

Zugriff auf ein MySQL-Backup mittels Lazarus und XAMPP

(C) Heiko Rompel 21. Juli 2023

1. Vorgeschichte

Ich stand vor der Aufgabe, lokal auf ein Backup einer SQL-Datenbank zuzugreifen.

2. Was wird benötigt?

- Ich brauchte unter Linux also erst einmal einen lokalen SQL-Server.
 - Ich entschied mich für das XAMPP-Paket (<https://www.apachefriends.org/de/index.html>).



XAMPP Apache + MariaDB + PHP + Perl

Was ist XAMPP?

XAMPP ist die beliebteste PHP-Entwicklungsumgebung

XAMPP ist eine vollständig kostenlose, leicht zu installierende Apache-Distribution, die MariaDB, PHP und Perl enthält. Das XAMPP Open-Source-Paket wurde für eine extrem einfache Installation und Nutzung eingerichtet.


Herunterladen
Klicken Sie hier für weitere Versi

XAMPP für Windows
8.2.4 (PHP 8.2.4)

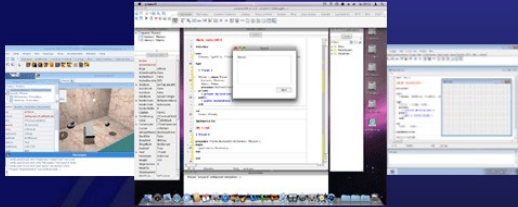
XAMPP für Linux
8.2.4 (PHP 8.2.4)

XAMPP für OS X
8.2.4 (PHP 8.2.4)

- Um die die Daten der SQL-Datenbank hinterher weiter verarbeiten zu können, brauchte ich auch noch eine Programmiersprache.
 - Ich entschied mich für Lazarus.

 **Lazarus**

HOME ABOUT SCREENSHOTS FAQ FEATURES DOWNLOADS FORUM WIKI



Lazarus
The professional Free Pascal RAD IDE

- ☐ Cross platform
- ☐ Drag & Drop Form Designer
- ☐ Open source (GPL/LGPL)
- ☐ Delphi converter

[Download Now](#)

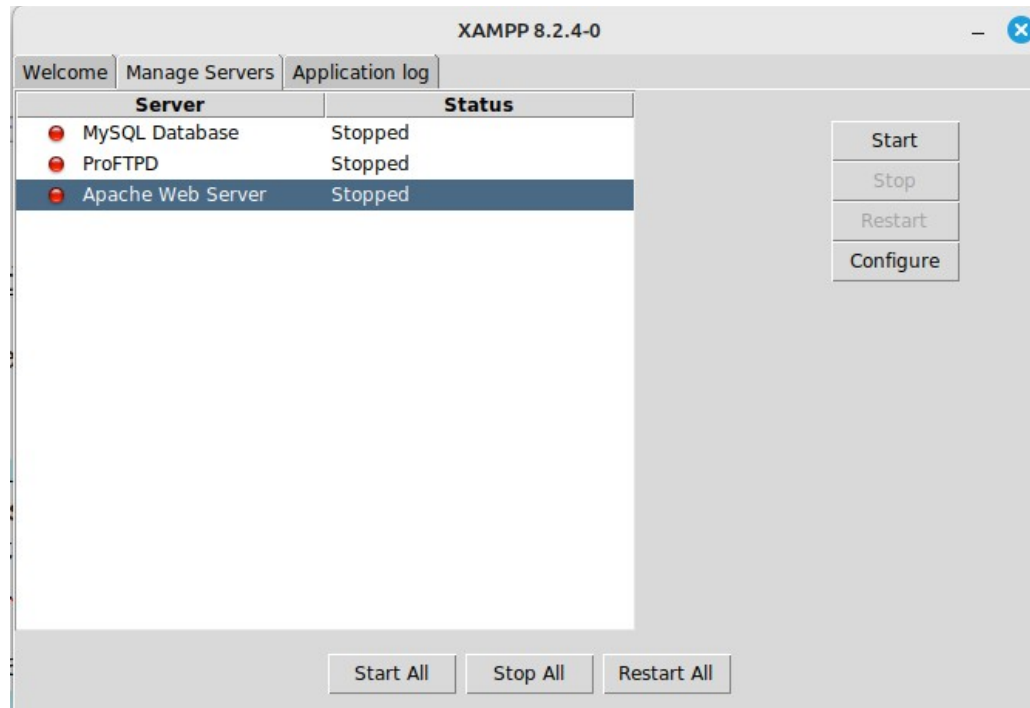
Version 2.2.6 for Debian 64 bit Try DEB? | Other ▼

3. XAMPP und die Datenbank einrichten

Für hier gestellte Aufgabe, ist die Einrichtung von XAMPP eigentlich nicht sonderlich aufwendig.

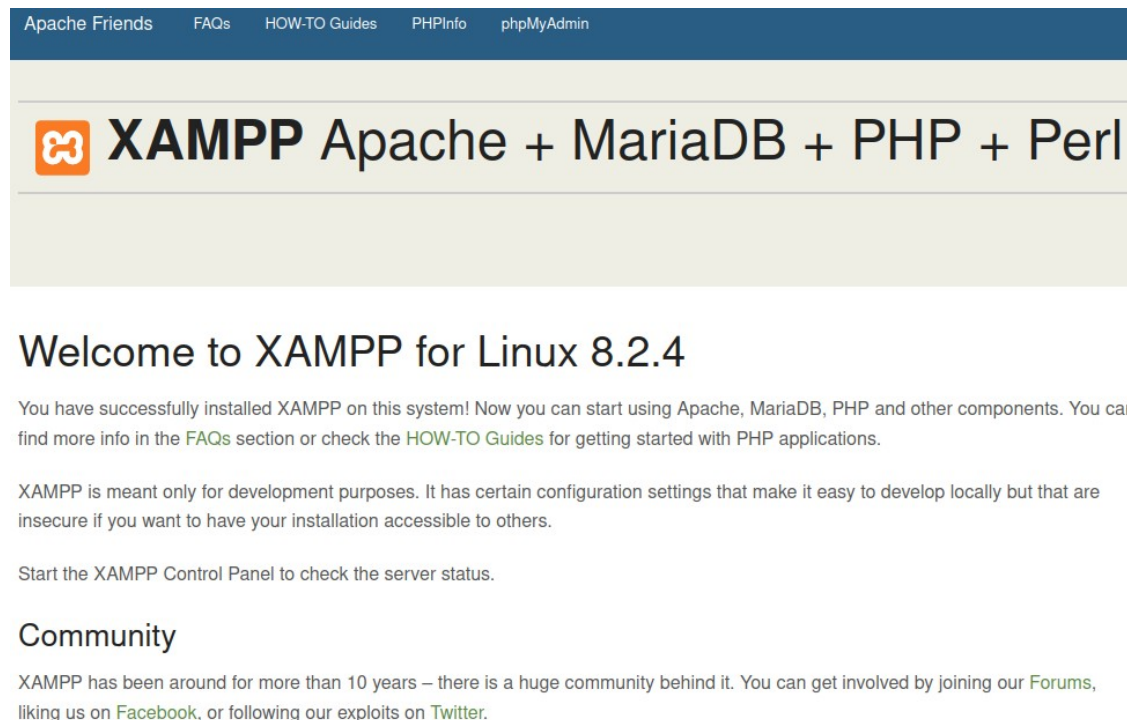
- XAMPP für das entsprechende Betriebssystem herunterladen.
- Das Paket installieren.

- XAMPP starten. Je nach Betriebssystem und Distro, wird bei der Installation das „XAMPP-Panel“ schon mit eingerichtet.



Wenn nicht, muss man den Apache und die MariaDB via Kommandozeile starten.

- `sudo /opt/lampp/lampp start`
- `sudo /opt/lampp/lampp stop`
- Upps, „MariaDB“?
Wir wollen doch auf „MySQL“-Daten zugreifen.
Glücklicherweise sind die beiden Datenbanken zu einander kompatibel.
- Jetzt noch im Browser den „localhost“ aufrufen um zu sehen das alles läuft.



- Und weiter geht es.

4. Einlesen der vorhanden Daten in die MariaDB



- Wir starten nach dem Start von XAMPP im Browser wieder „localhost“ und starten dann „phpMyAdmin“

The screenshot displays the phpMyAdmin web interface for a local MySQL/MariaDB server. The interface is organized into several panels:

- Left Sidebar:** Contains navigation links for 'Neu' (New), 'information_schema', 'mysql', 'performance_schema', 'phpmyadmin', 'test', and 'usr_web46_9'. It also includes 'Letzte' (Recent) and 'Favoriten' (Favorites) sections.
- Top Navigation Bar:** Features tabs for 'Datenbanken' (Databases), 'SQL', 'Status', 'Benutzerkonten' (User accounts), 'Exportieren' (Export), 'Importieren' (Import), 'Einstellungen' (Settings), 'Replikation' (Replication), 'Variablen' (Variables), 'Zeichensätze' (Charsets), 'Formate' (Formats), and 'Erweiterungen' (Extensions).
- Main Content Area:**
 - Allgemeine Einstellungen (General Settings):** Shows the 'Zeichensatz/Kollation der Verbindung zum Server' (Charset/Collation of the connection to the server) set to 'utf8mb4_unicode_ci'. A link for 'Weitere Einstellungen' (More settings) is provided.
 - Anzeige-Einstellungen (Display Settings):** Shows the 'Sprache (Language)' set to 'Deutsch - German' and the 'Oberflächendesign' (Skin) set to 'pmahomme'. A button for 'Alle zeigen' (Show all) is also present.
- Right Sidebar:**
 - Datenbank-Server (Database Server):** Lists server details: 'Server: Localhost via UNIX socket', 'Server-Typ: MariaDB', 'Server-Verbindung: SSL wird nicht verwendet', 'Server-Version: 10.4.28-MariaDB - Source distribution', 'Protokoll-Version: 10', 'Benutzer: root@localhost', and 'Server-Zeichensatz: UTF-8 Unicode (utf8mb4)'.
 - Webserver:** Lists web server details: 'Apache/2.4.56 (Unix) OpenSSL/1.1.1t PHP/8.2.4 mod_perl/2.0.12 Perl/v5.34.1', 'Datenbank-Client Version: libmysql - mysqlnd 8.2.4', 'PHP-Erweiterung: mysqli curl mbstring', and 'PHP-Version: 8.2.4'.
 - phpMyAdmin:** Lists version and resource information: 'Versionsinformationen: 5.2.1 (auf dem neuesten Stand)', 'Dokumentation', 'Offizielle Homepage', 'Mitmachen', 'Unterstützung erhalten', 'Liste der Änderungen', and 'Lizenz'.

- Wir gehen auf den Punkt „Datenbanken“ und erstellen eine neue Datenbank mit einem Namen unserer Wahl. Ich nahm den gleichen wie in der Ursprungsdatenbank.


Datenbanken








Neue Datenbank anlegen


utf8mb4_general_ci
▼

Anlegen

☐ Alle auswählen


Löschen

	Datenbank ▲	Kollation	Aktion
<input type="checkbox"/>	information_schema	utf8_general_ci	 Rechte überprüfen
<input type="checkbox"/>	mysql	utf8mb4_general_ci	 Rechte überprüfen
<input type="checkbox"/>	performance_schema	utf8_general_ci	 Rechte überprüfen
<input type="checkbox"/>	phpmyadmin	utf8_bin	 Rechte überprüfen
<input type="checkbox"/>	test	utf8mb4_general_ci	 Rechte überprüfen
<input type="checkbox"/>	usr_web46_9	utf8mb4_general_ci	 Rechte überprüfen

Insgesamt: 6

- Wie wählen die gerade erstellte Datenbank aus und gehen auf „Importieren“.

Importiere in die Datenbank "usr_web46_9"

Zu importierende Datei:

Datei kann komprimiert (gzip, bzip2, zip) oder unkomprimiert sein.
Der Dateiname einer komprimierten Datei muss mit **.[Format].[Komprimierung]** enden. Beispiel: **.sql.zip**

Durchsuchen Sie Ihren Computer: (Maximal: 100MiB)

Durchsuchen...

Keine Datei ausgewählt.

Sie können auch per Drag & Drop eine Datei auf einer beliebigen Seite legen.

Zeichencodierung der Datei:

utf-8

Teilweiser Import:

☒ Erlauben Sie die Unterbrechung eines Imports, falls das Skript erkennt, dass es sich dem PHP-Timeout-Limit nähert.
Dies kann eine gute Möglichkeit sein, große Dateien zu importieren, kann jedoch Transaktionen unterbrechen.

Diese Anzahl Abfragen (für SQL) überspringen, beginnend von der ersten:

0

Andere Optionen

☒ Fremdschlüsselüberprüfung aktivieren

Format

SQL

Formatspezifische Optionen:

SQL-Kompatibilitätsmodus:

NONE

☒ AUTO_INCREMENT nicht für Nullwerte verwenden

Importieren

- Im Importdialog, navigieren wir zu unserem „.sql“-Backup und starten den Importvorgang.
- Kommt es (wie bei mir) an dieser Stelle zu Problemen, weil die zu importierende Datei zu groß ist, befolgen Sie diese Tipps:

Um den Import und Export via phpMyAdmin zu bewerkstelligen, ist eine Änderung der Datei **php.ini** (bei XAMPP: **INSTALLATIONSPFAD\xampp\php\php.ini**) erforderlich.

Das Problem ist hierbei, dass nicht alle Webhoster dem Kunden diese Datei zur Verfügung stellen.

Sofern man darauf Zugriff hat, müssen drei Werte geändert werden.

Die maximale Ausführungszeit wird von 30 Sekunden auf 2000 Sekunden hochgesetzt:

max_execution_time=2000

Die maximale Dateigröße wird von 2 MB auf 200 MB geändert:

upload_max_filesize=200M

Und die maximale POST-Größe wird von 8 MB auf 200 MB festgelegt:

post_max_size=200M

Des Weiteren muss der Wert der Variable **\$cfg['ExecTimeLimit']** in der Datei **config.default.php** von phpMyAdmin (in XAMPP:

INSTALLATIONSPFAD\xampp\phpMyAdmin\libraries\config.default.php) von 300 Sekunden auf 2000 Sekunden abgeändert werden:

\$cfg['ExecTimeLimit'] = 2000;

Nun muss die Datei gespeichert und der Webserver neu gestartet werden.

Bei XAMPP öffnen wir hierfür das **XAMPP Control Panel** (**INSTALLATIONSPFAD\xampp\xampp-control.exe**) und beenden den Apache-Server. Anschließend muss dieser wieder gestartet werden.

Quelle: <https://www.homeconstructor.net/de/mysql-import-export-grosser-datenbanken>

- Nun sollten Sie schon den Inhalt der importierten Datenbank in phpMyAdmin sehen können.

Filter

Beinhalten das Wort:

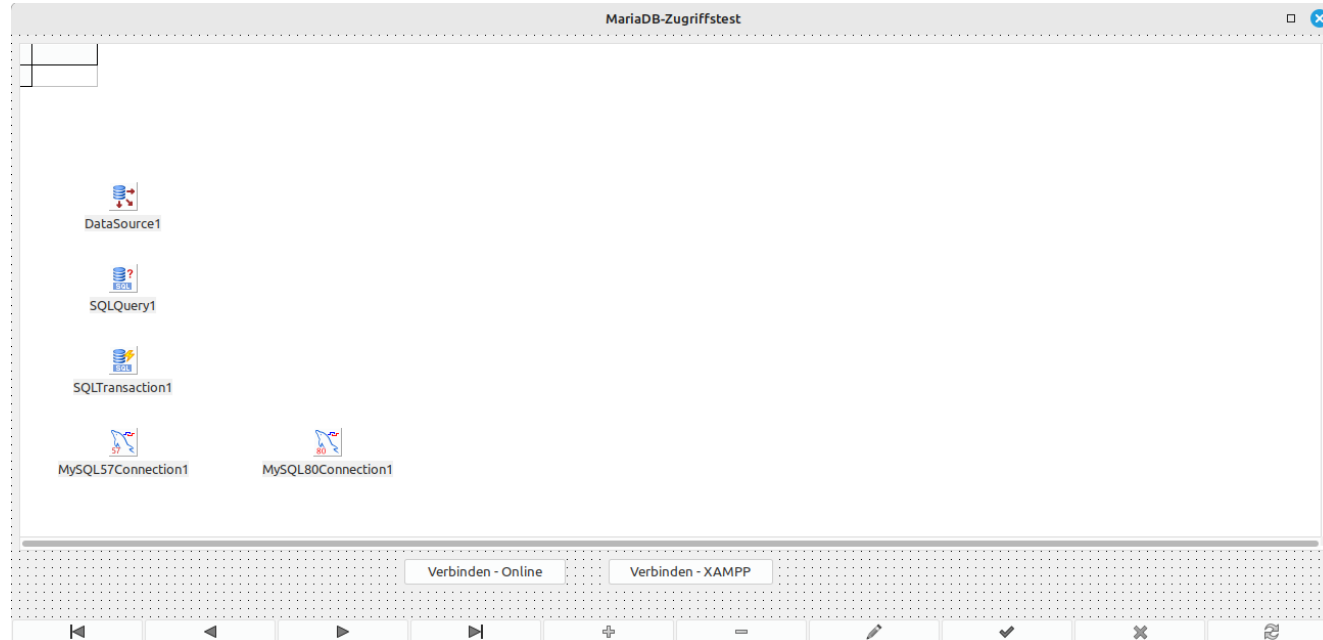
	Tabelle	Aktion	Datensätze	Typ	Kollation	Größe	Überhang
<input type="checkbox"/>	jos311_action_logs	★	2.249	InnoDB	utf8mb4_unicode_ci	1,9 MiB	-
<input type="checkbox"/>	jos311_action_logs_extensions	★	20	InnoDB	utf8mb4_unicode_ci	16,0 KiB	-
<input type="checkbox"/>	jos311_action_logs_users	★	1	InnoDB	utf8mb4_unicode_ci	32,0 KiB	-
<input type="checkbox"/>	jos311_action_log_config	★	25	InnoDB	utf8mb4_unicode_ci	16,0 KiB	-
<input type="checkbox"/>	jos311_akeeba_common	★	5	MyISAM	utf8_general_ci	7,4 KiB	-
<input type="checkbox"/>	jos311_ak_profiles	★	1	InnoDB	utf8mb4_unicode_ci	16,0 KiB	-

- Kommen wir nun zum LAZARUS-Teil.

5. Das Lazarus-Formular

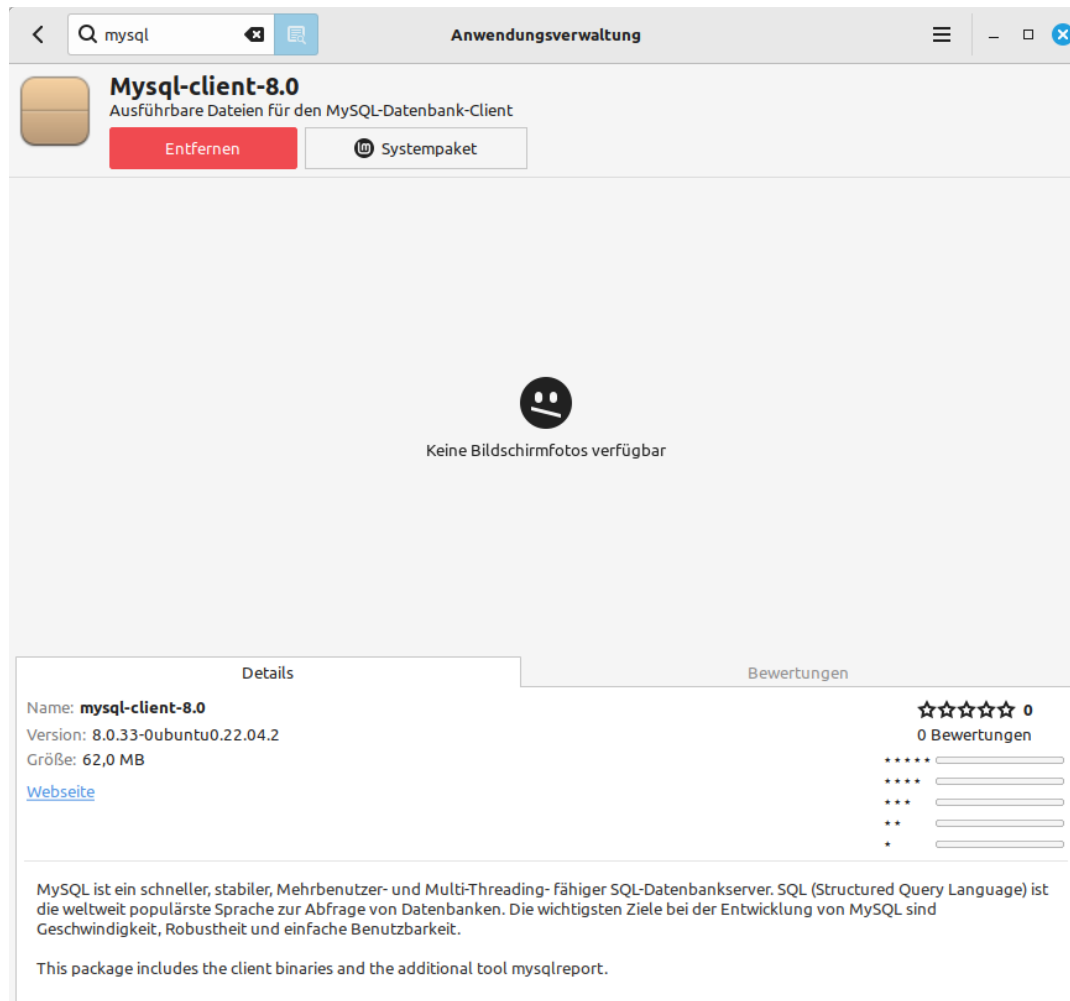
Nun wollen wir das Formular für den Zugriff auf die Daten erstellen.
Wir platzieren folgende Komponenten auf dem Formular:

- TDBGrid
- TdataSource
- TSQLQuery
- TSQLTransaction
- TMySQL80Connection
- TButton



Damit die Verbindung per „TMySQL80Connection“ funktioniert, muss im Betriebssystem noch der entsprechende MySQL-Client installiert werden.

Im Fall meines Linux Mint benötige ich den „Mysql-client-8.0“.



6. Probleme bei der Installation des MySQL-Clients

An dieser Stelle lauert noch ein Problem:

Als ich die erste Version des Programms auf meinem „Manajaro“-PC gestartet habe, fehlte der entsprechende Client.

Der richtige Client ließ sich aber nicht installieren, weil das Multimedia-Programm „Kodi“ eine „MariaDB“-Instanz installiert und dessen Client verhinderte das Installieren des benötigten Clients.

7. Verbinden der Komponenten verbinden

Jetzt müssen die Komponenten nur noch wissen, wer mit wem zusammenarbeiten soll.

Bei „TMySQL80Connection“ muss unter „Transaction“ die Komponente „TSQLTransaction“ ausgewählt werden.

Bei der „TSQLTransaction“ muss unter „Database“ die „TMySQL80Connection“ ausgewählt werden.

Bei der „TSQLQuery“ muss unter „Database“ die „TMySQL80Connection“ ausgewählt werden.

Bei der „TDataSource“ muss unter „DataSet“ die „TDataSource“ ausgewählt werden.

Beim „TDBGrid“ muss unter „DataSource“ die „TDataSource“ ausgewählt werden.

8. Code im „OnClick“-Ereignis

```
procedure TForm1.Button_Verbinden_XAMPPClick(Sender: TObject);
begin
    // Parameter
    // =====
    MySQL80Connection1.CharSet:='utf8mb4';
    MySQL80Connection1.DatabaseName:= 'usr_web46_9';
    MySQL80Connection1.HostName:= '127.0.0.1';
    MySQL80Connection1.Port:= 3306;
    MySQL80Connection1.UserName:= 'root';
    MySQL80Connection1.Password:='';
    MySQL80Connection1.Params.Text:='Port=3306';
    MySQL80Connection1.SkipLibraryVersionCheck:= False;
    //MySQL80Connection1.Transaction.;    // Wie an dieser Stelle definieren???
    // =====
    DBGrid1.DataSource:= DataSource1;
    // Spaltenbreite automatisch aktivieren
    // Memo-Feldinhalt als Text anzeigen
    DBGrid1.Options := DBGrid1.Options + [dgAutoSizeColumns]+ [dgDisplayMemoText];
    // =====
    SQLTransaction1.Database:= MySQL80Connection1;
    SQLTransaction1.Active:= true;
    // =====
    // Ermittelt die Anzahl der Einträge in der Tabelle
    SQLQuery1.Close;;
    SQLQuery1.SQL.Text:='SELECT COUNT(*) FROM jos311_jdownloads_files_backup_3_2_69;';
    SQLQuery1.Open;
    // ShowMessage(SQLQuery1.Fields[0].AsString); // Shows "105"
    SQLQuery1.Close;
    //
    //SQLQuery1.UniDirectional:=false; // Was ist hier richtig???
    //
    // SQLQuery1.PacketRecords:=105; // Keine Änderung. Weiter Absturz.
    SQLQuery1.Database:= MySQL80Connection1;
    SQLQuery1.SQL.Text:='SELECT * FROM jos311_jdownloads_files_backup_3_2_69;';
    SQLQuery1.Active:= true;
end;
```