

***Ein RagTime-Dokument mit Hilfe von RagTime Connect  
unter Mac OS X an eine MySQL-Datenbank anbinden***

***Für RagTime 6.5 unter Mac OS X 10.5  
überarbeitete Version***

von Benedikt Quirnbach

Lektoriert von Thomas Kaegi

Überarbeitung für RT 6.5/OS X 10.5: Thomas Kaegi

12.3.2009

# Inhalt

Systemvoraussetzungen .....	3
Was ist MAMP? .....	3
Warum MAMP benutzen? .....	3
Warum nicht den in Mac OS X eingebauten WebServer Apache benutzen? .....	3
Was ist phpMyAdmin? .....	4
Was ist RagTime-Connect?.....	4
Einbahnstrasse.....	4
Hinweise zur Installation von RagTime Connect.....	5
Eine Datenbank anlegen .....	5
MAMP starten.....	5
phpMyAdmin starten .....	6
Datenbank anlegen .....	6
Tabelle anlegen .....	7
Daten eintragen.....	11
Eine ODBC-Datenquelle einrichten.....	13
ODBC-Datenquelle mit ODBC-Administrator einrichten .....	13
RagTime Connect .....	15
RagTime-Connect-Komponente in RagTime erzeugen .....	15
Erste Abfrage formulieren.....	16
Alle Daten aus der Tabelle holen .....	16
RagTime-Connect im Inventar .....	17
Die Adressdaten in einem Brief-Formular nutzen .....	18
Empfänger-Adresse anlegen .....	18
Wie geben Sie solche Formeln ein?.....	18
Was bedeutet «\$0»? .....	19
Durch die Datensätze «blättern» .....	19
Brief-Formular benutzen.....	20
Fixierung der benutzten Daten .....	20
Fixierung aufheben.....	20
Ein Beispiel mit dynamischem Datenbank-Zugriff.....	21
Den Ort aus der Liste der vorkommenden Orte auswählen .....	21
Eine selektive Abfrage definieren.....	23
Die SQL-Formel für die selektive Abfrage zusammenstellen.....	23

# Systemvoraussetzungen

Unter der folgenden Konstellation wurde diese Anleitung getestet:

- RagTime 6.5 mit RagTime-Connect
- Mac OS X 10.5.6 auf MacBook Pro (Intel)
- MAMP 1.7.2  
(kostenloser Download: <http://www.mamp.info/de/index.php>)

## Was ist MAMP?

MAMP ist eine Abkürzung. Sie steht für **M**acintosh, **A**pache, **M**ySQL, **P**HP. Es handelt sich dabei um eine Programmsammlung für Apple Macintosh (Mac OS X), die einen kompletten Internetserver mit Datenbank zur Verfügung stellt.

### Warum MAMP benutzen?

MAMP ist kostenlos, nur für die Pro-Version ist ein kleiner Betrag fällig. Für unsere Zwecke reicht die Basis-Version vollkommen aus.

Mit der Installation von MAMP bekommt man alles, was man für professionelle Datenbank-Anwendungen braucht:

- MySQL ist ein sehr mächtiger Datenbank-Server. Er ist wohl der im Internet am häufigsten genutzte Datenbank-Server überhaupt.
- Apache und PHP sind ein Gespann aus einem kompletten Webserver und einem Programm-Interpreter, der die Scriptsprache PHP verarbeitet. Auch diese Kombination dürfte im Internet am weitesten verbreitet sein.
- Durch die weite Verbreitung im Internet bekommt man über alle Komponenten sehr viele Informationen. Die Handbücher sind im Internet frei verfügbar, es gibt tausende von Tutorials, es gibt hunderte von online-Foren und Mailinglisten, die jedes erdenkliche Thema zu den Programmen behandeln.

### Warum nicht den in Mac OS X eingebauten WebServer Apache benutzen?

MAMP hat den Vorteil, dass man nach dem kostenlosen Download den kompletten Server inkl. Konfiguration mit einem Klick installieren kann. Nach der Installation ist alles sofort benutzbar.

Dazu gehört auch das Datenbank-Verwaltungswerkzeug phpMyAdmin.

## *Was ist phpMyAdmin?*

Die Programme, die mit MAMP installiert werden, besitzen keinerlei Benutzeroberfläche. Man sieht also gar nichts. Man kann sie konfigurieren, indem man jeweils eine Textdatei bearbeitet. Man kann sie bedienen, indem man ziemlich kryptische Befehle ins Terminal-Fenster tippt.

Da wir hier MAMP benutzen, erübrigen sich Konfigurationsarbeiten. Zumindest solange es sich um eine Testumgebung handelt, die nicht mit dem Internet verbunden ist. Bei Verbindungen zum Internet muss man einige Sicherheitsaspekte in der Konfiguration beachten.

Aber für die Verwaltung von Datenbanken (anlegen, löschen, Daten eintragen usw.) wäre eine durchdachte Benutzeroberfläche sehr angenehm.

Und genau das liefert phpMyAdmin. Es ist eine WebSite, die alle Aspekte der Bedienung eines MySQL-Servers abdeckt (ausgenommen die Grundkonfiguration). Um phpMyAdmin benutzen zu können, benötigt man einen WebServer und PHP. Damit wären wir wieder bei den Komponenten von MAMP.

## *Was ist RagTime-Connect?*

RagTime-Connect ist eine Komponente für RagTime, die über ODBC eine Verbindung zu einer Datenbank herstellt. Es ist dabei vollkommen gleichgültig, um welches Datenbank-System es sich handelt. Man muss einfach einen passenden ODBC-Treiber für das gewünschte Datenbank-System installiert haben.

ODBC (**O**pen **D**ata**B**ase **C**onnectivity) bildet eine Art Software-Zwischenschicht, die zwischen den Anwendungen – hier RagTime – und den Datenbanken mit Hilfe der Treiber vermittelt.

Es ist nicht notwendig, dass der Datenbank-Server auf dem selben Rechner läuft, wie RagTime. Bei den Beispielen, die mit RagTime-Connect geliefert werden, ist es beispielsweise so, dass die Datenbanken auf einem Server im Internet liegen.

Der Treiber für MySQL wird mit RagTime-Connect zusammen installiert. Man braucht sich also hier um nichts weiter zu kümmern. Es gibt aber auch Treiber für viele andere Datenbank-Systeme, die man zumeist käuflich erwerben muss.

RagTime-Connect gibt es ab RagTime 6.5 nur noch in der Pro-Version, die lizenziert werden muss.

### *Einbahnstrasse*

RagTime-Connect ist eine Einbahnstrasse. Man kann Daten aus einer Datenbank holen, aber keine Daten in die Datenbank hineinschreiben.

## *Hinweise zur Installation von RagTime Connect*

- Wenn Sie eine frühere Version von RagTime-Connect installiert hatten, so sollten Sie vor der Installation von RagTime Connect mit dem «ODBC-Administrator» (im Ordner «Programme/Dienstprogramme») alle bisherigen Datenquellen löschen, bevor Sie mit der Installation von RagTime-Connect beginnen. Der «ODBC-Administrator» ist leider ein recht unstabiles Programm, das häufig abstürzt. Lassen Sie sich nicht entmutigen – versuchen Sie es einfach nochmals.
- Am Schluss der Installation von RagTime-Connect wird eine Fehlermeldung angezeigt. Trotzdem ist RagTime Connect richtig installiert. Wie festgestellt wurde, wird die Fehlermeldung durch eine Installationsroutine von MySQL erzeugt, die bei der Installation von RagTime-Connect benützt wird, hat also nichts mit RagTime zu tun.

## *Eine Datenbank anlegen*

In dieser Anleitung wird nicht auf alle Besonderheiten von Datenbanken im allgemeinen und MySQL im besonderen eingegangen. Darüber gibt es im Internet jede Menge Informationen. Darüber hinaus wäre das ein Thema für mindestens einen Regalmeter Buch.

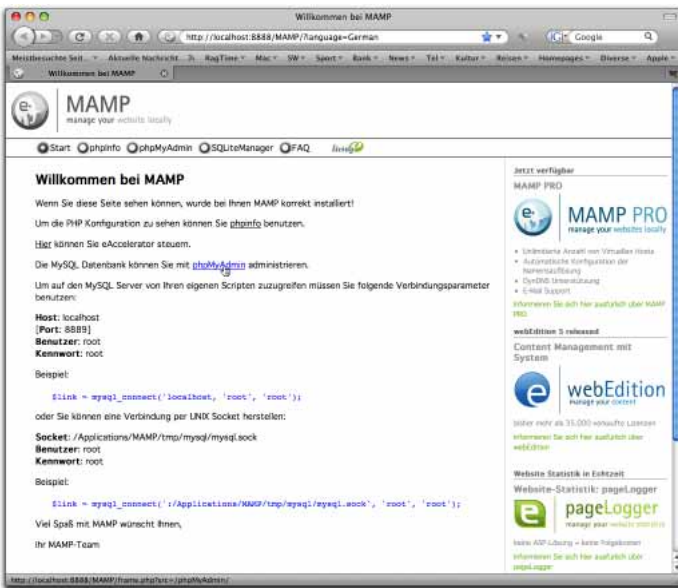
Diese Anleitung beschränkt sich darauf, ein kleines, ganz einfaches Beispiel zum Laufen zu bringen.

### *MAMP starten*

Als erstes muss sichergestellt sein, dass der in Mac OS X eingebaute Webserver nicht läuft. In den Systemeinstellungen muss also unter «Sharing» «Personal Web Sharing» ausgeschaltet sein.

Danach kann man MAMP starten. Wie das geht, steht in der Dokumentation.

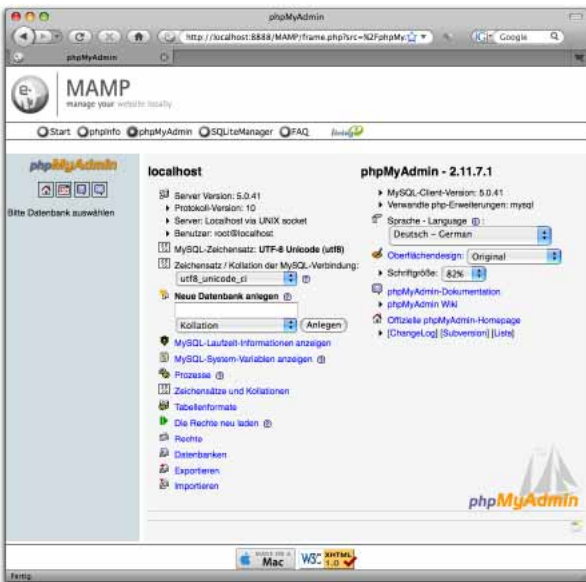
Starten Sie einen WebBrowser wie z. B. Safari. Geben Sie in die Adresszeile «localhost/MAMP» ein.



Es erscheint die Startseite der MAMP-Installation auf Ihrem Rechner.

### phpMyAdmin starten

Ein Klick auf «phpMyAdmin» startet die Datenbank-Verwaltung.

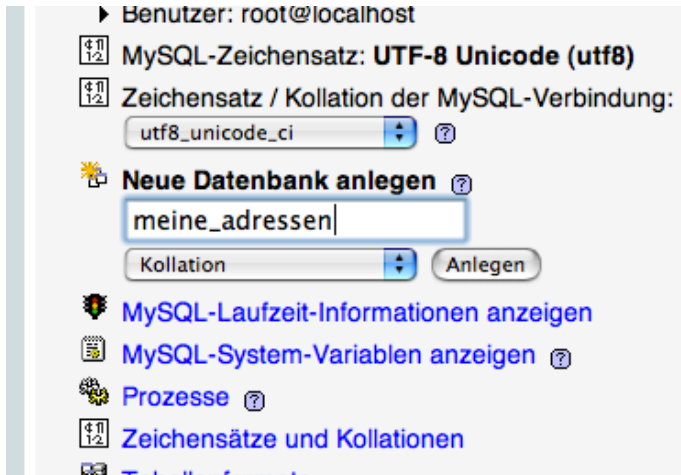


Falls die Texte nicht in deutscher Sprache erscheinen, kann man das in der rechten Spalte in einem Auswahlménü anpassen.

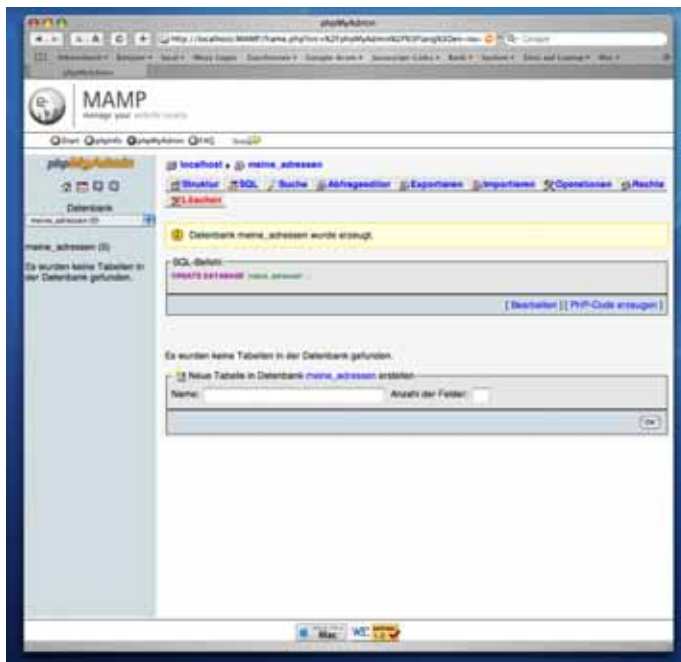
### Datenbank anlegen

Für unseren ersten Test legen wir eine Datenbank mit dem Namen «meine\_adressen» an. Das wird auf der Startseite von phpMyAdmin, die Sie gerade geöffnet haben, erledigt.

Bitte beachten Sie bei allen Namensgebungen in MySQL darauf, dass keine Leerzeichen und keine Sonderzeichen (Umlaute, Satzzeichen, mathematische Operatoren usw.) enthalten sind. MySQL beachtet bei allen Namen die Schreibweise. «meine\_adres- sen» ist also nicht gleich «Meine\_adressen»!



Geben Sie den Namen in das Textfeld unter «Neue Datenbank anlegen» ein und klicken Sie auf «Anlegen».



Es erscheint eine neue Seite. In der linken Spalte sehen Sie jetzt den Namen der neu angelegten Datenbank. Die Null in Klammern hinter dem Namen bedeutet, dass diese Datenbank keine Tabelle enthält.

### *Tabelle anlegen*

Jetzt haben Sie also eine Datenbank, aber noch keine Datentabelle. MySQL-Datenbanken können beliebig viele Tabellen enthalten. Für unsere Zwecke reicht eine. Sie wird den Namen «adressen» bekommen.

Es soll eine Adress-Datenbank werden. In die einzige Tabelle können Sie später Adressdaten eingeben. Vorgesehen sind Felder für Familienname, Vorname, Strasse, Hausnummer, PLZ, Ort, Telefon, Fax und E-Mail. Das sind also neun Felder. Hinzu kommt noch eines, mit dem jeder Datensatz eindeutig identifiziert werden kann: id.



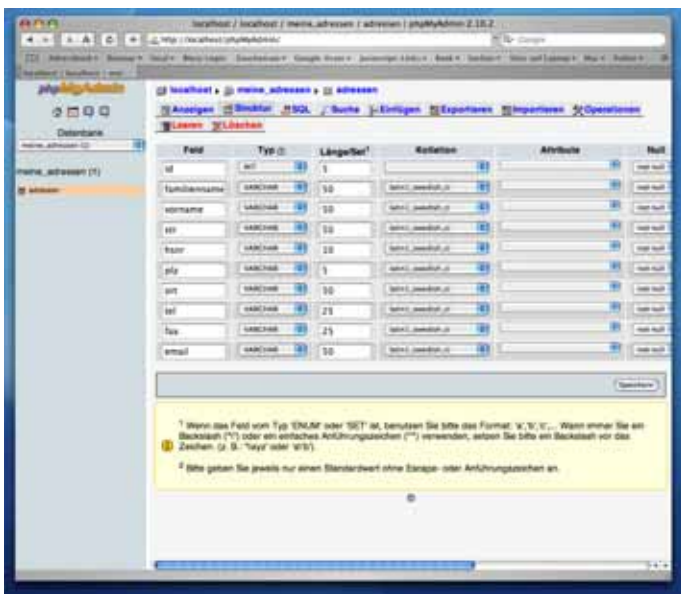
Geben Sie also den Tabellennamen («adressen»), die Anzahl Felder (10) ein und klicken Sie auf OK.



Auf der folgenden Seite werden die einzelnen Felder der Tabelle konfiguriert. Bitte geben Sie für das Beispiel die folgenden Daten ein (Beachten Sie dabei unbedingt die Schreibweise der Feldnamen):

Feld	Typ	Länge/Set	...	Extra	«U»
id	INT	5		auto_increment	•
familienname	varchar	50			
vorname	varchar	50			
str	varchar	50			
hsnr	varchar	10			
plz	varchar	5			
ort	varchar	50			
tel	varchar	25			
fax	varchar	25			
email	varchar	50			



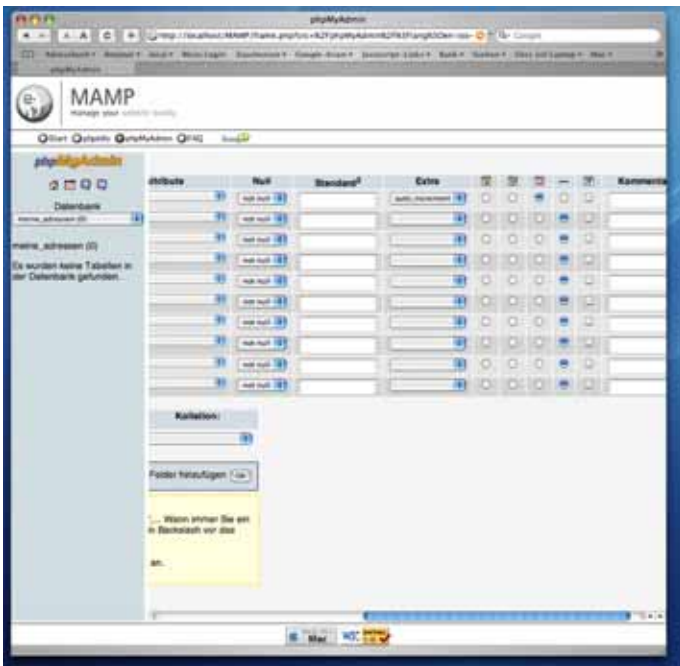


Füllen Sie das Formular wie im Bild bzw. in der Tabelle gezeigt aus. Sie sehen im Formular jeweils ein Feld für den Namen des Feldes, daneben ein Auswahlménú für den Datentyp und danach ein Textfeld, in dem die Länge der Daten eingetragen wird.

Das Feld «id» ist ein reines Zahlenfeld. Hier werden die Datensätze nummeriert. Für Ganzzahlen ist der Datentyp «INT» (steht für «integer») zuständig. Bei der eingeben Länge 5 sind maximal 99999 Datensätze möglich. Dabei muss man bedenken, dass die Nummer eines gelöschten Datensatzes nie wieder verwendet wird. Die maximal verwendete Nummer ist also unter Umständen sehr viel größer als die tatsächliche Anzahl der Datensätze.

Alle anderen Felder sind Textfelder. Dafür gibt es verschiedene Datentypen, die je nach erwarteter Textmenge verwendet werden. Der Datentyp «varchar» (für «various characters») ist für kleine Textmengen gedacht. Im Feld «Länge» gibt man jeweils an, wie lang der eingegebene Text maximal sein darf (Anzahl Zeichen). Werden längere Texte eingegeben, werden sie an der angegebenen Maximallänge kommentarlos abgeschnitten. Man kann maximal 256 Zeichen in ein solches Feld eintragen. Natürlich können Sie andere Feldlängen eintragen, z. B. wenn Sie internationale Postleitzahlen im Feld «PLZ» eintragen wollen.

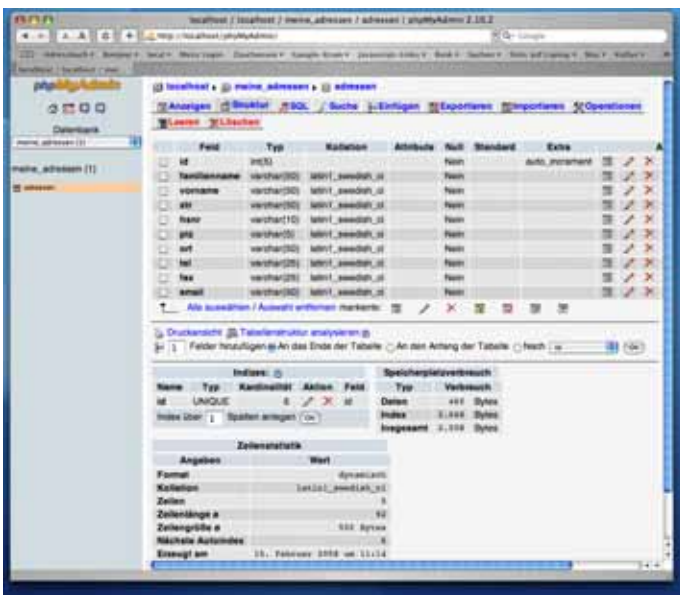
Damit das Feld «id» auch wie gewünscht funktioniert (Datensätze so nummerieren, dass jede Nummer nur einmal vorkommt), müssen hier noch zwei Einstellungen vorgenommen werden.



Wählen Sie für dieses Feld (und kein anderes) unter «Extras» «auto\_increment» (automatisches Hinaufzählen) und den Radioknopf unter dem Icon mit dem «U» (unique – einzigartig) aus. Die beiden Einstellungen befinden sich ziemlich weit rechts im Formular, also ggf. dorthin scrollen.

Klicken Sie auf «Speichern».

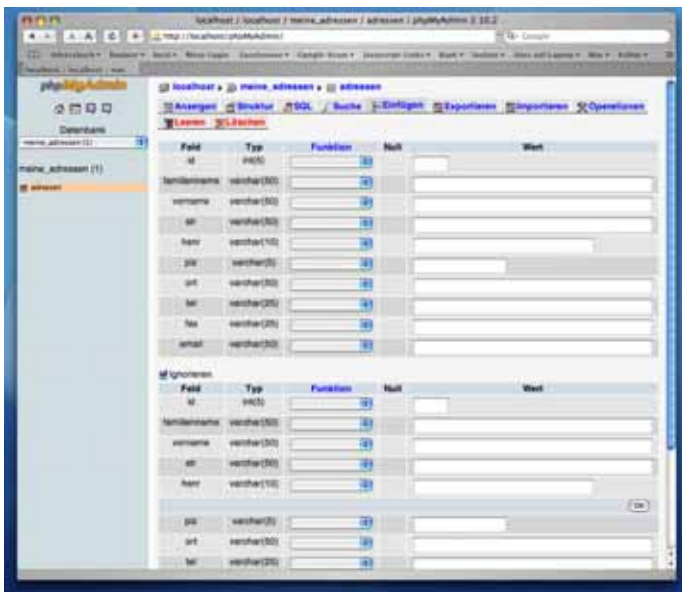
Erscheint eine neue Seite.



Sie können jetzt in der linken Spalte von phpMyAdmin unter dem Namen der Datenbank (jetzt mit einer 1 in Klammern) den Namen der neuen Tabelle sehen. Klicken Sie darauf. Sie sehen die oben abgebildete Seite.

## Daten eintragen

Zum Eintragen von Daten klicken Sie auf den Reiter «Einfügen». Es erscheint die folgende Seite:



Tragen Sie hier die Daten ein. Achten Sie darauf, dass das Feld «id» immer leer bleibt. Die Nummerierung der Datensätze nimmt MySQL automatisch vor. Da sollten Sie auf keinen Fall eingreifen.

Für das Beispiel können Sie z. B. folgende Daten eingeben. Sie müssen natürlich nicht die selben Daten verwenden. Für das Beispiel wäre es aber gut, wenn Sie mehrere Personen mit dem selben Familiennamen eintragen und auch hier und da die Felder email, fax oder tel leer lassen würden.

familienname	vorname	str	hsnr	plz	ort	tel	fax	email
Quirnbach	Benedikt	Musterstr.	13	32569	Musterstadt	02365 147 89 98	02365 147 89 99	bq@muster.de
Kaegi	Thomas	Musterstr.	45	32569	Musterstadt	02365 698 254 7	02365 698 254 8	tk@muster.ch
Müller	Luise	Luisenstr.	157 - 159	02365	Ostheim	0369 85 47	0369 58 71	l.mueller@muellerostheim.de
Müller	Karl-Heinz	Luisenstr.	157 - 159	02365	Ostheim	0369 85 47	0369 58 71	kh.mueller@muellerostheim.de
Schmitz	Fritz	Am Anger	5	98745	Dahinten	02587-147 85 23		
Menge	Marga	Feldweg	3	45896	Aufmdorf	03497-852 97 13		

# Eine ODBC-Datenquelle einrichten

Damit RagTime auf die Datenbank zugreifen kann, müssen zwei Vorkehrungen getroffen werden:

- Im System muss eine ODBC-Datenquelle registriert sein und
- im RagTime-Dokument muss eine RagTime-Connect-Komponente erzeugt werden.

Im Gegensatz zu früheren Versionen können Sie Datenquellen nur noch mit dem «ODBC-Administrator» einrichten, nicht mehr direkt aus RagTime!

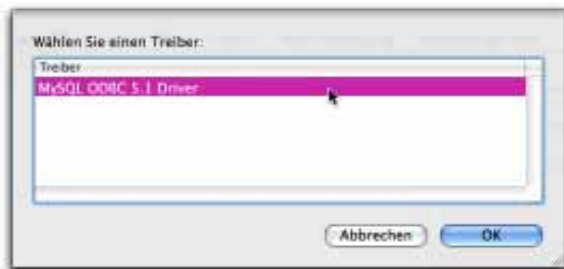
## ODBC-Datenquelle mit ODBC-Administrator einrichten

Das Dienstprogramm ODBC-Administrator starten und die Rubrik «Benutzer-DSN» wählen. Zur Erinnerung: Dieses Programm von Apple ist sehr instabil. Verzweifeln Sie nicht und machen Sie einfach nochmals einen Versuch, wenn das Programm abstürzen sollte.

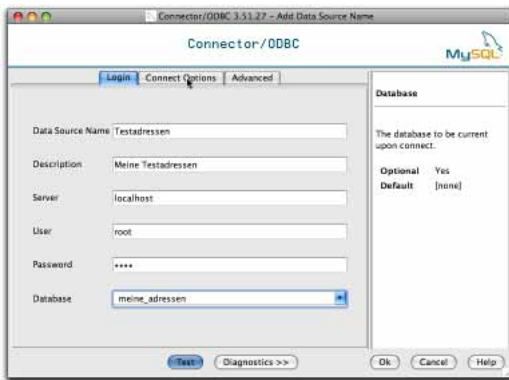


Links unten auf das Schloss klicken, um Änderungen zu ermöglichen und mit dem Administrator-Passwort bestätigen, dass man dazu berechtigt ist. – Die Tabelle in obigem Bild kann leer sein, oder aber bereits Einträge enthalten.

Auf «Hinzufügen» klicken, im Unterfenster den aktuellen MySQL-Treiber auswählen und auf OK klicken:

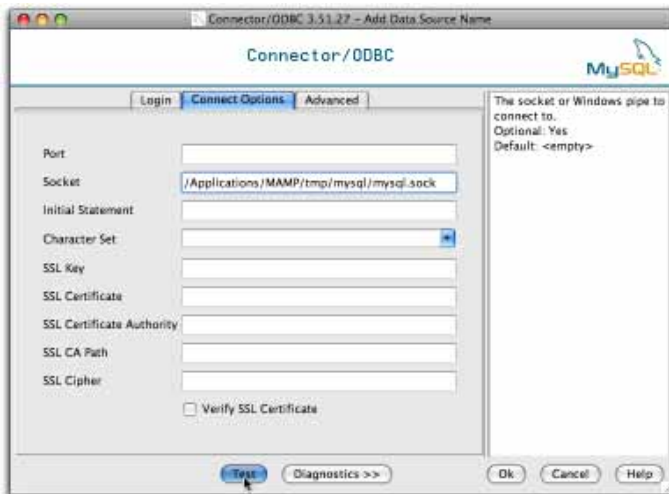


Im nächsten Unterfenster müssen verschiedene Eingaben gemacht werden, die den Zugriff zur Datenbank definieren. Füllen Sie in der Rubrik «Login» das Formular wie folgt aus:



- Für «Data Source Name» geben Sie der Datenbank einen beschreibenden Namen. Hier wurde «Testadressen» gewählt.
- Unter «Description» können Sie mit beliebigen Worten eine erklärende Beschreibung dazufügen.
- Für «Server» müssen Sie «localhost» eingeben.
- Für «User» und «Password» geben Sie zwei Mal «root» ein.
- Unter «Database» müssen Sie genau den Namen der Datenbank eingeben, wie Sie ihn mit «phpMyAdmin» definiert haben, hier also «meine\_adressen».

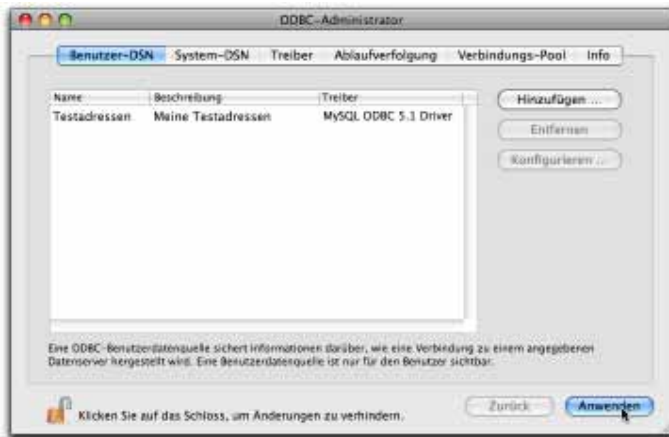
Dann klicken Sie auf den Reiter «Connect Options» und machen für «Socket» genau folgende Eingabe: /Applications/MAMP/tmp/mysql/mysql.sock



Zum Schluss klicken Sie auf «Test» und erhalten dann hoffentlich folgende Erfolgsmeldung:



die Sie mit Klick auf «OK» quittieren. Sollte stattdessen eine Fehlermeldung angezeigt werden so überprüfen Sie Ihre Eingaben und zudem im Fenster von MAMP, ob die Server auch laufen. Mit Klick auf «Diagnostics >>» können Sie möglicherweise mehr über die Fehlerursache herausfinden. Wenn alles in Ordnung ist, schliessen Sie die Eingaben mit Klick auf «Ok» ab. Leider scheint das für den «ODBC-Administrator» ein besonders kritischer Vorgang zu sein, denn gerade in diesem Moment stürzt er oft ab, offenbar aber in vielen Fällen erst nachdem der Eintrag erfolgt ist, was Sie nach dem Neustart des Programms feststellen können. Andernfalls müssen Sie die Eingaben wiederholen.



Wirklich übernommen werden die Eingaben aber erst mit Klick auf «Anwenden». Dieser Klick ist zwingend! Sonst kann der neue Eintrag ohne Vorwarnung beim Klick auf das Schloss gleich wieder verschwinden und die ganze Mühe war umsonst! Zum Schluss das Schloss wieder verriegeln und das Fenster schließen.

## *RagTime Connect*

### *RagTime-Connect-Komponente in RagTime erzeugen*

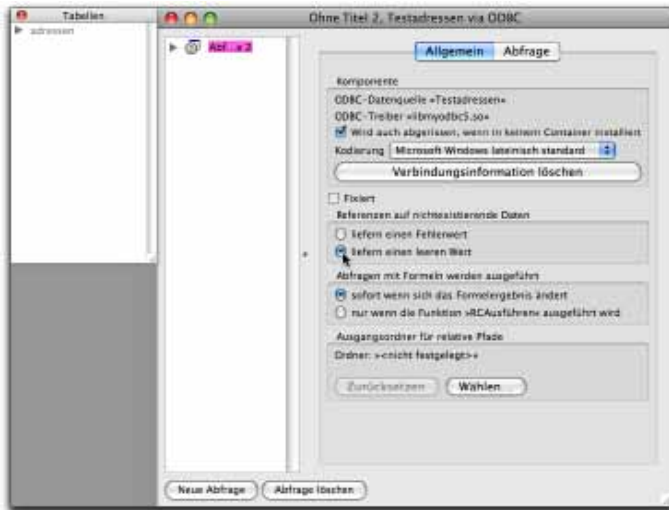
Jetzt öffnen wir ein neues RagTime Dokument und erzeugen eine neue RagTime-Connect-Komponente (Fenster → Neue Komponente → RagTime Connect).

Es erscheint ein Fenster zur Auswahl einer Datenbank.



Wählen Sie die eben angelegte Datenquelle aus und klicken Sie dann auf «OK».

Zunächst sollten Sie im neuen Fenster die Rubrik «Allgemein» wählen und dort zwei Einstellungen vornehmen:



Da MySQL ein plattformübergreifender Datenbank-Server ist, muss er mit vielen verschiedenen Betriebssystemen auskommen. Dabei gibt es immer wieder Probleme mit der Zeichenkodierung. Wahrscheinlich muss auch bei Ihnen die Kodierung auf «Microsoft Windows lateinisch standard» eingestellt werden, damit Umlaute und Sonderzeichen aus der Datenbank richtig ins RagTime-Dokument übernommen werden.

Die zweite Einstellung betrifft die Fälle, wo Datenfelder leer sind. Man bekommt dann entweder einen Fehlerwert oder nichts. Stellen Sie hier «liefern einen leeren Wert» (also nichts) ein.

## *Erste Abfrage formulieren*

MySQL wird mit der Sprache SQL (structured query language) betrieben. Hier wird keine Einführung in SQL gegeben. Darüber gibt es viele Bücher und auch Tutorials im Internet.

### *Alle Daten aus der Tabelle holen*

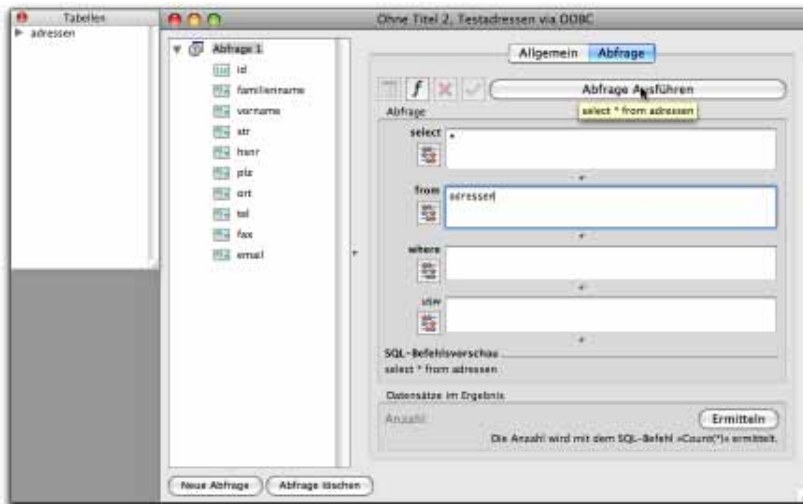
Als erste einfache Abfrage holen wir alle Daten aus der Datenbank. Dafür wechseln Sie zur Rubrik «Abfrage». Hier wird mit ein paar Eingabefeldern die SQL-Syntax schon vorgegeben.

Die Abfrage lautet in SQL: `select * from adressen`

Das heißt: Wähle alles (\*) aus der Tabelle «adressen»



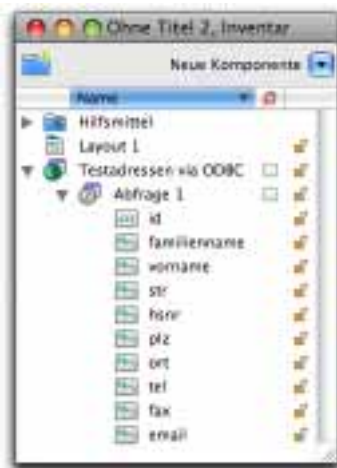
In RagTime Connect geben Sie dementsprechend in das Feld «select» ein Sternchen ein. Klicken Sie in das Feld «from» und dann in der Palette «Tabellen» auf den Namen der Tabelle. (Natürlich könnten Sie den Namen der Tabelle auch selbst eintippen...).



Wahrscheinlich müssen Sie das Fenster durch Ziehen an der Ecke unten rechts etwas breiter machen, damit die Feldnamen der Datenbank vollständig angezeigt werden. Dann klicken Sie auf «Abfrage ausführen» und in der linken Spalte erscheinen alle Datenfelder der Tabelle. Ein Klick auf «Ermitteln» zeigt an, wie viele Datensätze durch diese Abfrage geliefert werden, für unser Beispiel also 6.

### *RagTime-Connect im Inventar*

Danach können Sie das Fenster der Komponente schließen. Im Inventar sieht das dann so aus:



Wie mit der Komponente umzugehen ist steht im Handbuch von RagTime Connect. Im Folgenden kommt aber noch ein kurzes Beispiel.

# Die Adresdaten in einem Brief-Formular nutzen

Wenn Sie bis hierher der Anleitung gefolgt sind, haben Sie jetzt ein RagTime-Dokument mit einer RagTime-Connect-Komponente.

In dieser Anleitung wird keine Einführung in die Benutzung von RagTime-Connect geben. Das wäre sicherlich eine weitere Anleitung wert. An einem einfachen Beispiel wird hier kurz beschrieben, wie Sie am einfachsten die gerade angelegte Adresdatenbank benutzen können.

## Empfänger-Adresse anlegen

Ziehen Sie im Layout des Dokumentes einen Container auf und stellen seine Inhaltsart auf Text. Klicken Sie hinein und rufen Sie die Formelpalette auf. Klicken Sie in die Formelpalette und geben Sie die folgende Formel ein (aber ohne Zeilenumbruch).

Wie geben Sie solche Formeln ein?

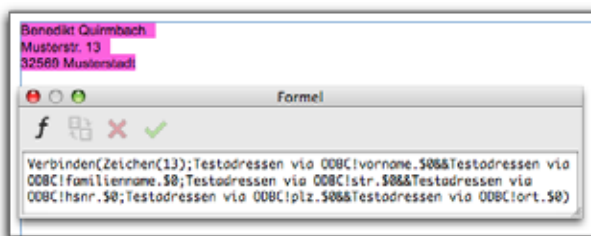
Die Formel sieht recht kompliziert aus, ist aber relativ leicht einzugeben:

Mit der blinkenden Einfügemarke in der Formelpalette rufen Sie mit Klick auf «f» die Funktionentabelle auf. Wählen Sie «Alle Funktionen». Wenn Sie nun auf der Tastatur «ve» eingeben, wird direkt die Funktion «Verbinden» ausgewählt. Markieren Sie die Option «Mit Argumenten einsetzen» und klicken Sie auf «OK» (oder betätigen die RETURN- oder ENTER-Taste). Nun ist «Verbindungstext ausgewählt. Sie können «Zeichen(13)» ganz eintippen oder dabei die Funktion mit der eben beschriebenen Methode einfügen. «Zeichen(13) steht für «Zeilende/RETURN», ein Zeichen welches nicht direkt, sondern nur auf diese Weise in die Formel eingegeben werden kann.

Wählen Sie nun das zweite Argument (Textliste) durch Doppelklick aus. Klicken Sie dann im Inventar auf den Feldnamen «vornamen» und in der richtigen Reihenfolge die übrigen Feldnamen in der RagTime-Connect-Komponente. Dazwischen fügen Sie je entweder ein Semikolon oder «&&» ein. Pro memoria: «&&» fügt zwei Elemente mit einem eingeschobenen Leerzeichen zusammen – überflüssige Leerzeichen werden dabei unterdrückt.

```
Verbinden(Zeichen(13);  
Testadressen via ODBC!vorname.$0 && Testadressen via ODBC!familienname.$0;  
Testadressen via ODBC!str.$0 && Testadressen via ODBC!hsnr.$0;  
Testadressen via ODBC!plz.$0 && Testadressen via ODBC!ort.$0)
```

Nach dem Klick aufs grüne Auslösehäkchen sollte Ihr Dokument so aussehen – ggf. müssen Sie in der Vorlage «Standardabsatz», Rubrik «Ränder», den «Abstand unten» noch auf «Einzeilig» einstellen:



Was bedeutet «\$0»?

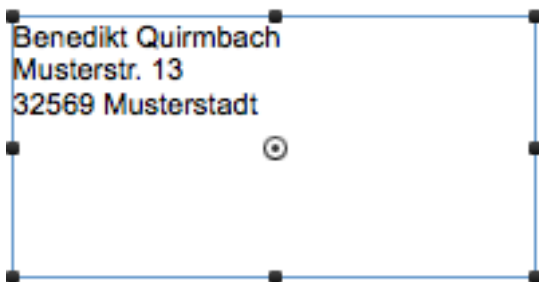
Die einzelnen Datenfelder werden - wenn Sie so vorgehen, wie beschrieben - mit «\$0» referiert: «Testadressen via ODBC!vorname.\$0».

Am Ende der Referenz steht immer die Nummer des Datensatzes (nicht zu verwechseln mit der «id» im Datensatz!). «\$0» ist in RagTime-Connect eine besondere Referenz. Es handelt sich dabei um den Datensatz des Abfrage-Ergebnisses, der gerade ausgewählt ist. Sie können durch das Abfrage-Ergebnis «blättern», also einen Datensatz nach dem anderen aufrufen, sofern die Abfrage mehr als einen Datensatz liefert. «\$0» ist immer eine Referenz auf den Datensatz, der gerade «aufgeblättert» ist.

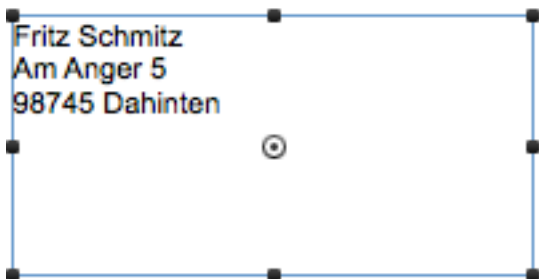
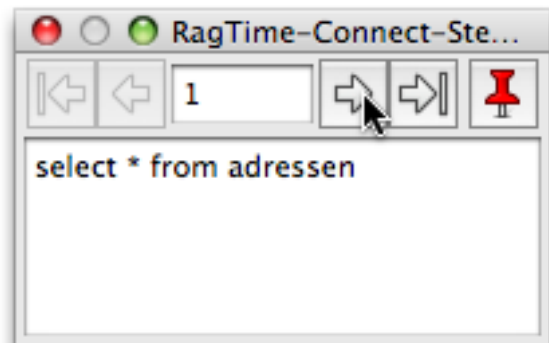
### *Durch die Datensätze «blättern»*

Bei der Beispiel-Abfrage werden alle Datensätze aus der Adresstabelle geholt. Es liegt also ein Ergebnis mit 6 Datensätzen vor. Im Adressfeld soll aber natürlich immer nur eine Adresse angezeigt werden. Das ist zunächst durch die \$0-Referenz in der Formel die erste Adresse im Abfrage-Ergebnis.

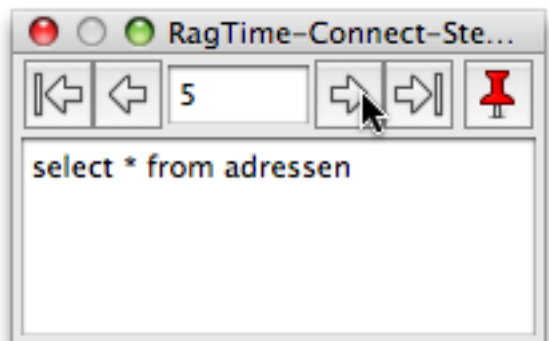
Mit Hilfe der Palette RagTime-Connect-Steuerung (Fenster → Paletten → RagTime-Connect-Steuerung) können Sie durch das Abfrage-Ergebnis bättern bis zu der Adresse, die Sie für einen Brief benötigen.



Benedikt Quirnbach  
Musterstr. 13  
32569 Musterstadt



Fritz Schmitz  
Am Anger 5  
98745 Dahinten



Speichern und schließen Sie das Brief-Formular als RagTime-Formular.

### *Brief-Formular benutzen*

Öffnen Sie das gerade gespeicherte Formular.

### *Fixierung der benutzten Daten*

Beachten Sie, dass die Auswahl des aktuellen Datensatzes im Dokument fixiert ist. In RagTime-Dokumenten wird die aktuelle Datenbank-Auswahl beim Schließen fixiert. Das heißt: Beim nächsten Öffnen befinden sich genau die gleichen Texte im Dokument wie beim Schließen. Das trifft auch zu, wenn sich die entsprechenden Datensätze in der Datenbank zwischenzeitlich geändert haben sollten oder die Datenbank nicht erreichbar ist (ist gelöscht oder der Server ist nicht erreichbar).

Den Sinn dieses Verhaltens lässt sich am einfachsten anhand eines Beispiels erklären:

Stellen Sie sich vor, Sie haben eine Rechnung geschrieben, die die Preise der einzelnen Posten aus einer Datenbank holt. Sie speichern die Rechnung zur Archivierung. Wenn Sie später die Rechnung ansehen, müssen darin genau die selben Preise stehen wie bei Rechnungsstellung. Die Preisangaben in der Datenbank könnten sich aber zwischenzeitlich geändert haben. Dasselbe trifft natürlich auf alle möglichen Daten zu, wie z.B. auch Adressdaten.

Außer durch das Sichern können Sie die Daten auch bei noch geöffnetem Dokument z.B. in der Palette RagTime-Connect-Steuerung durch Klick auf das rote Nadel-Symbol fixieren.

### *Fixierung aufheben*

Wenn Sie dieses Dokument wieder öffnen, ist die Auswahl des Datensatzes fixiert. Sie können also ohne weitere Vorkehrungen keinen anderen Datensatz auswählen.

Wenn Sie hier einen neuen Datensatz auswählen möchten, müssen Sie erst die Fixierung aufheben. Das können Sie entweder in der RagTime-Connect-Steuerung oder indem Sie die Komponente im Inventar doppelklicken – also öffnen – und dann in der Tafel «Allgemein» das Häkchen bei «Fixiert» wegklicken.

Öffnen Sie die RagTime-Connect-Steuerung (Fenster → Paletten → RagTime-Connect-Steuerung).

Blättern Sie zu einer neuen Adresse.

Sichern Sie das Dokument.

# Ein Beispiel mit dynamischem Datenbank-Zugriff

In obigem Beispiel wurde jeweils einfach der aktuelle Datensatz aus der Datenbank ins RagTime-Dokument geholt. Die Auswahl des anzuzeigenden Datensatzes, bzw. der daraus gebildeten Adresse, erfolgte mit Blättern in der RagTime-Connect-Steuerungspalette. In einem zweiten Beispiel wird nachfolgend gezeigt, wie die Auswahl der Adresse auch mit Eingaben im RagTime-Dokument selbst erfolgen kann. Das Dokument soll die Wahl von Adressen aufgrund des Ortsnamens, des Anfangs des Vor- oder Familiennamens, oder einer beliebigen Kombination dieser Auswahlkriterien ermöglichen.

## Den Ort aus der Liste der vorkommenden Orte auswählen

Legen Sie ein neues RagTime-Dokument an und löschen Sie dessen «Ur-Rahmen», bevor Sie es unter dem Namen «ODBC\_Test» speichern. Dann legen Sie mit «Fenster → Neue Komponente → RagTime Connect» eine RagTime-Connect-Komponente an. Im Fenster «Datenbank auswählen» doppelklicken Sie die vorher angelegte Datenquelle «Testadressen» und legen damit die «Abfrage 1» an, die Sie gleich zu «Ortswahl» umbenennen. Ziel ist ein Einblendmenü mit allen vorkommenden Ortsnamen, mit dem der zu suchende Ortsname ausgewählt werden kann.

Wählen Sie für die eben erstellte Abfrage in der Rubrik «Allgemein» unter «Komponente → Kodierung» die Option «Microsoft Windows lateinisch standard» und unter «Referenzen auf nichtexistierende Daten» die Option «liefern einen leeren Wert». Die Abfrage definieren Sie wie folgt:

```
select          DISTINCT ort
from            adressen
where           (leer lassen)
usw            ORDER BY ort
```

und klicken Sie dann auf «Abfrage ausführen». Wenn Sie die Datenbank so eingegeben haben wie vorgeschlagen, dann liefert diese Abfrage 4 Datensätze. Fälschlicherweise wird unten im Fenster trotzdem die Anzahl mit 6 angegeben! Das lässt sich laut RagTime nicht vermeiden, da die SQL-Anfrage nach der Anzahl der gefundenen Datensätze von den verschiedenen Datenbanken/Treibern nicht einheitlich verarbeitet werde. Die SELECT COUNT(\*)-Technik sei die einzige, die mit jedem SQL-Server funktioniere. Und dabei wird eben die DISTINCT-Einschränkung unter SELECT leider ignoriert.

Legen Sie mit «Fenster → Neue Komponente» eine weitere neue Komponente an, diesmal ein Rechenblatt. Sie können das Fenster durch Ziehen an der rechten unteren Ecke so verkleinern, dass Sie nur noch die ersten ca. 10 Zeilen von Spalte A sehen. In Zelle A1 geben Sie ein Leerzeichen ein. Wählen Sie dann Zelle A2 und öffnen Sie die Formelpalette. Klicken Sie ins Formelfeld, so dass der Cursor dort blinkt, und dann im Inventar auf das Element «ort» in der RagTime-Connect-Komponente. Automatisch wird folgender Code erzeugt:

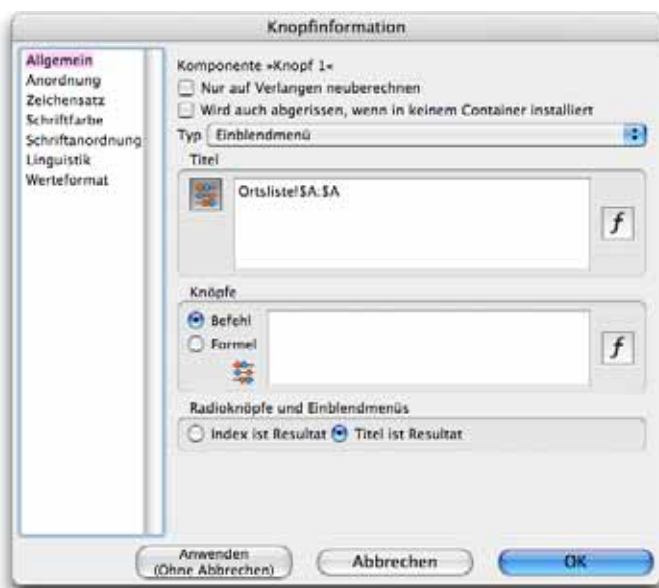
```
Testadressen via ODBC!ort.$0
```

Ersetzen Sie die beiden letzten Zeichen («\$0») durch eine «1» und schliessen Sie die Formeleingabe ab. Ziehen Sie die Formel bis zur Zelle A6 nach unten. Dann sieht Ihr Rechenblatt so aus:



Die Referenz zum gefundenen Datensatz wurde beim Ziehen nach unten automatisch angepasst. Zeile 6 bleibt dank der Einstellung in der Tafel «Allgemein» der Abfrage leer. Es macht also nichts, wenn Sie z.B. bis Zeile 100 gezogen haben, um sicher zu sein, dass auch nach Eingabe weiterer Adressen in Ihrer Datenbank alle verschiedenen Ortsnamen angezeigt werden. Bei einer sehr grossen Zahl verschiedener Ortsnamen müsste ein mehrstufiges Verfahren angewendet werden, z. B. nur Ortsnamen mit einem bestimmten Anfangsbuchstaben oder in einem bestimmten Postleitzahlenbereich. Hier wird nur die einstufige Lösung betrachtet. – Nennen Sie das Rechenblatt im Inventar «Ortsliste». Ordnen Sie das Fenster mit der Komponente «Ortsliste» neben dem Layoutfenster an.

Wechseln Sie ins Fenster «Layout 1» und ziehen Sie einen Rechteck-Rahmen mit Inhalt «Knopf» auf, etwa 4 cm breit und 0,7 cm hoch. Öffnen Sie die Knopfinformation, ändern Sie in der Rubrik «Allgemein» den Typ auf «Einblendmenü» und klicken Sie unter «Titel» auf das Abakus-Symbol für Formeleingabe. Wählen Sie alles aus, was bereits im Formelfeld steht und klicken Sie auf den Spaltenkopf «A» im Rechenblatt «Ortswahl». Schliesslich wählen Sie ganz unten noch die Option «Titel ist Resultat».



Dann schliessen Sie die Knopfinformation mit Klick auf «OK» und nennen den Knopf im Inventar «Ortsauswahl».

Ziehen Sie im Layout einen Rechteckcontainer mit Inhalt «Rechenblatt» ca. 12 cm breit und 7 cm hoch auf. Nennen Sie das Rechenblatt im Inventar «Auswahl». Geben Sie in den Zellen A1:A3 folgende Texte ein: «Familienname beginnt mit:», «Vorname beginnt mit:» und «Ort ist:». Doppelklicken Sie auf den Spaltenkopf «A» um die Spaltenbreite zu optimieren und wählen Sie mit ausgewählter Spalte «Format → Ausrichtung → Rechtsbündig». Ziehen Sie den Knopf aus dem Inventar in Zelle B3 und doppelklicken Sie auch hier auf den Spaltenkopf, damit Sie den Knopf auch sehen können. Nun können Sie den Container mit dem Knopf im Layout löschen. Im Rechenblatt «Auswahl» sind nun die Eingabefelder für die Auswahl definiert. Ergänzen Sie das Rechenblatt noch mit den Texten gemäss folgendem Bild:

1	Familienname beginnt mit:	
2	Vorname beginnt mit:	
3	Ort ist:	▼
4		
5	Anzahl Adressen:	
6		
7	Auswahl Familienname:	
8	Auswahl Vorname:	
9	Auswahl Ort:	
10		
11	Auswahlformel:	

### *Eine selektive Abfrage definieren*

Klicken Sie im Fenster der RagTime-Connect-Komponente links unten auf «Neue Abfrage» und nennen Sie diese «Adressen». Natürlich ändern Sie wieder die Einstellung für die Kodierung und die Leeranzeige bei nichtexistierenden Daten. Die Abfrage definieren Sie wie folgt:

```
select      *
from        adressen
where       Auswahl!$B$11 (Abakus anklicken und dann im Rechenblatt
                    «Auswahl» auf die Zelle B11 klicken)
usw        ORDER BY familienname, vorname, ort
```

Wenn Sie jetzt rechts oben auf «Abfrage ausführen» klicken, dann werden links alle Feldnamen der Tabelle «adressen» unserer Datenbank angezeigt und unten wird angezeigt, dass alle 6 Datensätze ausgewählt wurden. Kunststück – es sind ja noch gar keine Auswahlkriterien eingegeben.

Eine kleine Zwischenbemerkung: Als Sie die Formeln im Rechenblatt «Ortsliste» eingegeben haben, existierte erst die eine Abfrage in der RagTime-Connect-Komponente. Wenn Sie sich jetzt die Formel in Zelle A2 dieses Rechenblattes nochmals anschauen, so sehen Sie, dass mit der Definition einer zweiten Abfrage automatisch der Name der ersten Abfrage in die Formel eingefügt wurde. Diese lautet deshalb jetzt:

```
Testadressen via ODBC!Ortswahl!ort.1
```

### *Die SQL-Formel für die selektive Abfrage zusammenstellen*

NB: Wegen der Verwendung der doppelten Anführungszeichen in der DE-CH-Version von RagTime sind die nachfolgend angegebenen Formeln für die DE und DE-CH-Versio-

nen unterschiedlich. Es wird immer zuerst die DE-Version angegeben, gefolgt von der DE-CH-Version!

In Zelle B11 des Rechenblatts «Auswahl» soll die Auswahlformel als SQL-Formel zusammengestellt werden. Es wird hier nicht auf die SQL-Syntax eingegangen. Akzeptieren Sie einfach, dass die Formel so aussehen muss wie nachfolgend angegeben. In einem Zwischenschritt sollen zuerst die Teilformeln für Familien- und Vorname sowie Ort erstellt werden. Im Rechenblatt «Auswahl» wählen Sie Zelle B7 und öffnen die Formelpalette. Als Formel geben Sie ein:

```
DE: Wenn(IstLeer(B1);'';'famiIienname LIKE "'&B1&'%"')
```

```
DE-CH: Wenn(IstLeer(B1);"";"famiIienname LIKE """&B1&"%"")
```

Kopieren Sie die Formel in Zelle B8 und ändern Sie in der dortigen Formel lediglich «famiIienname» zu «vorname».

In Zelle B9 geben Sie folgende Formel ein, wobei die Referenzen auf den Knopf mit Klick auf denselben eingefügt werden können:

```
DE: Wenn(Ortsauswahl!X=' ';'';'ort = "'&Ortsauswahl!X&'")
```

```
DE-CH: Wenn(Ortsauswahl!X=" ";"";"ort = """&Ortsauswahl!X&"")
```

Wenn Sie jetzt in die Felder B1 oder B2 eine beliebige Buchstabenkombination eingeben, oder mit dem Knopf einen Ort auswählen, so sehen Sie in Zelle B7, B8 oder B9 den SQL-Befehl, der daraus gebildet wird. Nun müssen diese Befehle nur noch in Zelle B11 kombiniert werden. Geben Sie dort die folgende Formel ein:

```
DE: Verbinden(' AND ';B7;B8;B9)
```

```
DE-CH: Verbinden(" AND ";B7;B8;B9)
```

Jetzt können Sie wieder mit Auswahlkriterien in den Eingabefeldern experimentieren und sehen dann, wie die kombinierte Formel lautet.

In Zelle B5 geben Sie schliesslich noch die nachfolgende Formel ein und richten den Zellinhalt linksbündig aus:

```
DE: RCAnzahlDatensätze('Testadressen via ODBC!Adressen')
```

```
DE-CH: RCAnzahlDatensätze("Testadressen via ODBC!Adressen")
```

Zur Kontrolle geben Sie für einen grafischen Text im Layout noch die Formel ein, mit der die Adresse für den gewählten Datensatz, bzw. den ersten von mehreren Datensätzen gebildet wird – analog zum ersten Beispiel (diese Formel ist in beiden Sprachversionen identisch – ohne Zeilenumbrüche eingeben!):

```
Verbinden(Zeichen(13);  
Testadressen via ODBC!Adressen!vorname.$0&&  
Testadressen via ODBC!Adressen!famiIienname.$0;  
Testadressen via ODBC!Adressen!str.$0&&  
Testadressen via ODBC!Adressen!hsnr.$0;  
Testadressen via ODBC!Adressen!plz.$0&&  
Testadressen via ODBC!Adressen!ort.$0)
```

Falls durch RagTime-Connect mit Ihren Auswahlkriterien mehrere Datensätze gefunden werden, was Sie sich ja in der Zelle B5 anzeigen lassen, so müssen Sie mit der RagTime-Connect-Steuerungspalette in den gefundenen Adressen blättern.



Familienname beginnt mit:				Karl-Heinz Müller
Vorname beginnt mit:				Luisenstr. 157-159
Ort ist:	Ostheim			02365 Ostheim
Anzahl Adressen:	2			
Auswahl Familienname:				
Auswahl Vorname:				
Auswahl Ort:	ort = "Ostheim"			
Auswahlformel:	ort = "Ostheim"			

Familienname beginnt mit:				Luise Müller
Vorname beginnt mit:				Luisenstr. 157-159
Ort ist:	Ostheim			02365 Ostheim
Anzahl Adressen:	2			
Auswahl Familienname:				
Auswahl Vorname:				
Auswahl Ort:	ort = "Ostheim"			
Auswahlformel:	ort = "Ostheim"			

RagTime-Connect-Ste...

Adressen

← 2 →

select \* from adressen where ort = "Ostheim" ORDER BY familienname, vorname, ort

Familienname beginnt mit:				Luise Müller
Vorname beginnt mit:	lu			Luisenstr. 157-159
Ort ist:	Ostheim			02365 Ostheim
Anzahl Adressen:	1			
Auswahl Familienname:				
Auswahl Vorname:	vorname LIKE "lu%"			
Auswahl Ort:	ort = "Ostheim"			
Auswahlformel:	vorname LIKE "lu%" AND ort = "Ostheim"			

Natürlich müsste an diesem Dokument kosmetisch noch einiges verbessert werden. Insbesondere müsste das Rechenblatt «Auswahl» im Layout als nichtdruckend deklariert werden. Aber diese Aspekte haben mit dem Gegenstand dieser Anleitung nichts zu tun und die Anforderungen variieren von Anwendung zu Anwendung.

Damit sollten Sie den Einstieg in die Benutzung von RagTime-Connect gefunden haben. Alles Weitere müssen Sie sich selbst erarbeiten. Viel Erfolg!

Falls Sie an dieser Anleitung Fehler oder Mängel entdecken, so bitte ich Sie um Rückmeldung. Meine Adresse finden Sie im RagTime-Forum.

Solothurn, 12.3.2009  
Thomas Kaegi