

Dies ist eine Klausur mit geschlossenen Büchern: Die einzigen erlaubten Hilfsmittel sind leeres Papier, Stifte und Ihr Kopf, aber Sie können eine handgeschriebene A4-Seite mit Informationen verwenden, die Sie für die Lösung der Aufgaben für nützlich halten. Erläutern Sie Ihre Überlegungen. Schreiben Sie klar und deutlich, im Sinne von Logik, Sprache und Lesbarkeit. Die Klarheit Ihrer Erklärungen wirkt sich auf Ihre Note aus. Viel Glück!

Schreiben Sie Ihren Namen und Ihre Matrikelnummer auf alle Lösungsblätter und hier. Name:
Am Ende der Prüfung geben Sie bitte alle Blätter, die Sie erhalten haben, einschließlich dieses, ab. Matrikelnummer:

Problem 1 [30%] Entwerfen Sie das Entity-Relationship-Schema einer Anwendung zur Verwaltung von Gerichten, die von Caterern angeboten und an Veranstaltungsorte geliefert werden, für die folgende Informationen von Interesse sind. Von jedem *Caterer* interessieren uns die Steuernummer (Identifikator) und die Adresse. Von jedem *Gericht* interessieren uns der Name und der Typ (normal, vegetarisch, vegan, etc.), die zusammen das Gericht identifizieren, die Kalorien sowie die Caterer, die das Gericht anbieten, jeweils mit dem Preis, zu dem sie es anbieten. Es ist zu beachten, dass ein Gericht von mindestens einem Caterer angeboten werden muss und von mehreren Caterern angeboten werden kann. Zusätzlich interessiert uns von jedem Gericht die *Menge der Veranstaltungsorte*, an denen es angeboten wird. Diese Menge darf nicht leer sein, kann sich jedoch im Laufe der Zeit ändern, höchstens einmal pro Monat. Wir sind daran interessiert, das Datum (d.h. den Tag zusätzlich zum Monat und zum Jahr) zu erfassen, seitdem die Menge der Veranstaltungsorte das Gericht anbieten. *Gourmetgerichte* sind eine Art von Gericht mit besonderen Zutaten, für die einer der anbietenden Caterer als Hauptanbieter fungiert und einen Mindestpreis für das Gericht festlegt. Von jedem Gourmetgericht interessieren uns die Anzahl der besonderen Zutaten, der Hauptanbieter und der Mindestpreis, den dieser für das Gericht festgelegt hat. Beachten Sie, dass kein Caterer ein Gourmetgericht zu einem niedrigeren Preis als dem vom Hauptanbieter festgelegten Mindestpreis anbieten darf. Schließlich interessieren uns von jedem *Veranstaltungsort* der Name (Identifikator) und die Stadt, in der er sich befindet.

Problem 2 [40%] Führen Sie das logische Design der Datenbank durch und erstellen Sie das vollständige relationale Schema mit Constraints, wobei folgende Angaben zu berücksichtigen sind: (i) Wir greifen separat auf Gourmetgerichte und gewöhnliche Gerichte zu. (ii) Jedes Mal, wenn wir auf ein Gourmetgericht zugreifen, möchten wir immer wissen, wer der Hauptanbieter ist, sowie die Anzahl der besonderen Zutaten und die Kalorien.

In Ihrem Design sollten Sie der im Kurs vorgestellten Methodik folgen und Folgendes erstellen:

1. [7%] das restrukturierte Entity-Relationship-Schema (falls nötig mit externen Constraints),
2. [25%] die direkte Übersetzung in das relationale Modell (falls nötig mit externen Constraints), und
3. [8%] das restrukturierte relationale Schema (erneut mit Constraints).

Sie sollten explizit begründen, wie die obigen Hinweise Ihr Design beeinflussen.

Problem 3 [15%] Betrachten Sie eine Datenbank D , die die folgenden zwei Relationen enthält:

- (i) $\text{Activity}(\underline{\text{code}}, \text{type})$, die den Code und den Typ (einen String) der von einem Resort angebotenen Aktivitäten speichert;
- (ii) $\text{Schedule}(\text{guest}, \text{activitycode}, \text{time})$, mit einer Fremdschlüsselbedingung von activitycode auf code in Activity , die die Aktivitäten speichert, die Gäste im Resort geplant haben durchzuführen.

Beachten Sie, dass die Menge aller Aktivitätstypen durch alle Typen bestimmt wird, die als Typ einer von dem Resort angebotenen Aktivität erscheinen.

1. Schreiben Sie in **relationaler Algebra** eine Anfrage über D , die alle Paare (g, t) berechnet, so dass keine Aktivität vom Typ t im Zeitplan des Gastes g erscheint.
2. Schreiben Sie in **SQL** eine Anfrage über D , die alle Gäste berechnet, die in ihrem Zeitplan mindestens eine Aktivität jedes Typs haben.

Problem 4 [15%] Geben Sie die Definition einer Integritätsbedingung sowie einer referentiellen Integritätsbedingung an. Betrachten Sie dann das folgende relationale Schema S (in dem Primärschlüssel unterstrichen sind und Attribute mit einem Sternchen NULL enthalten können):

$R(\underline{A}, B^*, C)$	$Q(D, E, \underline{F})$
Tuple constraint: $B \text{ IS NOT NULL implies } B = A$	Foreign key: $Q[D, E, F] \subseteq R[A, B, C]$

Für jedes der folgenden drei Elemente geben Sie an, ob es mindestens eine Datenbankinstanz gibt, die im Hinblick auf das Schema S korrekt ist und die entsprechende Eigenschaft besitzt:

1. die Menge $\{B\}$ ist kein Superschlüssel für R ;
2. die Menge $\{C\}$ ist kein Superschlüssel für R ;
3. die Menge $\{D, E\}$ ist kein Superschlüssel für Q .

Für jedes der oben aufgeführten drei Elemente gilt: Falls die Antwort positiv ist, geben Sie die entsprechende Datenbankinstanz an; falls die Antwort negativ ist, erläutern Sie im Detail, warum eine Datenbankinstanz, die im Hinblick auf das Schema S korrekt ist, nicht existieren kann.