

Teil 1: Bank ERM

Installiere eine beliebige Freeware zum Erstellen eines ERM sowie eine dazu passende Datenbank. Erkläre in der Dokumentation, wieso du diese Wahl getroffen hast.

Erstelle ein ERM für eine Bank. Diese ist in einzelne Filialen unterteilt. Jede Filiale hat ein Vermögen, einen eindeutigen Namen und kann Konten und Kredite haben. Jedes Konto und jeder Kredit haben eine eindeutige Nummer und kann jeweils einem oder mehreren Kunden zugewiesen sein.

Der Name der Datenbank soll „bank“ sein.

Die Tabellen könnten beispielsweise wie folgt aussehen: Filiale, Kundenkonto, Kredit, Kontoinhaber, Kreditnehmer

Es sollten mindestens folgende Informationen gespeichert werden:

- Filiale: Name der Filiale, Adresse, Telefonnummer, E-Mail, Vermögen

- Konto: IBAN, Kontostand, Typ des Kontos (bspw. Girokonto, Sparkonto, Kreditkonto, ...)

- Kredit: Höhe des Kredits, Typ des Kredits (bspw. Autokredit, Hypothekenkredit, ...)

- Kunde: Anrede, eindeutiger Name (bspw. Herr Ing. Franz Mustermann), Adresse, Geburtsdatum

- Ein Kunde kann mehrere Konten besitzen (bspw. Giro- und Sparkonto)

- Mehrere Kunden können ein Konto besitzen (bspw. Ehekonto)

- Mehrere Kunden können einen Kredit aufnehmen (bspw. gemeinsamer Hauskredit)

- Jeder Kunde hat eine Stammfiliale

Achte auf die sinnvolle Verwendung von Datentypen.

Jede Tabelle muss mindestens 5 Testeinträge deiner Wahl haben.

Erstelle nun Anhand des ERM und der Testeinträge eine Datenbank. Der Source Code (SQL) hierfür muss auch in die Dokumentation. Bitte achte darauf, dass dieser verständlich kommentiert ist.

Erstelle 3 sinnvolle Abfragen für die Datenbank und dokumentiere diese inklusive der Ergebnisse. Achte auf die Formatierung der Ausgaben. Teste nun deine Datenbank ausführlich und dokumentiere den Prozess.

Drucke die Dokumentation mit deinem Namen und der Seitenzahl auf jedem Blatt aus. Öffne das ERM auf deinem Computer und präsentiere es dem Prüfer.

Teil 2: Webinterface / Server für Datenbankexplorer

Erstelle eine Seite auf der alle verfügbaren Datenbanken des verbundenen Datenbankservers angezeigt werden.

Diese könnte wie folgt aussehen:

mysql	Details
information_schema	Details
schema_1	Details
schema_2	Details

Durch einen Klick auf „Details“ öffnet sich eine zweite Seite. Auf dieser soll der Name und die Tabellen der ausgewählten Datenbank angezeigt werden.

Diese könnte wie folgt aussehen:

Datenbank: schema_1

table_1	Details
table_2	Details
table_3	Details
table_4	Details

Durch den Klick auf „Details“ öffnet sich eine dritte Seite. Auf dieser soll der Name der Datenbank, der Name der Tabelle und die Informationen zu den Spalten dieser angezeigt werden.

Diese könnte wie folgt aussehen:

Datenbank: schema_1

Tabelle: table_1

name	type	nullable	default	key	extra
id	INT(11)	NO		PRI	auto_increment
name	VARCHAR(50)	YES			

Teste nun deine Software ausführlich und dokumentiere deine Vorgehensweise. Dokumentiere zudem den Source Code deiner Software.

Drucke die Dokumentation mit deinem Namen und der Seitenzahl auf jedem Blatt aus. Öffne die Oberfläche auf deinem Computer und präsentiere diese dem Prüfer.

Teil 3: Fachgespräch

Nach einer kurzen Vorstellungsrunde werden alle Teilnehmer einzeln aufgerufen. Das Prüfungsteam besteht aus einem Vorsitzenden und zwei Prüfern. Die meisten fachkundigen Fragen werden von letzteren gestellt.

Am Anfang des Gespräches wird oft gefragt, wo man arbeitet und was man dort so macht. Je nach Antwort können hier Folgefragen gestellt werden. Datenschutz ist da anscheinend immer ein beliebtes Thema.

Mein Gespräch war leider etwas kurz. Hier einmal trotzdem die Fragen, die ich im Gespräch hatte:

- Worum handelt es sich bei einer abstrakten Klasse (in Java)?
- Was für Arten von Suchalgorithmen kennst du?
- Was für Arten von Schleifen (in Programmiersprachen) kennst du?
- Was sind sensible Daten (in Bezug auf DSGVO)?
- Was ist ein Code Review? Wieso macht man diese? Was genau wird dabei überprüft?
- Was sind Coding Guidelines?

Und hier noch ein paar Fragen/Themen die ich von anderen Teilnehmern gehört habe:

- Alle möglichen Fragen rund um Datenschutz
- Was ist ein Dateisystem? Was sind die Unterschiede zwischen FAT und NTFS?
- Wie gehst du bei einem Projekt vor, dass du programmieren musst (nicht aus Projektleiter, sondern aus Entwicklersicht)?
- Was ist das OSI Schichtenmodell? Welche Schichten gibt es? Was spielt sich auf welchen ab?
- Was ist der Unterschied zwischen TCP und UDP?
- Was ist „Big Data“?
- Was für aktuelle Programmiersprachen kennst du und was kannst du zu diesen erzählen?
- Was kannst du zu Cross-Plattform Entwicklung erzählen?
- Wofür stehen die Begriffe KISS und DRY?
- Was ist ein BIS?
- Worum handelt es sich bei einem Data Warehouse? Was ist OLAP?

Allgemeines

Die Beispiele im Praxisteil können variieren, jedoch der grundlegende Aufbau dürfte bei allen recht ähnlich sein. Für jeden Teil hat man jeweils 3,5h Zeit zum Erarbeiten. Die restlichen 0,5h sind jeweils für das Präsentieren an den Prüfer gedacht. Das Fachgespräch dauert ungefähr 15 bis 20 Minuten.

Das „Präsentieren“ am Ende in diesem Fall ist einfach eine kurze Vorstellung des ERMs oder der Nutzeroberfläche. Die Lösungen müssen alle Voraussetzungen erfüllen, jedoch müssen sie nicht perfekt sein. Oft fragen die Prüfer dann nach, wieso man denn etwas auf diese Art gelöst hat und nicht anders (bspw. Warum hast du für Feld XY einen VARCHAR und keinen INT verwendet?).

Bei den beiden Praxisteilen sind auch immer drei Prüfer da. Diese helfen bei Verständnisfragen, jedoch (obviously) nicht bei der eigentlichen Umsetzung. Dafür darf jedoch jederzeit das Internet zur Recherche verwendet werden. Sich wo anmelden und mit anderen kommunizieren ist verboten. Es dürfen alle Programme verwendet werden, die keine Lizenz benötigen. Dazu zählen auch Testversionen von kostenpflichtigen Programmen.

Die Downloads sind nicht gerade schnell (~2 MB/s), jedoch werden alle Dateien die einmal herunter geladen wurden im lokalen Netzwerk zwischengespeichert. Umso schneller sind dafür die Computer. Meiner hatte 8 Kerne/16 Threads, 16 GB Arbeitsspeicher und eine SSD. Man hat auf den Computern Adminrechte.

Beim zweiten Praxisteil ist keine Technologie vorgeschrieben. Es dürfen also sämtliche Programmiersprachen, Libraries und Frameworks verwendet werden. Folglich ist es auch in Ordnung sich ein grundlegendes Projekt (Webserver, Datenbankverbindung, Error Handling) vorzubereiten. Ich würde das empfehlen, da die Zeit vor allem mit Dokumentation relativ knapp werden kann. Eventuell Fragen die Prüfer am Anfang nach, was für eine Technologie man verwenden möchte.

Info ohne Gewähr.

Installers:

- Chrome
- Git
- MySQL Workbench
- Java 11
- NodeJS 16
- MySQL 8
- IntelliJ, WebStorm

Backend:

- Config
- Database
- Database Dao
- Web Server
- Session Management
- Request Validation
- CORS
- Error Handling
- Tests

Frontend:

- Stylesheet
- Validation
- Form to Json / Json to Form
- Date / Time Picker
- Loading spinner
- Select with remote data
- modals
- data tables
- buttons in data tables

Project:

- Run tests (from console) | gradlew test
- Build Jar | gradlew jar farJar

Preparations:

- Docs list
- Backend, Frontend, Project
- Upload to Github
- Try it