

# Datenbanken überprüfen, reparieren und optimieren

Bei einer gecrashten Datenbank kann es teilweise helfen, eine Reparatur durchzuführen, statt ein Backup einzuspielen. So entfällt der Datenverlust bzw. die Datenrücksetzung zu einem älteren Stand.

**Bei einem Rootserver lässt sich eine gecrashte Datenbank über den LOG feststellen und sieht in der Regel wie folgt aus:**

```
[ERROR] /usr/local/mysql5/sbin/mysqld: Table './usrdb_exampe/exampleusr' is marked as crashed and last (automatic?) repair failed
```

Bei Hosting-Tarifen haben Sie keinen direkten Zugriff auf die Error-Logs. Hier ist es im Regelfall so, dass die Webanwendung einen MySQL-Fehler anzeigt.

Hierbei kommt es oft vor, dass nur die Meldung erscheint, dass beispielsweise auf die Tabelle „exampleusr“ nicht zugegriffen werden kann oder diese nicht existiert. Wenn diese aber in phpMyAdmin ersichtlich ist, empfiehlt sich hier eine Reparatur.

**Im Folgenden ist das Vorgehen für einen solchen Fall Schritt für Schritt mit phpMyAdmin beschrieben:**

Vorbereitung

Rufen Sie die phpMyAdmin Verwaltung für Ihren Server/Ihre Datenbank auf.

Melden Sie sich anschließend mit den Zugangsdaten der Datenbank bei phpMyAdmin an.

Wählen Sie nun in der linken Liste die zu überprüfende (ggf. defekte) MySQL-Tabelle aus Ihrer Datenbank aus.

1 Datenbank überprüfen:

Wählen Sie nun den Reiter „Operationen“ in der oberen Leiste aus.

**1.1** Wählen Sie nun im unteren Bereich „Hilfsmittel“ die Funktion „**Überprüfe Tabelle**“ aus. Danach wird Ihnen das Resultat in phpMyAdmin angezeigt.

Der Befehl „CHECK TABLE“ wird dadurch ausgeführt. (Details zu dem Befehl finden Sie in der offiziellen MySQL Dokumentation. <http://dev.mysql.com/doc/refman/5.7/en/check-table.html>)

**1.2** Wählen Sie nun im unteren Bereich „Hilfsmittel“ die Funktion „**Analysiere Tabelle**“ aus. Danach wird Ihnen das Resultat in phpMyAdmin angezeigt.

Der Befehl „ANALYZE TABLE“ wird dadurch ausgeführt. (Details zu dem Befehl finden Sie in der offiziellen MySQL Dokumentation. <http://dev.mysql.com/doc/refman/5.7/en/analyze-table.html>)

2 Datenbank reparieren:

Wählen Sie nun den Reiter „Operationen“ in der oberen Leiste aus.

**2.1** Wählen Sie nun im unteren Bereich „Hilfsmittel“ die Funktion „**Repariere Tabelle**“ aus. Danach wird Ihnen das Resultat in phpMyAdmin angezeigt.

Der Befehl „REPAIR TABLE“ wird dadurch ausgeführt. (Details zu dem Befehl finden Sie in der offiziellen MySQL Dokumentation. <http://dev.mysql.com/doc/refman/5.7/en/repair-table.html>)

3 Datenbank optimieren:

Wählen Sie nun den Reiter „Operationen“ in der oberen Leiste aus.

**3.1** Wählen Sie nun im unteren Bereich „Hilfsmittel“ die Funktion „**Optimiere Tabelle**“ aus. Danach wird Ihnen das entsprechende Resultat in phpMyAdmin angezeigt.

Der Befehl „OPTIMIZE TABLE“ wird dadurch ausgeführt. (Details zu dem Befehl finden Sie in der offiziellen MySQL Dokumentation. <http://dev.mysql.com/doc/refman/5.7/en/optimize-table.html>)

4 Hinweise:

Die oben aufgeführten Befehle können im phpMyAdmin auch über die SQL Konsole im oberen Reiter unter „SQL“ ausgeführt werden.

In diesem Fall geben Sie einfach den jeweils erläuterten Befehl mit dem Datenbanknamen an.

Beispiel [1.1]: CHECK TABLE `exampleusr`

Beispiel [1.2]: ANALYZE TABLE `exampleusr`

Beispiel [2.1]: REPAIR TABLE `exampleusr`

Beispiel [3.1]: OPTIMIZE TABLE `exampleusr`

5 Befehlsbeschreibung:

**CHECK TABLE** - Die hier genannte MySQL Operation überprüft die gewählte MySQL-Tabelle auf mögliche Fehler.

**ANALYZE TABLE** - Die hier genannte MySQL Operation analysiert und prüft die Schlüssel der MySQL-Tabelle auf mögliche Probleme.

**REPAIR TABLE** - Die hier genannte MySQL Operation repariert eine möglicherweise beschädigte MySQL-Tabelle.

**OPTIMIZE TABLE** - Die hier genannte MySQL Operation optimiert und reorganisiert die gewählte MySQL-Tabelle.