopyresampled($thumbdata, $imgdata, 0, 0, 0, 0,
        $thumb_w, $thumb_h, $imgsize[0], $imgsize[1]);
    imagejpeg($thumbdata, $thumbfile);
}
```

} 整数の除算と乗算
だけど、大丈夫？

カスタムコンポーネントを作成するには？

<table>...</table>コンポーネントが無い!!!!!!!!!!!!!!

カスタムコンポーネントの作成

プロパティの作成

イベントの作成

<table>...</table>コンポーネント

- VCL for PHP には、<table>を生成するコンポーネントが無い!!
 - サードパーティのコンポーネントにはあるが...
- カスタムコンポーネントを設計してみよう!!
 - 親クラスは、VCL for PHP の「CustomControl」
 - 列数プロパティ, 行数プロパティ
 - セル(<td>...</td>)の中身を生成するイベント
 - function dumpContents() { ... } のオーバーライド

カスタムコンポーネントの作成

- UTF-8は未サポート **重要!!**
- パッケージとコンポーネントの新規作成

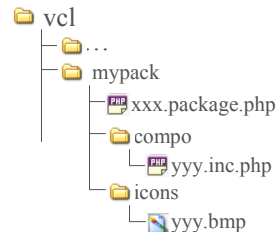
The screenshot shows two windows from the Delphi IDE. On the left is the 'Options' dialog, with 'Editor Options' selected in the tree view. A red box highlights the checkbox 'Use UTF-8 to create new units and forms', which is currently unchecked. An arrow points from this checkbox to the text below. On the right is the 'Component Wizard' dialog, with 'Component creation' selected. It shows fields for 'Ancestor class' (CustomControl), 'Classname' (KenTable), and 'Palette Page' (Ken). A red box highlights the 'Create package' checkbox, which is checked. At the bottom of the wizard are 'OK', 'Cancel', and 'Help' buttons.

コンポーネント作成時だけ
一時的にOFF

Copyright ©2008 CodeGear. All Rights Reserved.本文書の一部または全部の転載を禁止します。

カスタムコンポーネントの作成(続き...)

- CustomControl は、ビジュアル系コンポーネントのベースとなるクラス
- パッケージのファイル名
 - xxxx.package.php
- コンポーネントのファイル名
 - yyy.inc.php
- パッケージおよびコンポーネントは、必ずVCLのサブディレクトリに配置 **重要!!**
 - 例:C:\DelphiPHP¥1.0¥vcl¥mypack



パッケージファイル

```
<?php
require_once("vcl/vcl.inc.php");
use_unit("designide.inc.php");
setPackageTitle("Ken Components");
setIconPath("./icons");
registerComponents("Ken", array("KenTable"), "ken/compo/kentable.inc.php");
?>
```

パッケージの説明文

ツールパレットアイコンのフォルダ名

ツールパレットのカテゴリ名

コンポーネントのパス(vcl/は省く)

コンポーネントのクラス名

コンポーネントの基本的な書式

```
<?php
...
class クラス名 extends 親クラス {
    function __construct($owner = null) {
        parent::__construct($owner);
    }
    } } コンストラクタ

    function dumpContents() {
        ...
    } } HTMLデータを出力するため
    メソッドをオーバーライド

    protected $_xxx=デフォルト値;
    function getプロパティ名() { return $this->_xxx; }
    function setプロパティ名($value) { $this->_xxx=$value; }
    function defaultプロパティ名() { return デフォルト値; }
    } } プロパティ

    protected $_yyy=null;
    function get0nイベント名() { return $this->_yyy; }
    function set0nイベント名($value) { $this->_yyy=$value; }
    function default0nイベント名() { return null; }
    } } イベント
}
?>
```

<table>コンポーネントの実装

```

<?php
require_once("vcl/vcl.inc.php");
use_unit("stdctrls.inc.php");

class KenTable extends CustomControl {
    ...
    protected $_col=5;
    function getCol() { return $this->_col; }
    function setCol($value) { $this->_col=$value; }
    function defaultCol() { return 5; }
    protected $_row=5;
    function getRow() { return $this->_row; }
    function setRow($value) { $this->_row=$value; }
    function defaultRow() { return 5; }

    protected $_ondrawcell=null;
    function getOnDrawCell() { return $this->_ondrawcell; }
    function setOnDrawCell($value) { $this->_ondrawcell=$value; }
    function defaultOnDrawCell() { return null; }
}
    
```

デフォルトは5列x5行のテーブル

OnDrawCellイベント

Copyright ?>

<table>コンポーネントの実装(続き...)

```

...
function dumpContents() {
    echo "<table id=\"%${this->_name}_table%\" border=\"%1%\">\n";
    for($r = 0; $r < $this->_row; $r++) {
        echo "<tr>\n";
        for($c = 0; $c < $this->_col; $c++) {
            echo "<td>\n";
            if( ($this->ControlState & csDesigning) != csDesigning &&
                $this->_ondrawcell != null ) {
                // index = 0..N-1
                $this->callEvent('ondrawcell', array("index" => ($r*$this->_row)+$c));
            }
            else {
                echo "TestData\n"; // デザイン時のダミー表示
            }
            echo "</td>\n";
        }
        echo "</tr>\n";
    }
    echo "</table>\n";
}
    
```

実行時か判定
イベントハンドラのnullチェック

イベントハンドラ(_ondrawcell)を起動
パラメータはハッシュテーブル

Copyright

パッケージのインストールと利用

The screenshot illustrates the process of installing and using a component in the Delphi IDE. On the left, the 'Components' palette shows 'KenTable' under 'Installed Components'. Below it, the 'Packages' list includes 'Ken Components'. In the center, the 'Unit1: Unit1' component palette shows 'KenTable1: KenTable' selected. On the right, the 'unit1.php' design view shows a table with 5 columns and 5 rows of 'TestData'. The 'Object Inspector' shows the 'KenTable1' component with its 'OnDrawCell' event set to 'KenTableDrawCe'. The 'Code Explorer' shows the 'OnDrawCell' event handler code.

```

function KenTable1DrawCell($sender, $params) {
    $rb = new RadioButton();
    $rb->Name = 'RadioButtonX' . $params["index"];
    $rb->Caption = $params["index"];
    $rb->Group = 'imgindex';
    $rb->dumpContents();

    $imgfile = $this->GetImageFileName($params["index"]);
    $thumbfile = $this->GetThumbFileName($params["index"]);
    if( is_file($thumbfile) ) {
        $image = new Image();
        $image->Name = 'ImageX' . $params["index"];
        $image->Width = 100; // for <div> element
        $image->Height = 100; // for <div> element
        $image->ImageSource = $thumbfile;
        $image->Link = $imgfile;
        $image->LinkTarget = '_blank';
        $image->dumpContents();
    }
}
    
```

Copyright ©2008 CodeGear. All Rights Reserved. 本文書の一部または全部の転載を禁止

イベントハンドラの実装 - <td>の中身を出力

```

function KenTable1DrawCell($sender, $params) {
    $rb = new RadioButton();
    $rb->Name = 'RadioButtonX' . $params["index"];
    $rb->Caption = $params["index"];
    $rb->Group = 'imgindex';
    $rb->dumpContents();

    $imgfile = $this->GetImageFileName($params["index"]);
    $thumbfile = $this->GetThumbFileName($params["index"]);
    if( is_file($thumbfile) ) {
        $image = new Image();
        $image->Name = 'ImageX' . $params["index"];
        $image->Width = 100; // for <div> element
        $image->Height = 100; // for <div> element
        $image->ImageSource = $thumbfile;
        $image->Link = $imgfile;
        $image->LinkTarget = '_blank';
        $image->dumpContents();
    }
}
    
```

Tips

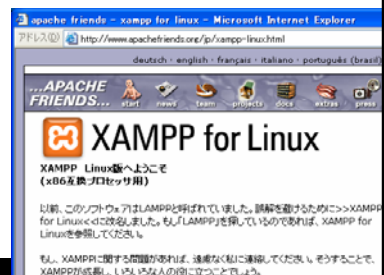
コンポーネントを動的に一時的に生成して
dumpContentsメソッドを呼び出して
HTMLを出力させる

作成したWebアプリケーションを配布するには？

LinuxにXAMPPをインストール
php.iniの確認とXAMPPの起動
アプリケーションを転送
VCL for PHP 1.2 とカスタムコンポーネントを転送

LinuxにXAMPPをインストール

- XAMPP for Linux 1.6.5a
 - <http://www.apachefriends.org/download.php?xampp-linux-1.6.5a.tar.gz> をダウンロード
 - `sudo tar xvfz xampp-linux-1.6.5a.tar.gz -C /opt`
- Windows向けには...
 - XAMPP for Windows
 - **WampServer 2.0**
 - <http://www.wampserver.com/en/>



php.iniの確認とXAMPPの起動

- /opt/lampp/etc/php.ini を編集
- 起動: `sudo /opt/lampp/lampp start`
- 終了: `sudo /opt/lampp/lampp stop`



Copyright ©2008 CodeGear. All Rights Reserved.本文書の一部または全部の転載を禁止します

ファイルを転送

- *.php と *.xml.php をFTPで転送
- カスタムコンポーネントを含め、vcl/ 以下すべてのファイルをFTPで転送
- 注意点!!
 - vcl/ はアプリケーションのサブフォルダに配置する
[vcl-bin サブディレクトリは、どこにありますか ?]
<http://support.codegear.com/article/37402>
 - classes.inc.php を更新
[Delphi for PHP Update 2 での EPropertyNotFound エラーについて]
<http://support.codegear.com/article/36967>