

# Datenbanken und Informationssysteme

## Workshop TRIGGER

### TRIGGER

<b>Ausgangslage</b>	<p>Dieser Workshop basiert auf der Datenbank <code>sqlteacherdb</code>.</p> <p>Die Geschäftsleitung wünscht eine Änderungsnachverfolgung für sensible Daten. Konkret betrifft dies die Tabellen <code>Mitarbeiter</code> und <code>Artikel</code>. Es soll irgendwo ersichtlich sein, wer einen Datensatz erstellt, bearbeitet oder gelöscht hat.</p>
<b>Aufgabenstellung</b>	<p>Um da Ziel zu erreichen, sollen Trigger eingesetzt werden. Diese sollen dazu dienen, in einer neuen Tabelle <code>Tracking</code> jeweils einen Datensatz zu erstellen, wenn Daten in der Tabelle <code>Mitarbeiter</code> oder <code>Artikel</code> eingefügt, bearbeitet oder gelöscht werden.</p> <p>In der Tabelle <code>Tracking</code> sollen folgende Informationen gespeichert werden:</p> <ul style="list-style-type: none"> <li>- Aktion (I = Insert, U = Update, D = Delete)</li> <li>- Benutzername, der die Änderung vornimmt</li> <li>- Datum und Uhrzeit der Änderung</li> <li>- Tabellename, auf die die Änderung durchgeführt wurde</li> <li>- Wert des Primärschlüssels des Datensatzes, der verändert wurde</li> </ul>
<b>Schritt 1</b>	<p>Erstellen Sie auf der Datenbank <code>sqlteacherdb</code> die neue Tabelle <code>Tracking</code> mit den korrekten Spalten und Datentypen. Datum und Uhrzeit sollen dabei mit einem Default-Wert gesetzt werden.</p>
<b>Schritt 2</b>	<p>Recherchieren Sie im Internet, wie mittels SQL der aktuelle an der Datenbank angemeldete Benutzer ermittelt werden kann.</p>
<b>Schritt 3</b>	<p>Studieren Sie folgenden Link, der die Funktionsweise von Trigger in MySQL beschreibt:</p> <p><a href="http://dev.mysql.com/doc/refman/5.7/en/triggers.html">http://dev.mysql.com/doc/refman/5.7/en/triggers.html</a></p>
<b>Schritt 4</b>	<p>Recherchieren Sie im Internet, wie Sie für Mysql Trigger erstellen können, die bei folgenden Aktionen ausgelöst werden:</p> <ul style="list-style-type: none"> <li>- Datensatz einfügen (insert)</li> <li>- Datensatz aktualisieren (update)</li> <li>- Datensatz löschen (delete)</li> </ul>

<p><b>Schritt 5</b></p>	<p>Erstellen Sie für die Tabellen <code>Mitarbeiter</code> und <code>Artikel</code> je drei Trigger, die bei folgenden Aktionen ausgeführt werden:</p> <ul style="list-style-type: none"> <li>- Datensatz erstellen</li> <li>- Datensatz manipulieren</li> <li>- Datensatz löschen</li> </ul> <p>Die Trigger sollen vorerst mal bei jedem Aufruf einen neuen Datensatz mit dem aktuellen Benutzer, sowie der Aktion und dem Tabellennamen in die Tabelle <code>Tracking</code> einfügen</p> <p><b>WICHTIG:</b> Bei Trigger kann man jeweils angeben, ob Sie vor oder nach der definierten Aktion ausgeführt werden. Überlegen Sie sich für die einzelnen Trigger, ob die Auslösung des Triggers vor oder nach der Aktion geschehen soll damit die eingangs beschriebene Anforderung erfüllt werden kann.</p>
<p><b>Schritt 6</b></p>	<p>Erstellen Sie nun für die Tabellen <code>Mitarbeiter</code> und <code>Artikel</code> SQL-Anweisungen, um einen Datensatz einzufügen, zu aktualisieren sowie wieder zu löschen. Nachdem Sie diese Anweisungen ausgeführt haben, prüfen Sie den Inhalt in der Tabelle <code>Tracking</code>.</p>
<p><b>Schritt 7</b></p>	<p>Ermitteln Sie eine Möglichkeit, wie Sie herausfinden können, welche Trigger auf der Datenbank <code>sqlteacherdb</code> nun vorhanden sind.</p>
<p><b>Schritt 8</b></p>	<p>Wir benötigen nun eine Möglichkeit, innerhalb des Triggers den Primärschlüsselwert (<code>mitarbeiternr</code> bei Tabelle <code>Mitarbeiter</code> bzw. <code>artikelnr</code> bei Tabelle <code>Artikel</code>) des betreffenden Datensatzes, der eingefügt, aktualisiert oder gelöscht wird zu ermitteln. Die entsprechenden Infos finden sich unter <a href="http://dev.mysql.com/doc/refman/5.7/en/trigger-syntax.html">http://dev.mysql.com/doc/refman/5.7/en/trigger-syntax.html</a> in folgendem Absatz:</p> <p><i>Within the trigger body, the OLD and NEW keywords enable you to access columns in the rows affected by a trigger. OLD and NEW are MySQL extensions to triggers; they are not case sensitive.</i></p> <p><i>In an INSERT trigger, only NEW.col_name can be used; there is no old row. In a DELETE trigger, only OLD.col_name can be used; there is no new row. In an UPDATE trigger, you can use OLD.col_name to refer to the columns of a row before it is updated and NEW.col_name to refer to the columns of the row after it is updated.</i></p> <p>Passen Sie nun die Trigger an, damit zusätzlich jeweils der Primärschlüsselwert in den Datensatz, der in der Tabelle <code>Tracking</code> erstellt wird, hinzugefügt wird.</p> <p>Hinweis: Ein Trigger kann auf MySQL nur angepasst werden, in dem er gelöscht und neu erstellt wird.</p>
<p><b>Schritt 9</b></p>	<p>Nachdem Sie die Trigger nun angepasst haben, wiederholen Sie Schritt 6.</p>

## Links:

<http://dev.mysql.com/doc/refman/5.7/en/trigger-syntax.html>  
<http://dev.mysql.com/doc/refman/5.7/en/create-trigger.html>  
<http://dev.mysql.com/doc/refman/5.7/en/drop-trigger.html>  
<http://dev.mysql.com/doc/refman/5.7/en/information-functions.html>  
<http://dev.mysql.com/doc/refman/5.7/en/triggers-table.html>