

## Zend\_Controller

```
class IndexController extends Zend_Controller_Action {
    public function init() { // will be called for each action

        $this->view->doctype('XHTML1_TRANSITIONAL');
        $this->view->headMeta()->appendName('robots', 'index, follow');

    }

    public function indexAction() {
        $this->view->headMeta()->appendName('description', 'My description
for this action');
        $this->view->headMeta()-
>appendName('keywords', 'keywords, for, this, action');

        // append inline styles to that action
        $this->view->headStyle()->appendStyle('p { color: #AAA; }');

        // append javascript files and an inline script
        $this->view->appendFile()->('/url/to/file.js');
        $this->view->appendScript('alert("Page loaded!");');
    }

    public function displayFirstnameAction() {

        // fetch user id from GET or POST
        $uid = $this->getRequest()->getParam('uid');

        // fetch firstname from database
        $userTable = new User();
        $this->view->firstname = $userTable->getFirstname($uid);

    }

    public function activateUser() {

        $this->view->activationForm = $this->_getActivationForm();

        // check if the request is POST (so the form is submitted)
        if ($this->getRequest()->isPost()) {

            // validate the form values and assign error messages (if not valid)
            if ($this->view->activationForm->isValid($_POST)) {

                // get the filtered values of the activation form
                $values = $this->view->activationForm->getValues();

                $userTable = new User();
                $userTable->activate($values['uid']);

                // display another view with success message
                $this->render('activationComplete');
            }
            else {
                $this->view->activationForm->setDescription('Your account could
not be activate. Please correct the form values.');
```

## Zend\_Registry

```
// get the sitewide database adapter
$db = Zend_Registry::get('db');

// get the current language (if any is set)
$lang = Zend_Registry::get('Zend_Translate')->getLocale();

// get a configuration value
$mail = Zend_Registry::get('config')->mail->webmaster;
```

## Zend\_Db/Zend\_Db\_Table

```
class user extends Zend_Db_Table_Abstract {

    // name of the database table
    protected $_name = 'user';

    // primary key of this table
    protected $_primary = 'uid';

    // array with table names that contain the uid as a foreign key
    protected $_dependentTables = array('Book');

    public function getFirstname($uid) {
        $select = $this->select()->where('uid = ?', $uid);

        // return a single value
        return $this->getAdapter()->fetchOne($select);
    }

    public function getByEmail($email) {

        $select = $this->select()->where('email = ?', $email);

        // we only return one row here
        return $this->fetchRow($select);
    }

    public function getPairs() {

        // fetch all users from database to build a drop down
        return $this->getAdapter()->fetchPairs('SELECT uid, name FROM user
WHERE type="student" ORDER BY lastname');
```

## Zend\_View/Zend\_Layout

```
// Layout deaktivieren
$this->getHelper('layout')->disableLayout();

// View deaktivieren
$this->getHelper('ViewRenderer')->setNoRenderer();

/* Redirection
With $this->_forward('/foo/bar/') you transfer the execution of the current
request to a different action and the URL in the browser doesn't change.

With $this->_redirect('/foo/bar/') you actually make another round trip
from the server to the client and the URL in the browser will change.

Therefore, using $this->_forward() will be a less resource intensive call,
but sometimes you may actually need to redirect to a new location e.g. in
an access denied situation. */
```