

# Ein kurzes Tutorial zu Formularen

Dieser Artikel ist Teil der **HOWTOs**-Reihe.

[Installation](#) • [Nutzung](#) • [PDF-Probleme](#) • [Ausschießen](#) • [Sonstiges](#)

Dieser Wiki-Eintrag wurde mit freundlicher Genehmigung von Gerrit Bruijnes übermittelt.

**Auszug aus „Waar zit de Kneep?“ ([http://wiki.scribus.net/canvas/Waar\\_zit\\_de\\_Kneep%3F\\_Opmaak,\\_Vormgeving\\_%26\\_Publicatie\\_met\\_Scribus\\_1.4.1](http://wiki.scribus.net/canvas/Waar_zit_de_Kneep%3F_Opmaak,_Vormgeving_%26_Publicatie_met_Scribus_1.4.1)) , ISBN 978 90 72445 00 1**

Ein Buch über Scribus 1.4.1...1.4.3 und Veröffentlichung in niederländischer Sprache (Niederländisch) von Gerrit Bruijnes, [www.ipaa.nl](http://www.ipaa.nl). (<http://www.ipaa.nl>)

*Es wurden einige Änderungen vorgenommen, um die Scribus-Quelldatei für konsistenten englischsprachigen Inhalt zu modifizieren, sodass neue Screenshots erhalten werden konnten.*

**Voraussetzungen:** Sie beherrschen das von Ihnen verwendete Betriebssystem. Sie verwenden Scribus 1.4.1...1.4.3 (Sprache: Englisch) und sind mit diesem Programm auf einem angemessenen fortgeschrittenen Niveau vertraut. Wir erklären hier nicht ausführlich, wie Sie gängige Aufgaben wie das Anlegen eines neuen Dokuments, das Eingeben von Text in einen Textrahmen und das Anwenden von Eigenschaften, das Zeichnen von Linien, die Verwendung des Eigenschaftensfelds usw. ausführen.

Wenn Sie hingegen ein erfahrener Programmierer sind, finden Sie diesen Artikel möglicherweise etwas ausführlich.

## Inhalt

### PDF-Formulare

[Woraus besteht ein PDF-Formular?](#)

[Wie erstellen wir ein PDF-Formular?](#)

[JavaScript](#)

### Das PDF-Formular

#### Layout

#### Hinzufügen von PDF-Formularelementen

[Veranstaltungen](#)

#### Symbol auf einer Schaltfläche

#### Felder einrichten

[Namenskonventionen](#)

#### Einrichten anderer Felder

[Bemerkungen](#)

#### Programmierung hinzufügen (Codeausschnitte)

#### Die Kontrollkästchen

[Programmierbegriffe](#)

#### Codeausschnitte für die Combobox

#### Berechnungen

#### Testen des Formulars

[Ergebnisse](#)

[Formularergebnisse](#)

## PDF-Formulare

PDF-Formulare sind so konzipiert, dass ein Benutzer sie online oder lokal auf einem Computer anzeigen, Daten in ein Formular einfügen und es anschließend ausdrucken oder per E-Mail an Sie zurücksenden kann.

### Woraus besteht ein PDF-Formular?

Da es sich im Grunde um ein PDF handelt, können Sie alles hinzufügen, was Sie möchten: Text, Bilder, Formen – kurz gesagt alles, was Scribus zu bieten hat. Wenn Sie ein PDF-Formular erstellen möchten, müssen Sie auch die speziellen PDF-Werkzeuge erlernen. Sie finden diese in der PDF-Symbolleiste. Von links nach rechts sehen wir:



- **PDF-Button** – Ein Button löst eine Aktion aus. Im folgenden Beispiel erstellen wir zwei Buttons: einen zum Drucken des Formulars und einen zum Senden per E-Mail. Ein Button kann mit einem Icon versehen werden. Ein einzelner Button kann bis zu drei verschiedene Icons enthalten, die einen Klickeffekt und sogar ein Icon beim Überfahren des Buttons mit der Maus ermöglichen.
- **Textfeld** – Ein Textfeld ist vergleichbar mit einem Textrahmen, bietet jedoch die Möglichkeit, Zahlen zu verwenden. Denken Sie an Zahlen, die als Datumsangaben dargestellt werden, eine bestimmte Anzahl von Dezimalstellen aufweisen oder als Prozentsatz angezeigt werden.
- **Kontrollkästchen** – Ein bekanntes kleines Kästchen, das beim Anklicken ein Häkchen [x] anzeigt. In unserem Beispiel verwenden wir dieses Kästchen, um einen männlichen oder weiblichen Bewerber für einen Kurs zu kennzeichnen.
- **Kombinationsfeld** – Ein Kombinationsfeld ist eine Dropdown-Liste mit Auswahlmöglichkeiten in eingeklappter Form. Der Benutzer öffnet die Liste, um eine Auswahl zu treffen.
- **Listenfeld** – Ein Listenfeld dient ebenfalls zur Anzeige von Auswahlmöglichkeiten. In diesem Fall sind jedoch immer zumindest einige Elemente der Liste sichtbar. Der Benutzer kann durch die Liste scrollen, muss sie aber nicht zuerst öffnen.
- **Textanmerkung** – Dies ist ein Textfeld mit Anmerkungen. Eine Anmerkung ist im PDF-Formular grundsätzlich nicht sichtbar. Sie können damit nicht nur Text zum Formular hinzufügen, sondern auch Links zu anderen Stellen im Dokument oder zu Webseiten im Internet einfügen.
- **Link-Annotation** – Wird auch für Links zu Stellen im Dokument verwendet. Dabei können sogar die genauen Koordinaten gemäß den in Scribus verwendeten Maßsystemen in Punktform verwendet werden. Auch Links zu externen Websites können verwendet werden.

## Wie erstellen wir ein PDF-Formular?

Beim Entwerfen eines PDF-Formulars müssen wir diese Funktionen berücksichtigen:

- **Ein Layout** – Versuchen Sie, Ihr Formular im Voraus zu entwerfen. Sie können Textrahmen, Tabellen, Bilder und Formen hinzufügen – kurz gesagt alles, was Scribus darstellen kann. Diese Elemente dienen dazu, den Stil Ihres Unternehmens oder Ihrer Organisation zu repräsentieren. Außerdem müssen Sie im Formular angeben, welche Informationen vom Benutzer erwartet werden.
- **PDF-Felder** – Sie müssen die verfügbaren Felder von Scribus verwenden. Wie Sie gesehen haben, gibt es verschiedene Typen. Beachten Sie, dass in Scribus 1.4.1 noch nicht alle Felder implementiert sind! Die Entwicklung ist noch in vollem Gange.
- **Einrichten** – Sie müssen sich mit den vielen möglichen Anpassungen der Felder vertraut machen. Sie können die Felder in verschiedenen Dialogfeldern einrichten. Es gibt viele Möglichkeiten, aber noch nicht alle funktionieren einwandfrei.
- **Validierungen** – Es ist möglich, Grenzwerte für Eingaben in einem Feld festzulegen. So können Sie beispielsweise die Eingabe einer Zahl auf bestimmte Grenzen beschränken, wie wir es in diesem Beispiel tun.
- **Berechnungen** – In einem Dialog können Sie eine bestimmte Berechnung für ein Feld einrichten. Möglich sind beispielsweise die Addition der Inhalte zweier oder mehrerer Felder oder eine Multiplikation usw. Es ist möglich, Höchst- oder Tiefstwerte oder sogar einen Durchschnitt der Werte vieler Felder zu berechnen. Auch komplexere Berechnungen sind möglich, erfordern aber einen höheren Programmieraufwand.
- **Programmierung** – Wenn Sie Ihr Formular intelligenter gestalten möchten, sollten Sie sich mit der Programmierung in JavaScript vertraut machen. Diese Programmiersprache wird nicht nur in PDF-Formularen, sondern auch auf Webseiten verwendet. Wenn Sie noch nie programmiert haben, sollten Sie einige Monate einplanen, um einigermaßen gut darin zu sein. Im nächsten Kasten finden Sie einige Hinweise und Tipps, wo Sie weitere Informationen zu JavaScript finden.

## JavaScript

**Java** ist eine Programmiersprache, die 1995 von Sun Microsystems auf den Markt gebracht wurde. Die verantwortlichen Entwickler James Gosling, Mike Sheridan und Patrick Naughton verbrachten rund vier Jahre mit der Entwicklung der Sprache. Man erzählt sich, dass die Herren viel Kaffee tranken – etwas, das man zwangsläufig tut, wenn man mit dem Programmieren anfängt! Die Kaffeemarke stammte von der Insel Java, daher der Name der Programmiersprache und das Java-Symbol mit der stilisierten Kaffeetasse.

**JavaScript** hat viele Konventionen und Namensdefinitionen von Java übernommen, ist aber dennoch anders aufgebaut. Ursprünglich wurde die Sprache clientseitig in Browsern eingesetzt. Wenn Sie im Internet surfen und eine vom Server angebotene Seite aufrufen, sind Sie der Client. Die Sprache wurde 1995 von Brendan Finch entwickelt, der für Netscape arbeitete. Dieses Unternehmen entwickelte einen bekannten Browser, der Microsofts Internet Explorer lange Zeit starke Konkurrenz machte.

### JavaScript für Acrobat

Der Softwarehersteller *Adobe* hat den PDF-Standard entwickelt. Sie finden den Acrobat Reader wahrscheinlich auf Ihrem PC, da er ein weltweit verbreiteter Reader für PDF-Dokumente ist. Daher ist es nicht verwunderlich, dass Adobe ausführlich erklärt, wie JavaScript in PDF-Formularen optimal eingesetzt wird. Zahlreiche Beispiele und Tutorials finden Sie auf dieser Website:

Javascript auf der Adobe-Site (<http://www.adobe.com/devnet/acrobat/javascript.html>)

### Scribus und Javascript

Es ist etwas schwierig herauszufinden, wie viel Scribus 1.4.1 in Bezug auf JavaScript und Formulare „versteht“. Im Wiki finden Sie einige Beispiele, leider funktionieren nicht alle korrekt oder sind nicht vollständig erklärt. Nehmen wir an, jeder gibt sich Mühe. Hier finden Sie mehr über Scribus und JavaScript in PDF-Formularen:

[Ihr erstes PDF-Formular mit Scribus](#)

[Ihr zweites PDF Formular mit Scribus \(und Javascript\)](#)

## Das PDF-Formular

Das nächste Bild zeigt das Formular, das wir Schritt für Schritt erstellen werden, so wie es in Scribus dargestellt wird. Die roten Kreise dienen lediglich als Markierungen für die folgenden Formularelemente.

Alles andere, wie die Kopfzeile und die Zeilen, sind normale Scribus-Objekte. Es gibt auch einige einfache Textrahmen – die mit **M** und **F**, **Kosten (Euro)**, **Anzahl der Teilnehmer** und **Gesamtsumme (Euro)** sind lediglich Beschriftungen für die angrenzenden Formularelemente.

The screenshot shows a PDF form titled "Learn & Teach" with the subtitle "Training center for English language teachers using TEFL-methods." The form contains several input fields and controls, each marked with a red circle and a number:

- 1**: Textfelder (Text fields) - Marked on the "Name", "Address", "Postcode", "City", "Cost (euro)", "Nr attending", and "Totals (euro)" fields.
- 2**: Kontrollkästchen (Checkboxes) - Marked on the "M" and "F" gender selection boxes.
- 3**: Kombinationsfeld (Combobox) - Marked on the empty field next to "Cost (euro)".
- 4**: Anmerkungsfeld (Text area) - Marked on the large empty rectangular box at the bottom left.
- 5**: Schaltflächen (Buttons) - Marked on the "Print Form" button and the button with a right arrow and a document icon.

Other elements include a "Name" field, an "Address" field, a "Postcode" field, a "City" field, a "Cost (euro)" field, a "Nr attending" field with the value "0", a "Totals (euro)" field, and a "Print Form" button.

Unten sehen wir das Formular, wie es in Adobe Reader angezeigt wird. Abgesehen von den Schaltflächen zeigt das Hellblau die Bereiche an, in denen Eingaben vorgenommen werden können. Das Wort „Name“ ist ein Standardeintrag, der auch die Art der Eingabe angibt. Löschen Sie diesen Eintrag und geben Sie Ihren Namen ein, um das Formular auszufüllen.

The screenshot shows the Adobe Reader interface with a PDF form titled "Learn & Teach". The form is for a training center for English language teachers using TEFL-methods. It includes fields for Name, Address, Postcode, City, Cost (euro), Nr attending, and Totals (euro). There are also checkboxes for M (Male) and F (Female), with F being selected. A "Print Form" button and a button with a right arrow are at the bottom right. A purple banner at the top of the form area says: "Please fill out the following form. You cannot save data typed into this form. Please print your completed form if you would like a copy for your records." The Adobe Reader window title is "PDF form - completed.pdf - Adobe Reader".

Es ist wichtig zu wissen, dass der Entwurf eines Formulars zwar in Scribus erfolgt, aber nicht in dieser Umgebung getestet werden kann! Anders ausgedrückt: Alle Formularelemente sind in Scribus nicht funktionsfähig. Jedes Mal, wenn Sie einen Schritt im Entwurfsprozess abgeschlossen haben und dessen Funktion überprüfen möchten, müssen Sie die folgenden Schritte ausführen:

- Speichern Sie das Formular in Scribus als .sla-Datei.
- Exportieren Sie das Formular im PDF-Format.
- Suchen Sie die resultierende PDF-Datei und öffnen Sie sie in Adobe Reader.
- Testen Sie das Formular.
- Wenn Sie Fehler finden oder mit der Gestaltung fortfahren möchten, schließen Sie Adobe Reader und kehren zur Scribus-Umgebung zurück.
- Nehmen Sie Änderungen vor oder erweitern Sie das Formular.
- Speichern Sie es erneut in Scribus, exportieren Sie das PDF und *überschreiben Sie die vorherige PDF-Version* ! Dies mag selbstverständlich erscheinen, wird aber in der Realität oft vergessen. Dies führt zu vielen verschiedenen Versionen und zu völliger Verwirrung bei Ihnen, wenn Sie versuchen herauszufinden, welche die „wahre“ endgültige Version ist.
- Wiederholen Sie diese Schritte so oft wie nötig, bis das Formular genau Ihren Vorstellungen entspricht. Mit etwas Erfahrung werden Sie mit den verschiedenen Formularelementen vertraut sein und müssen das PDF nicht nach jeder Änderung erneut überprüfen.

## Layout

Für unser Beispiel erstellen wir ein fiktives Anmeldeformular für eine bestimmte Art von Ausbildung für Lehrer für Englisch als Fremdsprache (EFL), die sich in einer bestimmten Methodik qualifizieren möchten.

1. Bereiten Sie ein neues Dokument im A4-Format vor. (*US Letter ist ähnlich.*)
2. Wir benötigen einige Textrahmen. Oben steht der Name des Instituts. Die Farben werden später angewendet. Wir übernehmen die Farbe der Schaltflächenoberfläche. Wir entwerfen ein eigenes Symbol für die Schaltflächen mit einem einfachen Zeichenprogramm wie Paint (unter Windows).
3. In der unteren linken Ecke verwenden wir zwei kleine Textrahmen mit der Aufschrift „M(ale)“ und „F(emale)“.
4. Rechts finden wir drei Textrahmen mit den Texten „Kosten (Euro)“, „Anzahl Teilnehmer“ und „Gesamt (Euro)“. Die Rahmen sind ca. 5,5 mm hoch, die Schriftart ist Arial, 12 pt oder ähnlich. (*Die Beispiele hier im Wiki, die unter Linux erstellt wurden, verwenden DejaVu Sans Condensed.*)
5. Wir verkleinern den Text in den Rahmen etwas, das sieht besser aus. Klicken Sie dazu in der Eigenschaften-Box unter „Text in Spalten & Textabstände“ auf „1 mm Abstand oben“ (Einstellung „Oben“).

6. Zwei dicke Linien werden hinzugefügt. Die bisherigen Ergebnisse sind im nächsten Bild dargestellt. Die Positionen der Hilfslinien sind nicht so wichtig, eine Schätzung genügt. Beachten Sie, dass ihr Hauptzweck darin besteht, verschiedene Objekte auszurichten und ein ausgewogenes Layout zu schaffen.

## Hinzufügen von PDF-Formularelementen

Im nächsten Schritt fügen wir die PDF-Formularelemente hinzu – sieben Textfelder, zwei Kontrollkästchen, eine Combobox und zwei Schaltflächen. Jedes davon hat seine eigene Methode:

- **Textfeld** – Der Benutzer des Formulars kann Informationen in ein Textfeld eingeben. In diesem Beispiel fügen wir einen Standardtext hinzu, der dem Benutzer anzeigt, welche Informationen erwartet werden. Im nächsten Bild sehen Sie links zwei ausgefüllte Textfelder und eines, das noch mit einem passenden Text ausgefüllt werden muss. Sie können Text direkt in die Textfelder eingeben, wie Sie es von normalen Textrahmen gewohnt sind. Wählen Sie ein Textfeld aus und klicken Sie auf die Schaltfläche „Rahmeninhalte“ oder verwenden Sie den Story-Editor.
- **Kontrollkästchen** – Es gibt zwei Felder, die mit einem Häkchen versehen werden können. Wir verwenden sie zur Angabe des Geschlechts. Es ist schwierig, gleichzeitig Frau und Mann zu sein, daher sollte das Formular nur ein Häkchen in einem der Kästchen zulassen. Mit einem kleinen Programmierschritt stellen wir sicher, dass ein Häkchen in einem der Kästchen das Häkchen aus dem anderen entfernt. In anderen Situationen könnten wir eine Liste nicht-exklusiver Optionen anbieten, aus denen der Benutzer auswählen kann. Da mehrere Auswahlmöglichkeiten zur Verfügung stehen, wäre diese restriktive Programmierung nicht erforderlich.
- **Kombinationsfeld** – Wir füllen das Kombinationsfeld mit verschiedenen Schulungsoptionen. Das Ausfüllen des Feldes erfolgt durch ein kleines Programm, sobald der Benutzer darauf klickt. Sobald die Anmeldung zu einer bestimmten Schulung erfolgt ist, werden die Kosten in eines der Textfelder eingetragen. Auch hierfür ist ein wenig Programmierung erforderlich. So wie wir es hier machen, wird ein Ergebnis nur dann eingegeben, wenn der Benutzer in ein anderes Feld klickt. Wir zwingen den Benutzer dazu, indem wir bei der Anzahl der Bewerber eine Null eintragen. Der Benutzer muss dies ändern, damit das Programm ausgeführt wird; aktivieren Sie dazu das nächste Kontrollkästchen mit der Bezeichnung „Ereignisse“. Es gibt auch andere Möglichkeiten, dies zu tun, diese liegen jedoch außerhalb des Rahmens dieser Einführung in die Programmierung von PDF-Formularen.
- **Schaltflächen** – Durch Klicken auf eine Schaltfläche kann eine Aktion ausgelöst werden. Als erfahrener Softwarenutzer sind Sie daran gewöhnt. Die bekannteste Schaltfläche, die wir alle täglich verwenden, ist der OK-Button. Auch Scribus ermöglicht es, Formularen solche Schaltflächen hinzuzufügen. Hinter der Schaltfläche zum Drucken des Formulars verbirgt sich ein Programmausschnitt, der diese Aktion für Sie ausführt. Die Schaltfläche zum Senden des Formulars benötigt keine Programmierung, da diese Aktion in einem Dialogfeld definiert werden kann.

## Veranstaltungen

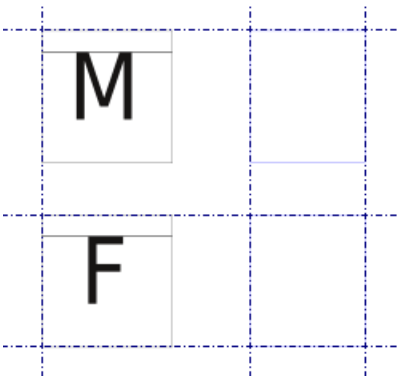
Das Klicken auf eine Schaltfläche oder in ein Textfeld ist ein **Ereignis**. Auch wenn Sie etwas in ein Textfeld eingeben oder eine Auswahl in einer Listbox oder Combobox treffen und anschließend in ein anderes Feld klicken, bezeichnen wir dies als Ereignis. Ein Ereignis ist mit einem Textfeld, einer Schaltfläche oder einem anderen Steuerelement in einem PDF-Formular verknüpft. Jedes erkennbare Ereignis bietet die Möglichkeit, einen Programmausschnitt hinzuzufügen.

Nehmen Sie sich nun die Zeit, den nächsten Satz auf sich wirken zu lassen:

*Wenn mit einem Ereignis keine Programmierung verknüpft ist, passiert bei dessen Eintreten nichts. Ist mit einem Ereignis ein Programm verknüpft, wird es bei jedem Eintreten des Ereignisses ausgeführt!*

In unserem Formular fügen wir eine Schaltfläche ein, die das Formular druckt, sobald der Benutzer darauf klickt. Wir verwenden das Ereignis „Mausklick“ (die Maustaste wird gedrückt, wenn sich der Mauszeiger über der Schaltfläche befindet), um das Fragment zu programmieren, das diese Aufgabe ausführt. Wenn der Benutzer beispielsweise zehnmal auf die Schaltfläche klickt, wird das Formular zehnmal gedruckt. Diese Programmiermethode nennt man ereignisgesteuerte Programmierung. Alle gängigen Betriebssysteme wie Apple OS X, OS/2, Linux, Windows und andere basieren auf diesem Prinzip. Sie sind bereits damit vertraut. Jetzt lernen Sie die Grundlagen dieser Technik kennen und programmieren sie selbst.

1. Zeichnen Sie links im Formular ein PDF-Textfeld.
2. Bauen Sie es wie folgt auf:  
Schriftart: Arial, 12 pt, 1mm Abstand nach oben, Breite ca. 60 mm und Höhe 5,5 mm.
3. Kopieren Sie dieses Feld dreimal (Windows: Strg + C und dann Strg + V). Platzieren Sie die Kopien sauber untereinander.  
*Sie können für diese Aufgabe auch „Element > Mehrfach duplizieren“ verwenden, indem Sie entweder eine Reihe von Kopien mit einem bestimmten vertikalen Abstand erstellen oder mehrere Zeilen mit dem von Ihnen festgelegten Abstand anlegen.*
4. Klicken Sie in das erste Textfeld und wählen Sie im Popup-Menü „Text bearbeiten...“. Das nächste Bild zeigt dies nur für das Feld, das die Postleitzahl enthalten muss.
5. Der Story-Editor wird geöffnet. Geben Sie den Text ein und wählen Sie anschließend im Menü „Datei“ die Option „Text bearbeiten und schließen“. Der Text wird im Textfeld angezeigt.
6. Wiederholen Sie diesen Vorgang, um die folgenden Texte in die nächsten vier Felder einzufügen: Name, Adresse, Postleitzahl und Ort.
7. Zeichnen Sie neben den Textrahmen zwei quadratische Kontrollkästchen mit der Aufschrift „M“ und „F“.



Auf der rechten Seite des Formulars benötigen wir die folgenden PDF-Steuerelemente:

1. Oben eine Combobox.
2. Drei Textfelder unterhalb der Combobox, rechts neben den drei bereits vorhandenen Textrahmen.
3. Unterhalb dieser Gruppe zeichnen Sie zwei Schaltflächen.
4. Das nächste Bild zeigt, wie es bisher aussehen sollte. *Beachten Sie, dass der ausgewählte Rahmen (rot umrandet) oben nicht erwähnt wird, sondern ein Textanmerkungsrahmen ist.*



# Learn & Teach

Training center for English language teachers using TEFL-methods

Name		
Address		
Postcode	Cost (euro)	
City	Nr attending	
	Totals (euro)	
M		
F		

## Symbol auf einer Schaltfläche

Schaltflächen können mit einem kleinen Bild versehen werden. Wir nennen das ein Symbol. Alternativ können Sie der Schaltfläche auch einfach einen kurzen Text hinzufügen. Das Hinzufügen von Text zu einer Schaltfläche in Scribus ist vergleichbar mit dem Hinzufügen von Text zu einem Textfeld. Auch ein Symbol für eine Schaltfläche ist durchaus möglich.

1. Verwenden Sie ein Zeichenprogramm, beispielsweise Paint (niederländische Version) von Windows – wie *im Buchbeispiel des Autors verwendet*.
2. Definieren Sie die Leinwand (Größe der Zeichnung), die der Größe des Buttons entsprechen soll. Sie muss nicht sehr groß sein, 200 x 100 Pixel sollten völlig ausreichend sein. Sie finden diese Einstellung in jedem Programm unter Eigenschaften im Menü Datei.
3. Mit einem Zeichenprogramm können Sie Text hinzufügen und mithilfe einer Schriftart wie Wingdings oder Symbol Sonderzeichen wie eine Computertastatur, einen Pfeil und einen Umschlag hinzufügen.
4. Fügen Sie etwas Hintergrundfarbe und ein paar dicke Linien in einem etwas dunkleren Farbton hinzu, um den Eindruck eines Schatteneffekts zu erzeugen.
5. Speichern Sie die Zeichnung der Schaltflächenfläche und wechseln Sie zu Scribus.

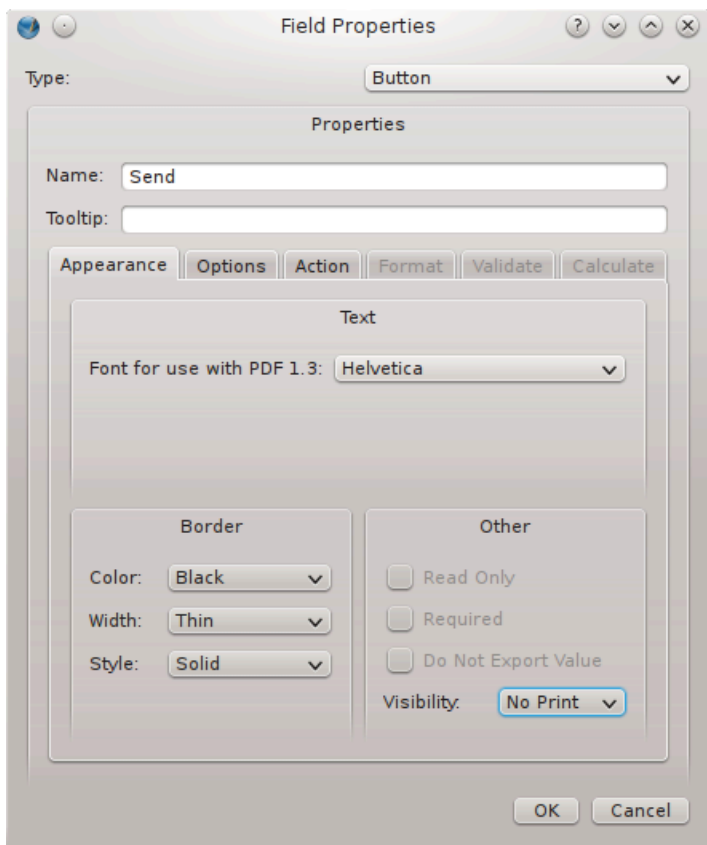


Die oben abgebildete Grafik wurde vollständig in Scribus mit der Schriftart Dingbats erstellt, mit einem Screenshot-Programm erstellt und anschließend als PNG für das Formular gespeichert. Es gab drei Rahmen: vorne einen Textrahmen ohne Hintergrundfarbe und die Symbole, darunter eine leere Kopie des Textrahmens mit orangefarbenem Hintergrund und dazwischen einen kleinen Rahmen mit weißem Hintergrund, damit der Umschlag weiß bleibt. Wie Sie sehen, eignet sich Scribus auch als Zeichenprogramm.

## Felder einrichten

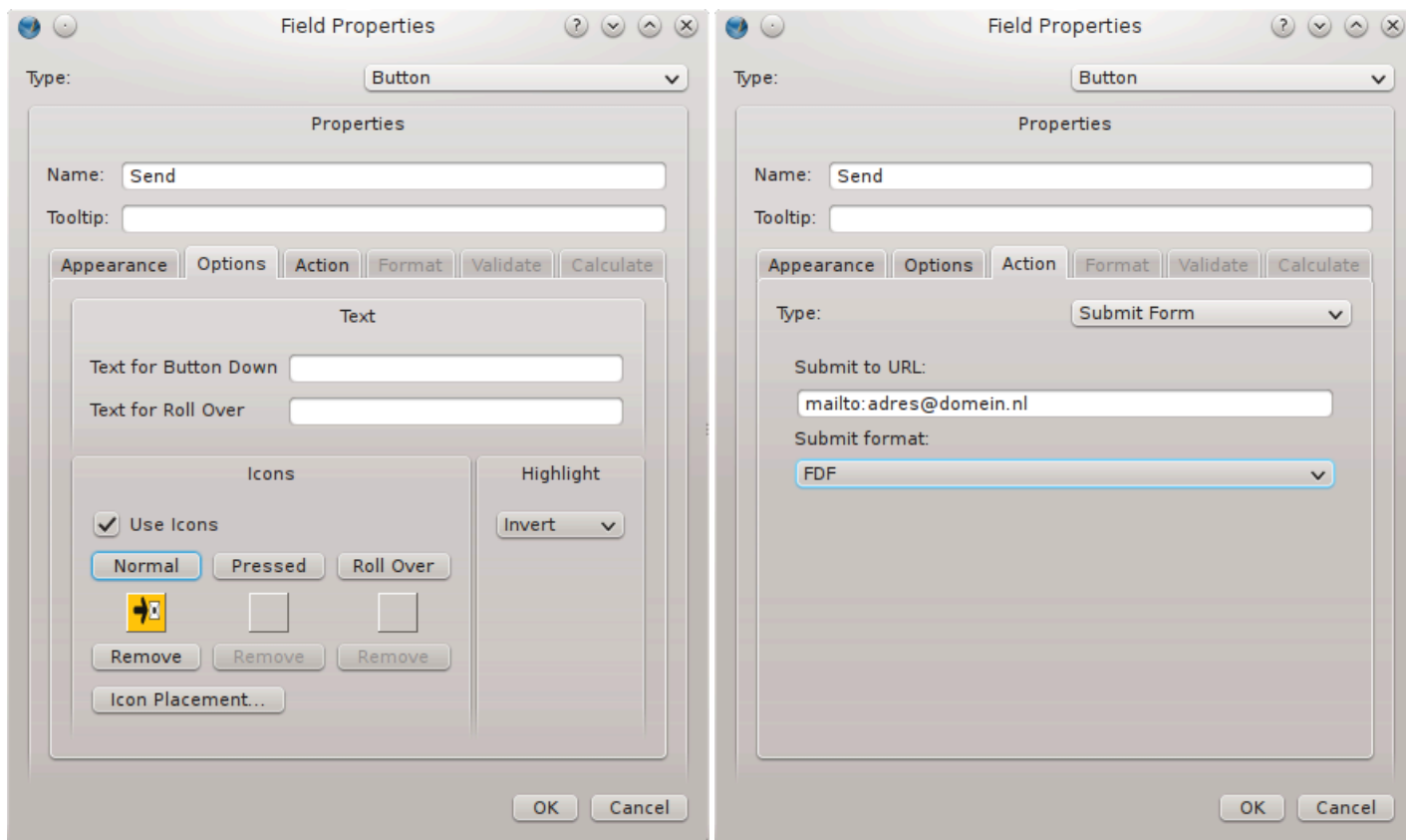
**Als Nächstes müssen wir alle Felder über das Dialogfeld Feldeigenschaften einrichten .** Beginnen wir mit einem der Buttons.

1. Doppelklicken Sie auf die Schaltfläche rechts. Das Dialogfeld Feldeigenschaften wird geöffnet.
2. Weisen Sie dem Feld einen Namen zu. Wenn Sie diesen überspringen, erhalten alle Steuerelemente im Formular einen generischen Namen, der aus dem Feldtyp und einer Nummer besteht. Das ist nicht ganz eindeutig. Gehen Sie stattdessen auf die Registerkarte „Darstellung“ und vergeben Sie hinter „Name“ einen Namen, der den Zweck des Felds klar beschreibt.
3. Bei Sichtbarkeit wählen wir „Kein Drucken“. Wenn der Benutzer das Formular ausdrucken möchte, ist es ja nicht notwendig, die Schaltflächen jetzt auszudrucken, oder?



1. Wählen Sie die Registerkarte Optionen und klicken Sie auf die Schaltfläche Normal.
2. Das Dialogfeld „Öffnen“ wird geöffnet und Sie wählen die Schaltflächenfläche (Symbol) aus, die Sie gerade vorbereitet haben.
3. Bedenken Sie, dass Sie bis zu drei Symbole verwenden können: Normal (was die meiste Zeit sichtbar ist), Gedrückt (was sichtbar ist, wenn auf die Schaltfläche geklickt wird) und Rollover (sichtbar, wenn der Mauszeiger über der Schaltfläche schwebt). In diesem Beispiel beschränken wir uns auf nur ein Symbol.
4. Klicken Sie auf die Registerkarte Aktion und wählen Sie bei Typ: Formular übermitteln aus.
5. Geben Sie die URL in das Feld „An URL senden:“ ein. Die Abbildung unten rechts zeigt, wie eine E-Mail-Adresse verwendet wird. Adobe Acrobat Reader versucht, einen E-Mail-Client auf dem Computer des Benutzers zu öffnen. Schlägt dies fehl, wird ein Dialogfeld mit Auswahlmöglichkeiten angezeigt.
6. Die Einrichtung der Schaltfläche ist nun abgeschlossen.
7. Der linke Button, der den Druck des Formulars ermöglicht, kann auf die gleiche Weise mit einem Symbol ergänzt werden. Wir haben hier lediglich den Story-Editor verwendet, um einen Text hinzuzufügen; siehe die nächsten Bilder.
8. Wenn Sie wie wir eine Hintergrundfarbe für die Schaltfläche verwenden, mit der die Formularergebnisse per E-Mail versendet werden, können Sie diese Farbe mit dem Pipettierwerkzeug von Scribus zu den Farben des Dokuments hinzufügen und damit den Titel des Formulars und die beiden Zeilen hervorheben, die wir zuvor hinzugefügt haben.





## Namenskonventionen

Durch die den Feldern zugewiesenen Namen können wir auf der „Rückseite“ des Formulars bestimmte Aufgaben mithilfe von Programmausschnitten ausführen. Die Benutzerseite (Vorderseite) und die Programmierseite (Rückseite) des Formulars müssen nicht identisch aussehen.

In practice generic names are cumbersome, because they are also used to present the results of a form when it is send by e-mail. It's much nicer to receive a result that looks like this:

Name	G. Bruijnes
Address	Langstraat 6
Postal code	1234 AA
City	Stadje
Female	ON

A much less clear result with generic names would look like this:

Text22	G. Bruijnes
Text37	Langstraat 6
Text42	ON

Naming can be in a language that differs from the language used in the form, because the results may be processed by employees who, in this case, do not speak English. Anyway, it is not necessary and you may use anything you want.

Programmers often use a different type of naming, the so called Hungarian Notation. The type of object is mentioned in the name (we call a field an object or control). In this case English is the predominant language and names would look like this:

```
chkFemale
txtAcqAddress
btnPrintForm
```

The prefix **chk** makes it immediately clear that we are dealing with a check box, **txt** would denote a textfield, and **btn** refers to a button.

## Set up other fields

You know now that you can set up the fields by double clicking them in order to make choices in the dialog box Field properties. The next table gives you all necessary information:

Control	Tab	Set up
---------	-----	--------

Checkbox (M)	Appearance Options	Name: M Default is Not checked
Checkbox (F)	Appearance Options	Name: F Default is Checked (Most teachers are female!)
Combo box	Appearance	Name: Training
Field next to Cost	Appearance Format	Name: Cost Field formatted as: Number Decimals: 0 ( <i>zero</i> ) Use Currency symbol: No tick mark ( <i>Does not work well</i> ) Formatting: 9,999.99
Field next to Nr Attending	Appearance Format	Name: NrAt Field formatted as: Number Decimals: 0 ( <i>zero</i> ) Use Currency Symbol: No tick mark Formatting: 9,999.99
	Validate	Value must be greater than or equal to: 1
	In Story Editor: Input Alignment	0 ( <i>zero</i> ) right
Field next to Totals	Appearance Format	Name: Totals Field formatted as: Number Decimals: 0 ( <i>zero</i> ) Use Currency symbol: No tick mark Formatting: 9,999.99
If you have added an Annotation Type: Field	Destination:	External weblink <a href="http://www.ipaa.nl">www.ipaa.nl</a> ( <i>the author's website</i> )

## Remarks

When defining the field NrAt you will see that we use a validation. This means that as long as the number 0 (zero), the form is considered to be not completed correctly. Thus we force the user to fill in a number equals to or larger than 1. By doing so the price of the training selected in the combo box will be processed in the text field Cost.

Check out the other possible links in the Annotation field. You'll see that it is also possible to refer to particular locations inside the document. This completes the setup of the fields.

## Adding programming (code snippets)

As I have explained before we use JavaScript to automate certain events. There is no need for you to have any specific programming experience to be able to understand the code snippets that we use here. If you want to make more pdf forms yourself, there will be no escape, you'll have to foot it and learn the language, not meaning English here!

Remember:

- A correct working piece of programming can only be obtained by using all kinds of special characters which very often have a special meaning. Every semicolon (;) and every equals sign (=) is essential. Sometimes even a space can make the difference!
- Copy all examples in exactly the same way, maintain the same lay-out.
- JavaScript uses braces abundantly. Use the right type { (open) or } (close) in the correct spot.
- Check your code snippets on correct spelling of the words. One misplaced letter and the fragment will not perform its task without any warning.
- Press the Enter-key at the end of each line.

- If you can't get it to work, [File:PDF form - completed.sla.gz](#) is a fully-fledged version so you can still check out how it works.

## The check boxes

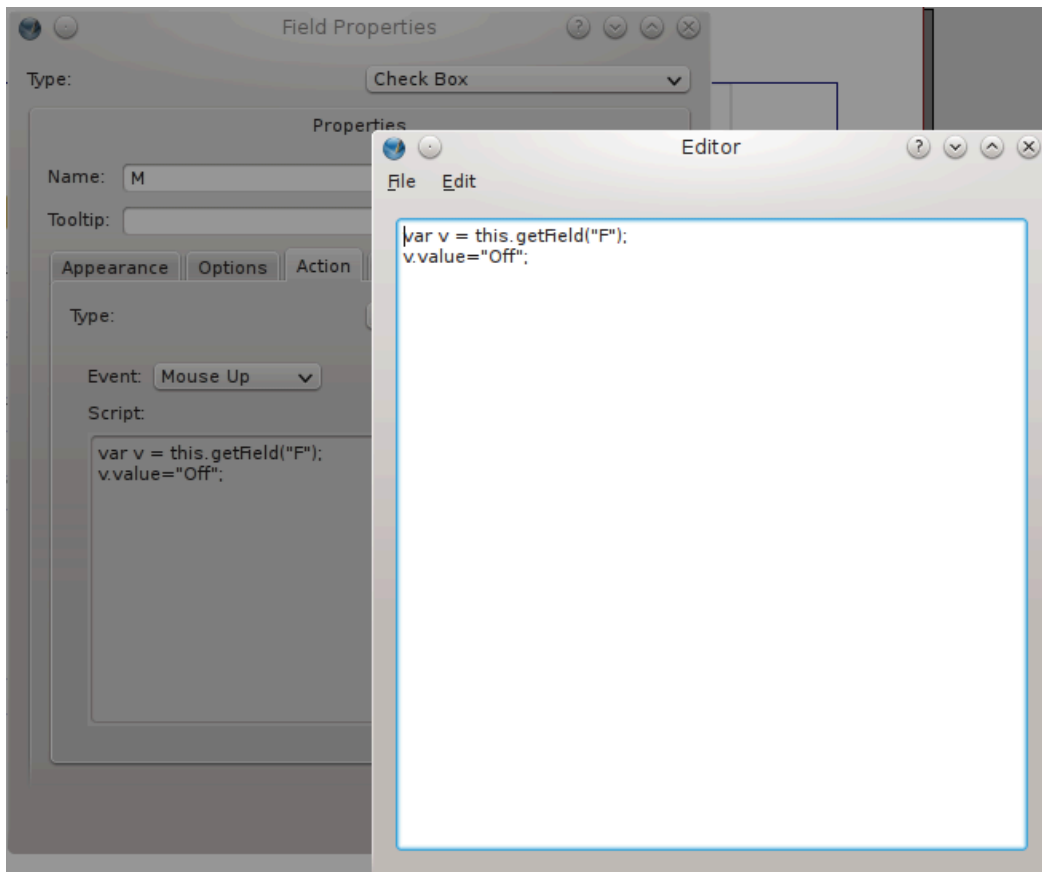
First we're going to fix the check boxes. Most teachers are females so that's why we tick off the check box F (see previous table).

- Double click check box M.
- Select the tab Action in the dialog box Properties.
- Choose Type: JavaScript.
- Choose Event: Mouse up.
- Click the button Edit.
- Copy the following code snippet:

```
var v = this.getField("F");
v.value="Off";
```

How to interpret this? The snippet is located in the **Mouse up** event of the check box M. So if some clicks here as a result a tick mark will be placed in the box. This means however that the standard tick mark in checkbox F must be removed. In the code snippet we first point to the field F. Once we have established this, we can turn its value "Off".

- If you're done, you chose *File > Save and Exit...* in the Editor's menu.
- Now you will see the code snippet in the dialog box.



- Click the OK button in the Field Properties dialog.
- Double click check box F.
- Select the tab Action and add the same set up as you did with check box M.
- The code snippet is comparable, but a little bit different. A user may make a mistake or change his mind so we also have to be able to remove the tick mark from check box M:

```
var m = this.getField("M");
m.value="Off";
```

## Programming terms

The meaning of the abbreviation **var** is variable. A variable is a random name, by using this name we can refer to something. In the code snippets above, it refers to the fields with the names M and F respectively. These are the fields we have placed in the form right next to the text fields M(ale) and F(emale). Value is a property of an object. The object is the field. Every object has properties, like in real life where for instance a car possesses a property Color and a property NumberOfDoors. While programming, properties can always be read and often set as well.

An event occurs when something happens like clicking with the mouse on a button or moving the mouse over an object such as a text field or a check box.

*Note that the variable **v** shows that the name of the variable in JavaScript doesn't have to match the field name.*

## Code snippets for the combo box

The combo box utilizes two programming fragments. One is active when the user clicks the box and supplies the box with the list of possible courses, thus presenting a number of choices. We ought to do this at the time the form is loaded into the PDF reader, but this is outside the scope of this article, so we do it this way.

- Double click the combo box.
- Select the tab Action. Choose Type: JavaScript.
- As event choose: On focus.
- Click the button Edit and copy the following code snippet:

```
var f = this.getField("Training");
f.editable = false;
f.doNotSpellCheck = true;
f.commitOnSelChange = true;
f.setItems( ["Choose course!", "Basic", "Intermediate", "Advanced"]);
```

It is possible to prevent a change in the settings of a field using programming code. Here we deny the user the possibility to fill in a non-existing training (f.editable = false;). If we allowed editing, the whole thing gets a lot more complicated, because we would also have to check the supplied user info. Spelling check is also disabled because we will add the correct names of trainings ourselves.

The combo box will be filled with four possibilities, the first being **Choose course!** It cues the user as to what information is required. The other three entries describe the available courses – you can add as many as you want, but then you will have to adjust the following code snippet!

- Choose Save and Exit in the editor's File menu.
- Select the next event, Event: On blur (this denotes the clicking with the mouse in the box or using the Tab key to hop the cursor into the combo box). Copy this fragment:

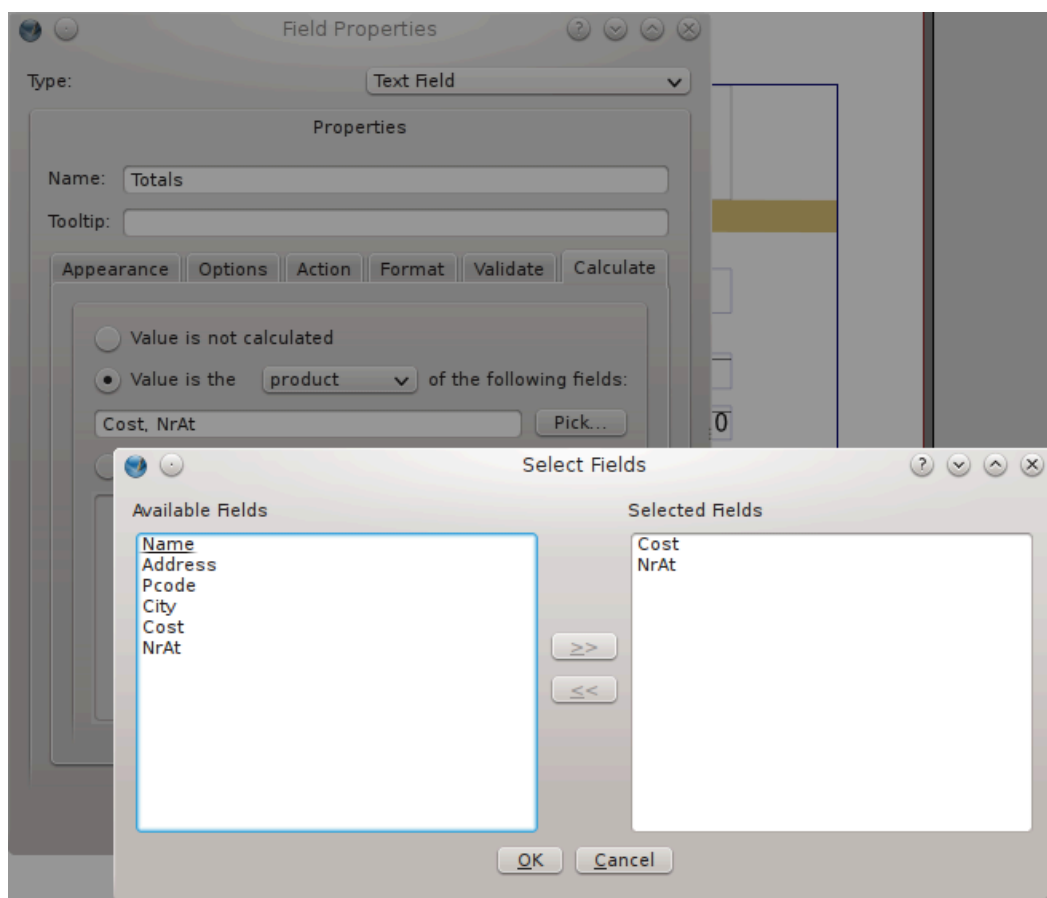
```
var t = this.getField("Training");
var c = this.getField("Cost");
switch (t.value){
  case 'Basic': {
    c.value=250;
  }
  break;
  case 'Intermediate': {
    c.value=275;
  }
  break;
  case 'Advanced': {
    c.value=325;
  }
  break;
  default: {
    c.value=0;
  }
}
```

Variable **c** refers here to the text field with the name Cost. Depending on the choice that was made in the combo box, an amount is written to this field. There is a default, 0 (zero), which is valid as long as no choice has been made.

## Calculations

There is one thing left to take care of. It's the calculation needed to multiply the number of participants with the cost of the course.

- Click the field behind the text frame reading 'Totals'.
- Select the tab Calculate.
- Click the radio button Value is the and select product in the list.
- Click the button Pick... and take care that the fields Cost and NrAt in the box on the right; see the next image.
- Close all dialogs with OK.



## Test the form

This is the moment of truth, let's test the form. Here is an overview of the process:

- Save your work in Scribus.
- Export the file as a pdf.
- Save the file again in Scribus so changes you made to the way Scribus export pdf's will also be saved.
- Find the pdf on disk and open it with Acrobat Reader or any other pdf reader.
- Check if everything is working correctly.
- If changes area required, close the pdf, make changes in the Scribus environment and repeat the above steps.

Let's do it shall we?

- First go to Scribus, open the menu File and choose Save.
- Go back to the menu File and choose Export. Choose Save as PDF...
- You only have to make changes on just a few tabs:

Tab	Change	Remark / setting
General	Output to File:	Use a name and a location that will make it easy for you to find the file afterwards.
	Compatibility	At least PDF 1.4 (Acrobat 5)
Fonts	Fonts to embed	Click: Embed all
Color	Output Intended For:	Screen / Web

- Now click the button Save.
- Open Explorer (Windows) or any other appropriate file manager for the operating system that you use, e.g. Nautilus (Ubuntu) or Finder (Apple). Find the pdf and open it. *You might just run Adobe Reader and open the file from it.* It may be possible that the reader will display a warning. This warning is obvious, the form is capable of e-mailing data.

- Check if everything is working as it is intended. Click the combo box, select a course. Also test the function of the Print form button.

If an e-mail client (e.g. Outlook for Windows) is installed, the form results will be processed directly. In case of an online mail service like Yahoo or Gmail, the pdf has to be saved first (only possible with Adobe Reader version 9 or higher) and then the results can be sent via e-mail.

When saving you can choose several file types:

- .fdf** Adobe's form file format, only the data in the fields is saved. In this case it is necessary that the receiver also has a copy of the original form. The data sent by e-mail will then be automatically added to the form.
- .xfdf** The extended form file format. It is capable of showing more options.
- .xml** A file format that is capable of showing data in the appropriate fields on a webpage.
- .txt** In diesem Fall werden die Daten als einfache Textdatei gespeichert, die mit jedem auf Ihrem System verfügbaren Textverarbeitungsprogramm oder Texteditor geöffnet und gelesen werden kann.

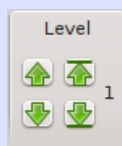
## Ergebnisse

So sehen die Ergebnisse als FDF-Datei aus:

```
%FDF-1.2
1 0 obj
<</FDF<</Fields
[<</T(Address)/V(1234 Main St)>>
<</T(City)/V(Shangri-La)>>
<</T(Cost)/V(250)>>
<</T(F)/V/Off>>
<</T(M)/V/Yes>>
<</T(Name)/V(Gregory Pittman)>>
<</T(NrA)/V(1)>>
<</T(Pcode)/V(3298Z)>>
<</T(Tekst19)>>
<</T(Totals)/V(250)>>
<</T(Training)/V(Basic)>>]
/ID[<0C8B119972294B0174917396D35A341F><0C8B119972294B
endobj
trailer
<</Root 1 0 R>>
%%EOF
```

## Formularergebnisse

Normalerweise werden alle Felder nacheinander ausgedruckt. Im vorherigen Bild haben wir sie der Übersichtlichkeit halber jeweils in einer eigenen Zeile dargestellt. Sie werden in der gleichen Reihenfolge angezeigt, wie sie im Formular platziert wurden. Wenn Sie eine andere Reihenfolge gewählt haben, wird sich dies im Ergebnis widerspiegeln. Dies sollte kein großes Problem darstellen, da die Feldnamen ebenfalls in die Datei übernommen wurden. Ein guter Grund, Ihre Felder sorgfältig zu benennen und nicht die Standardnamen zu verwenden.



Die Reihenfolge der Felder lässt sich in Scribus ändern. Verwenden Sie dazu die Eigenschaften-Box und auf der Registerkarte X, Y, Z die Schaltfläche Ebenen. Nach Auswahl eines Feldes wird dessen rotierende Nummer angezeigt. Der Feldname befindet sich ebenfalls oben auf dieser Registerkarte und kann hier ebenfalls geändert werden!

*Hinweis: Dies ist möglicherweise nicht der Fall. Aus dem obigen Beispiel geht hervor, dass sich die Reihenfolge auf eine alphabetische Auflistung basierend auf dem Feldnamen bezieht. Beispielsweise befand sich die Variable Name im Scribus-Dokument auf Ebene 1.*

Abgerufen von „[https://wiki.scribus.net/wiki/index.php?title=A\\_Brief\\_Tutorial\\_on\\_Forms&oldid=29913](https://wiki.scribus.net/wiki/index.php?title=A_Brief_Tutorial_on_Forms&oldid=29913)“

Diese Seite wurde zuletzt am 8. April 2014 um 19:51 Uhr bearbeitet.

Sofern nicht anders angegeben, sind die Inhalte unter der Bedingung „Namensnennung – Weitergabe unter gleichen Bedingungen“ verfügbar .