

Kontaktformular

- [ACHTUNG!](#)
- [Grundsätzliche Benutzung](#)
 - [PHP-Variante](#)
 - [PHTML-Variante](#)
 - [HTML-Variante](#)
 - [TYPO3/TV Variante](#)
- [Weitere Konfigurationsmöglichkeiten](#)

ACHTUNG!

Das [Formmail.pl](#) ist für immer abgeschaltet worden! Bei HTML/PHP-Webseiten bitte auf die hier auf der Seite beschriebene PHP Variante umsteigen!

Für einfache Kontaktformulare in TV (mit SPAM-Schutz) steht die Erweiterung (Plugin) [DLRG-Kontaktformular \(TV\)](#) zur Verfügung stehen. Komplexere Formulare können, wie in TYPO3 auch, mit der [TYPO3-Formular](#) Funktion realisiert werden.

Die DLRG bietet ein PHP-basiertes **Kontaktformular**, welches in ähnlicher Form auch den Gliederungen mit TYPO3-TV zur Verfügung steht.

Grundsätzliche Benutzung

PHP-Variante

Die Session ("session"=>true) im Kopfbereich definieren:

```
/* Dokument Konfiguration */
$_DLRG_DOC = array (
    "id"=>"03",
    "titel"=>"Kontakt",
    "keyword"=>"Kontakt,Mail,Fragen",
    "description"=>"Email an uns",
    "session"=>true
)+$_DLRG_DOC;
```

Auf der Seite, welche das Kontaktformular enthalten soll, wird der folgende Schnipsel eingefügt:

```
<!-- Beginn Seiteninhalt -->
<?php
    $_DLRG_CONF["k_zieladressen"] = array(
        "wm" => array("bez" => "Webmaster", "mail" => "webmaster@gld.dlrg.de"),
        "tl" => array("bez" => "Technische Leitung", "mail" => "tl@gld.dlrg.de"),
        "vs" => array("bez" => "Vorstand", "mail" => "vs@gld.dlrg.de"),
    );

    dlrModul("contact");
?>
<!-- Ende Seiteninhalt -->
```

Mit dem oberen Teil stellt man ein, welche Einträge in der Empfänger-ComboBox stehen sollen. Die zweistelligen Werte *tl*, *vs* und *wm* stellen dabei nur technische IDs dar, damit die E-Mail-Adresse nicht im Formular stehen muss (Verhinderung von E-Mail-SPAM). Diese IDs können auch fast beliebig verändert werden (bitte keine Umlaute oder ähnliches (äöüß...)). Kriterien: möglichst kurz und auf jeden Fall eindeutig!

Hinweis: Beim Verändern der obigen Zeilen ist strikt darauf zu achten, dass alle Hochkommata, Kommas, Klammern usw. vorhanden sind, jeweils an den Stellen wie oben! Fehlt irgendwo eines dieser Zeichen (oder ist eines zuviel), kann das Skript nicht funktionieren! Hinweise zur Fehlerursache findet man hier: [Error reporting \(PHP\)](#)

PHTML-Variante

Nach der Anleitung zur PHP-Variante vorgehen, jedoch folgendes zusätzlich *vor dem require_once* einfügen: `$_DLRG_SYS["call"] = "kontakt.phtml";` wobei "kontakt.phtml" für den Dateinamen der Seite steht, welche das Kontaktformular enthalten soll.

HTML-Variante

Für Gliederungen, welche die HTML-Layoutvorlage einsetzen, bedarf es eines kleinen Umweges. In der Seite, welche das Kontaktformular enthalten soll, wird wie bei der PHTML-Variante das Skript und die zusätzliche Zeile eingefügt (siehe Anleitung oben).

Zusätzlich muss die Seite die Dateierendung **.phtml** statt **.html** erhalten, man muss die Seite also umbenennen (dazu muss im Datei-Manager (z.b. Windows-Explorer) eingestellt werden, dass die Dateierendungen nicht ausgeblendet werden.). Beispiel: Aus der Datei **kontakt.html** muss **kontakt.phtml** werden.

TYPO3/TV Variante

Die Beschreibung des Formulars für TYPO3-TV ist hier zu finden: [TYPO3-DLRG Kontaktformular](#)

Weitere Konfigurationsmöglichkeiten

Weiterhin kann man noch folgende Schalter setzen (einfach **vor dem require_once** einfügen):

```
// Anfang der Betreffzeile verändern:
$_DLRG_CONF["k_suffix"] = "DLRG OG Musterhausen";
// Legendentext verändern:
$_DLRG_CONF["k_legend"] = "Kontaktaufnahme:";
// Bestätigungstext:
$_DLRG_CONF["k_best_text"] = "Vielen Dank für Ihr Interesse...";
// CAPTCHA an/aus
$_DLRG_CONF["captcha"] = true/false;
// Die folgenden Einträge sind wichtig, wenn
// das Formular in schmalere Spalten (column) als 70% eingesetzt wird
// Default CSS-Style von Labels
$_DLRG_CONF["def_label_style"] = 'width:7.3em';
// Default Länge von Eingabefeldern
$_DLRG_CONF["def_tf_size"] = 44;
```

Damit das CAPTCHA funktioniert muss die Session aktiv sein.

Experten können noch weitere Felder definieren:

```
$_DLRG_CONF["k_zusatzfelder"] = array(
    'gld' => array("text" => "Gliederung", "size" => "44", "pf" => false),
);
```

Es gibt noch weitere Parameter außer *text*, *size* und *pf*:

| Parameter | Typ | Beispiel/Werte | Beschreibung |
|-----------|------------|--------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| text | String | Gliederung | Text des Labels, einziger Pflichtparameter , Zusatzfeld wird automatisch ein Eingabefeld, kann auch HTML enthalten |
| size | Nummerisch | 10 | Länge des Eingabefeldes. Eingaben in Eingabefelder werden automatisch nach entsprechend vielen Zeichen abgeschnitten. Bei Select-Boxen gibt man mit size die Länge (sichtbare Einträge) an. |
| pf | bool | true/false | Mit diesem Parameter kann man Eingabefeldern zu Pflichtfeldern machen. Wird nur bei Textfeldern und Textareas ausgewertet |
| type | String | <ul style="list-style-type: none">textareaselectradiocheckboxhiddenstatic | Die zusätzlichen Darstellungsarten des Feldes: <ul style="list-style-type: none">mehrzeiliges EingabefeldAuswahlliste (einfach als ComboBox oder mehrfach (siehe size))RadiobuttonsCheckboxesVerstecktes Feld - Wert muss in <i>value</i> fest vorgegeben werdenStatische Texte als Erklärung/Beschreibung |
| options | array | array('w1' => "Wert1", 'w2' => "Wert2") | Hier werden die Optionen für die zusätzlichen Darstellungsformen <i>select</i> , <i>radio</i> und <i>checkbox</i> als Liste von Wertepaaren angegeben. |
| value | String | w1, w2 | In diesem Feld kann man die Inhalte der Felder vorbelegen. Für <i>select</i> , <i>radio</i> und <i>checkbox</i> vorgesehen einfach eine Liste der in Options angegeben Schlüssel eintragen, bei Textfeldern einfach den Text. |
| class | String | myClass | css-Klasse zur Darstellung des Feldes. |
| style | String | width:10em | css-Style zur Darstellung des Feldes. |
| l_style | String | width:10em | css-Style zur Darstellung des Labels. |

| | | | |
|---------|--------|--------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| err_msg | String | Fehler bei der Eingabe von %1 | Feldspezifische Fehlermeldung. Wenn %1 in dem Text enthalten ist, wird dieses mit dem Namen des Feldes überschrieben. |
| regexp | String | <code>^((\[[0-9]{1,3}\][0-9]{1,3}\[0-9]{1,3}\.)(\[[a-zA-Z0-9\-\>\.]+\])([a-zA-Z]{2,4})[0-9]{1,3})(\/?)\$</code> | Nur für absolute Experten! In diesem Feld kann ein regulärer Ausdruck angegeben werden, mit dem die Eingaben in einem Textfeld auf Gültigkeit geprüft werden. Der hier angegebene reguläre Ausdruck prüft z. B. ob eine Web-URL gültig eingegeben wurde. |

Beispiele für Auswahlfelder:

```
// Eine ComboBox (2. Zeile ausgewählt)
'cmb' => array("text" => "MyCombo", "type" => "select", "options" =>
    array('w1' => "Wert1", 'w2' => "Wert2"),
    "value" => "w2"),
// Eine Listbox mit Mehrfachauswahl (2. und 3. Zeile ausgewählt)
'lst' => array("text" => "MyListBox", "type" => "select", "size" => "4", "options" =>
    array('w1' => "Wert1", 'w2' => "Wert2", 'w3' => "Wert3", 'w4' => "Wert4"),
    "value" => "w2, w4"),
// RadioButtons (2. Knopf markiert)
'rdo' => array("text" => "MyRadio", "type" => "radio", "options" =>
    array('w1' => "Wert1", 'w2' => "Wert2"),
    "value" => "w2"),
// Checkboxes (1. und 3. Box angekreuzt)
'chk' => array("text" => "MyCheckboxes", "type" => "checkbox", "options" =>
    array('w1' => "Wert1", 'w2' => "Wert2", 'w3' => "Wert3"),
    "value" => "w1, w2"),
// Ein Bestätigungsfeld, das angeklickt werden MUSS!
'best' => array("text" => "Bestätigung", "type" => "checkbox",
    "options" => array ("ja" => "Hiermit bestätige ich, dass
        ich alles gut durchgelesen habe!"),
    "regexp" => "ja",
    "err_msg" => "Bitte bestätigen Sie durch Ankreuzen des
        Feldes %1, dass Sie die Bedingungen anerkennen!"),
```

Weiterhin kann man noch das Verhalten der beiden Eingabefelder "Betreff" und "Nachricht" (teilweise auch "Name") ändern/überschreiben:

```
$_DLRG_CONF["k_zusatzfelder"] = array(
// Versteckt das Feld "Betreff" komplett
'subject' => array("text" => "Ha! Versteckt", "type" => "hidden",
    "value" => "fester Wert"),
// Das Feld Nachricht heißt nun Bemerkungen und ist kein Pflichtfeld mehr
'text' => array("text" => "Bemerkung", "pf" => false),
// Das Feld Name zum Pflichtfeld machen);
'name' => array("pf" => true)
);
```