***РАБОТА С КАРТИНКАМИ***

require\_once("util/ImageUtil.class.php");

ImageUtil::uploadImage($image['tmp\_name'], "uploaded/news/", "{$id\_new}\_tmp.{$image\_type}");

$tmp\_img = new ImageUtil("uploaded/news/{$id\_new}\_tmp.{$image\_type}");

$tmp\_img->ResizeFromRaf(200, 120, "uploaded/news/{$id\_new}\_sm.{$image\_type}");

FileSystem::deleteFile("uploaded/news/{$id\_new}\_tmp.{$image\_type}");

***РАБОТА С ПОЧТОЙ***

require\_once('util/MailUtil.class.php');

$mail = new MailUtil();

$mail->setTo(ADMIN\_EMAIL);

$mail->setSubject('Обратная связь / '.$\_SERVER['HTTP\_HOST']);

$mail->setFrom("{$\_SERVER['HTTP\_HOST']} <".ADMIN\_EMAIL.">");

$tdata = array(

'fio' => $this->formData['fio'],

'message' => $this->formData['message'],

'email' => $this->formData['email'],

'phone' => $this->formData['phone'],

);

$mail->setEmailTextTemplate('mail/feedback.tpl', $tdata);

if (!$mail->doSend()){

$this->template->assign('errors',array("Не возможно отправить Ваше сообщение. Попробуйте позже."));

}

else{

$this->template->assign('notes',array('Ваше сообщение успешно отправлено!'));

}

***РАБОТА С XAJAX***

function doBeforeOutput(){

// XAJAX

$this->registerThis('aaa');

$this->processRequest();

}

function aaa(){

$xajax = new xajaxResponse();

$xajax->assign('map', 'innerHTML', "str\_text" );

//$this->template->assign('data', $data);

//$xajax ->assign('id\_name', 'innerHTML', $this->template->fetch('ajax/name.tpl'));

return $xajax;

}

*В id равным 'map' выводиться значение "str\_text"*

***РАБОТА С JOIN***

//достаем все категории в которых есть текущий магаз

$query = $conn->newStatement("

SELECT sc.\*, c.\*

FROM shop\_category sc

INNER JOIN category c ON sc.id\_category=c.id

WHERE sc.id\_shop=:id\_shop:

ORDER BY c.id ASC, sc.id\_category ASC");

$query->setInteger("id\_shop", $id\_shop);

$data\_category\_in\_cur\_shop = $query->getAllRecords();

***РАБОТА С PAGER***

define('COUNT\_PAGE', 10);

require\_once 'module/common/PagerFactory.class.php';

$pager = new PagerFactory();

$sql = "SELECT \* FROM faqs WHERE active=1 ORDER BY id ASC";

$fromWhereCnt = "faqs WHERE active=1";

$href = "/faqs/";

$pagerString = $pager->getPagerString($page, $sql, $fromWhereCnt, $href);

$data = $pager->getPageData();

$this->template->assign('pager\_string', $pagerString);

$this->template->assign('data\_faqs', $data);

**РАБОТА с ФОРМОЙ**

<select id="arch\_year" onchange="xajax\_UpdateCalendar(document.getElementById('arch\_month').value, this.value)">

<form id="zakazForm" action="" method="post">

<a href="#send" onclick="xajax\_saveZakaz(xajax.getFormValues('zakazForm'));return false;"><img src="/js/overlay/btn\_send.png"></a>

</form>

xajax\_SendDeposit(xajax.getFormValues('forma\_deposit'));

много форм на странице

<form id="forma\_edit\_{$cur.id}" action="" method="post" onsubmit="xajax\_EditCat(xajax.getFormValues('forma\_edit\_{$cur.id}')); return false;">

<input type="text" name="editcat" value="" /><input type="hidden" name="hid\_id" value="{$cur.id}" /><input type="submit" value="OK" />

</form>

//ajax // изменение названия категории

public function EditCat($forma){

$objResponse = new xajaxResponse();

if(strlen($forma['editcat'])>2){

$query = $this->conn->newStatement("UPDATE category SET name=:name: WHERE id=:id:");

$query->setText('name', $forma['editcat']);

$query->setInteger('id', $forma['hid\_id']);

$query->execute();

$objResponse->redirect('/admin/netlab\_move/');

}

return $objResponse;

}

<form action="" method="post" enctype="multipart/form-data" onsubmit="return SendComment(this);">

<button class="btn"><span>отправить</span></button>

</form>

// добавляем коммент в базу. по натию на кнопку формы ОТПРАВИТь

function SendComment(formData) {

if (!formData.text.value) errors[counter++] = 'Не заполнено поле: Сообщение';

}

Упаковать все поля в крипте для передачи в аякс.

<form onsubmit="checkBeforeSubmitAddForm(this);">

</form>

function checkBeforeSubmitAddForm(formData) {

alert($(formData).serialize());

}

**AJAX на JQuery**

$.ajax({

url: "/ajax\_check\_email/",

async: false,

data: 'email='+email,

//data: 'id='+id\_cart+'&value='+edited,

dataType: 'text',

success: function(data){

if(data == 'OK'){ //

}

else {

}

},

complete: function(){}

});

'/ajax\_check\_email/'=> array('moduleName' => 'ajax/CheckEmailPage', 'moduleType' => 'module'),

<?php

class CheckEmailPage extends AbstractPageModule {

function doContent(){

$email = $this->request->getValue('email');

$query = $this->conn->newStatement("SELECT \* FROM user WHERE email=:email:");

$query->setVarChar('email', $email);

$data\_user = $query->getFirstRecord();

if($data\_user){

echo 'ERR';

}

else{

echo 'OK';

}

}

}

?>

--------ВТОРОЙ ВАРИАНТ С JSON----------------------

----В скрипте

dataType: "json",

----В модуле

echo json\_encode(array('sum\_confirm'=>$sum\_confirm));

----В скрипте

success: function(rez){

jQuery.parseJSON(rez);

alert(rez['sum\_confirm']);

},