

BMSD24a - M106 Datenbank-Projekt

1. Kurzbeschreibung

In einer Gruppenarbeit (max. zwei Lernende) entwerfen und erstellen Sie eine Datenbank zu einem frei gewählten Thema.

Die Datenbank wird unter MS-SQL (T-SQL) erstellt, so dass alle Skripts mit dem «SQL-Server Management Studio» funktionieren.

Das Projekt umfasst:

- eine Dokumentation mit ER-Diagramm, Datenbankbeschreibung
- SQL-Code mit welchem die Datenbank inkl. «Stored Procedure» und «Function» erstellt werden kann
- SQL-Code mit dem die Daten in die Datenbank eingetragen werden
- SQL-Code für Abfragen/Manipulation bzw. Tests inkl. einer kurzen Testbeschreibung zur Kontrolle der beiden Stored Procedure und Function.
- Arbeitsrapport via Git führen (Datum, Beschreibung, Person).

2. Datenbank

Das Datenbank-Thema kann frei gewählt werden und muss vor dem Projektstart durch die Lehrperson genehmigt werden. Das Thema muss eindeutig sein, d.h. es darf innerhalb der Klasse bzw. des Jahrgangs nur einmal vorkommen.

Die Datenbank muss die folgenden Bedingungen erfüllen:

- Das ER-Modell umfasst mindestens vier Tabellen.
- Eine der Beziehungen im ER-Modell muss eine n-m-Beziehung sein. Daraus ergibt es sich, dass das relationale Modell mindestens fünf Tabellen umfasst.
- Es müssen mindestens einmal die folgenden Datentypen verwendet werden:
 - String, z.B. varchar, usw.
 - Ganze Zahl, z.B. integer, usw.
 - Fließ- und/oder Festkommazahl: z.B. float, numeric usw.
 - Wahrheitswert, z.B. bit usw.
 - Datum- und/oder Zeitwert, z.B. date, time usw.
- Alle Primärschlüssel zählen von 1 bis n automatisch hoch (IDENTITY)
- Die referentielle Integrität wird durch die Fremdschlüsselbeziehungen über das System definiert (FOREIGN KEY ... REFERENCES...)
- Es muss mindestens eine «Stored Procedure» verwendet werden
- Es muss mindestens eine «Function» verwendet werden
- Es müssen pro «Stored Procedure» bzw. «Function» mindestens drei verschiedene Abfragen/Manipulationen bzw. Tests programmiert werden
- Es sollen pro Tabelle mindestens 10 Testdatensätze verwendet werden.

3. SQL-Code

Es wird jeweils ein SQL-Skript mit den folgenden Funktionalitäten erwartet:

- zum Erstellen der Datenbank inkl. der «Stored Procedure» und der «Function»
- zum Abfüllen der Datenbank mit Testdaten
- zu jeder Abfrage / Manipulation bzw. Test der «Stored Procedure» bzw. «Function», also insgesamt mindestens zwei einzelne Skripts

4. Erwartetes Resultat

Eine Dokumentation mit:

- Kurzbeschreibung des Themas
- ER-Diagramm der Datenbank
- Relationales Modell der Datenbank inkl. Datentyp und Beschreibung der einzelnen Felder
- Beschreibung der Abfragen/Manipulationen bzw. Tests
- Beschreibung der «Stored Procedure» bzw. «Function» und deren Anwendung

Die «Stored Procedure» bzw. die «Function» sollen über ein einfaches Fehler-Handling verfügen, d.h.:

- falsche oder leere Argumente werden abgefangen
- leere Resultate werden z.B. durch eine Fehlermeldung angezeigt, sofern das möglich ist, oder geben einen definierten Wert, wie z.B. «null» zurück
- der Rückgabewert der «Stored Procedure» gibt die Anzahl der betroffenen Datensätze zurück
- zu den Fehler werden entsprechende Fehlermeldungen generiert, wie im Management Studio üblich
- jeder Fehler führt zu einem definierten Abbruch

Jegliche Dokumente, Diagramme und Skripte werden mittels Git eingchecked:

- Comitten sie nicht als Zweck eines Backups sondern um den Arbeitsfortschritt reproduzieren zu können. Besser viele kleine Commits als einen grossen.
- Es existiert ein Ordner mit der Bezeichnung Abgabe, der am Schluss bewertet wird. Alle anderen Dateien sollen auch im Git Repository geführt werden, diese werden jedoch nicht zur Bewertung hinzugefügt

5. Abgabe

Umfang: Der Ordner «Abgabe» innerhalb des Git Repositories enthält folgende Bestandteile:

- Dokumentation als Markdown-Datei: `Dokumentation.md`
- SQL-Skript für das Erstellen der Datenbank: `Create.sql`
- SQL-Skript für das Einfügen der Daten: `Insert.sql`
- SQL-Skripte mit den Abfragen/Manipulationen bzw. Tests zu der «Stored Procedure» bzw. «Function»: `Programmability.sql`

Art: Inhalt vom Git Repository zum Zeitpunkt des Termins. Für die Plattformen GitLab.com bin ich unter der E-Mail Adresse «manuel.kaspar@sluz.ch» erreichbar. Bitte fügen Sie mich bei Ihrem Projekt als «Betreuer» hinzu.

Termin: **bis spätestens Freitag, 16. Januar 2026 um 23:59 Uhr.**

Es gilt der letzte Commit vor obigem Zeitpunkt, Commits danach werden nicht mehr berücksichtigt.