Aufgabe 1 a) Geben Sie *minimale* Domänen für die Attribute an.
(Berücksichtigen Sie nur auftretende Werte)

b) Bestimmen Sie das volle Schema der Relation.
(Wählen Sie dafür geeignetere Domänen als die in a) bestimmten)

c) Geben Sie die Tabelle als Relation (Menge von Tupeln) an.

Lösung

a)

$$\begin{aligned}
&TelefonNr : D = \{NULL, 0338541212, 0338541214\}\\
&Talort : D = \{NULL, Grindelwald, Wengen\}\\
&Skigebiet : D = \{NULL, First, Kl.Scheidegg\}\\
&Lift : D = \{NULL, Oberjoch, Oberjäer, Fallboden\}\\
&Kapazität : D = \{NULL, 2000, 2500, 300\}
\end{aligned}$$

b)

$$\begin{aligned}
S_{Liftbetreiber} = (&\text{TelefonNr: string } \cup \ \{\text{NULL}\},\\
&\text{Talort: string } \cup \ \{\text{NULL}\},\\
&\text{Skigebiet: string } \cup \ \{\text{NULL}\},\\
&\text{Lift: string } \cup \ \{\text{NULL}\},\\
&\text{Kapazität: int } \cup \ \{\text{NULL}\})
\end{aligned}$$

c)

$$\begin{aligned}
S_{Liftbetreiber} = \{&(\text{0338541212, Grindelwald, First, Oberjoch, 2500}),\\
&(\text{0338541212, Grindelwald, First, Oberläger, 2000}),\\
&(\text{0338541212, Grindelwald, Kl. Scheidegg, Fallboden, 3000}),\\
&(\text{0338541214, Wengen, Kl. Scheidegg, Fallboden, 3000})\}
\end{aligned}$$

Aufgabe 2
Definition: Ein Schlüsselkandidat heisst **minimal** falls keines der Attribute darin weggelassen werden kann ohne dass er die Primärschlüssel-Eigenschaft verliert.
Das heisst formal:
Falls $K = (U_1, \ldots, U_n)$ eine Sequenz von Attributen ist, so dass für alle Instanzen R von Schema $\mathcal{S}$ gilt dass

$$\forall s, t \in R : s[K] = t[K] \implies s \simeq t,$$

dann ist K minimal falls eine Instanz R_1, zwei Tupel $s_1, t_1 \in R_1$ und ein $1 \leq i \leq n$ existieren, so dass gilt:

$$s_1[(U_1, \ldots, U_{i-1}, U_{i+1}, \ldots, U_n)] = t_1[(U_1, \ldots, U_{i-1}, U_{i+1}, \ldots, U_n)] \wedge \neg(s_1 \simeq t_1)$$

a) Bestimmen Sie alle für das Schema der gegebenen Tabelle noch möglichen minimalen Schlüsselkandidaten (möglichen Primärschlüssel) und zwei nichtminimale Schlüsselkandidaten. Sie müessen die Minimalität Ihrer Schlüssel *nicht* beweisen.

b) Welchen Primärschlüssel (primary key) würden Sie wählen? Warum? Das Hinzufügen von Spalten ist erlaubt, falls es sinnvoll ist.

Lösung
Minimale Schlüssel sind: Schlüssel, die minimal sind ⇒ 2 Eigenschaften!
Es sind Schlüssel, sie erschliesen also die restlichen Attribute.
Sie sind minimal, es können also keine Attribute weggelassen werden, ohne dass die Schlüsseleigenschaft verloren geht.

a) Minimale Schlüssel:

- K_1=(TelefonNr, Lift)
- K_2=(TelefonNr, Kapazität)
- K_3=(Talort, Lift)
- K_4=(Talort, Kapazität)

Nicht minimale **Schlüssel**. z.B.:

- K_5=(TelefonNr, Skigebiet, Lift)
- K_6=(Talort, Lift, Kapazität)

b) Einen neuen Schlüssel LiftbetreiberID oder ähnlich.

Aufgabe 3
Wie kann man die folgenden Integritätsbedingungen formulieren? Ausser bei der Aufgabe a) darf davon ausgegangen werden, dass keine NULL-Werte vorkommen.

a) Jeder Liftbetreiber muss eine Telefonnummer besitzen.

b) Jeder Liftbetreiber muss eine Kapazität von mindestens 1000 und höchstens 5000 aufweisen.

c) Liftbetreiber mit gleichem Talort müssen über die selbe Telefonnummer erreicht werden.

d) Verschiedene Liftbetreiber müssen entweder unterschiedliche Lifte haben, oder in unterschiedlichen Talorten sein.

Zusatzaufgabe(freiwillig): Wie müssen die entsprechenden Integritätsbedingungen in den Aufgaben c) und d) lauten, wenn NULL-Werte möglich sind?

Lösung

a) $\forall s \in Liftbetreiber$: s[TelefonNr] is not NULL

Eine **falsche** Lösung wäre zum Beispiel: $\forall s \in Liftbetreiber$: s[TelefonNr] $\neq$ NULL
Weil: Das Tupel (NULL, Grindelwald, First, Oberläger, 2000) erfüllt obige Bedingungen weil: NULL $\neq$ NULL

b) $\forall s \in Liftbetreiber : 1000 \leq s$[Kapazität] ≤ 5000

c) $\forall s,t \in Liftbetreiber$: (t[Talort]=s[Talort]⇒ t[TelefonNr] = s[TelefonNr])

Falls NULL-Werte möglich sind: $\forall s,t \in Liftbetreiber$: (s[Talort] NULL ∨ t[Talort] NULL) ∨ (t[Talort]=s[Talort]⇒ t[TelefonNr] ≃ s[TelefonNr])

d) $\forall s,t \in Liftbetreiber$: (s[Lift, Talort] = t[Lift, Talort] ⇒ s = t)

Falls NULL-Werte möglich sind: $\forall s,t \in Liftbetreiber$: (s[Lift] NULL ∨ t[Lift] NULL ∨ s[Talort] NULL ∨ t[Talort] NULL) ∨ (s[Lift, Talort] = t[Lift, Talort] ⇒ s ≃ t)

Warum ist $\forall s, t \in Liftbetreiber$: (s[Lift] $\neq$ t[Lift] $\vee$ s[Talort] $\neq$ t[Talort]) keine Lösung?
Nur ein Tupel in dem Lift oder Talort NULL sind, erfüllt diese Bedingung.
Begründung: Mit $\forall$ verlangen wir, dass wir jedes beliebige Tupel für s und t einsetzen können, insbesondere auch für beide das gleiche Tupel.