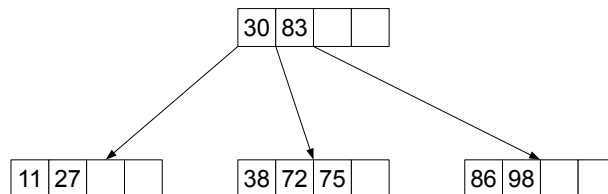
	Lehrveranstaltung	<b>Grundlagen von Datenbanken</b>		WS 2019/20
	Aufgabenzettel	<b>6 (Lösungsvorschläge)</b>		
	Gesamtpunktzahl	<b>30</b>		
	Ausgabe	<b>Mi. 08.01.2020</b>	Abgabe	<b>Fr. 24.01.2020</b>

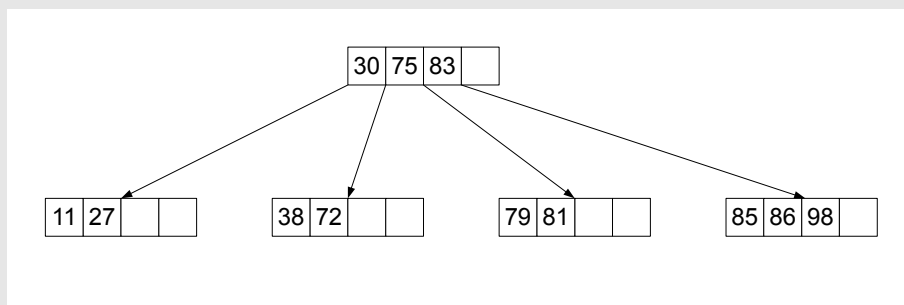
## 1 Präsenzaufgabe: Einfügen in B-Bäumen

Gegeben ist der unten vereinfacht dargestellte **B-Baum** der Klasse  $\tau(2, h)$ . Fügen Sie die (Datensätze mit den) Schlüsselwerte(n) **79, 85, 81, 1, 29, 92, 9 und 94** in dieser Reihenfolge ein. Geben Sie in jedem Einfügeschritt die verwendete Maßnahme (einfaches Einfügen in einen Knoten, Splitten) an und zeichnen Sie den Baum nach jedem Split-Vorgang neu. Als Splitfaktor wird dabei  $m = 1$  gewählt.



### Lösungsvorschlag:

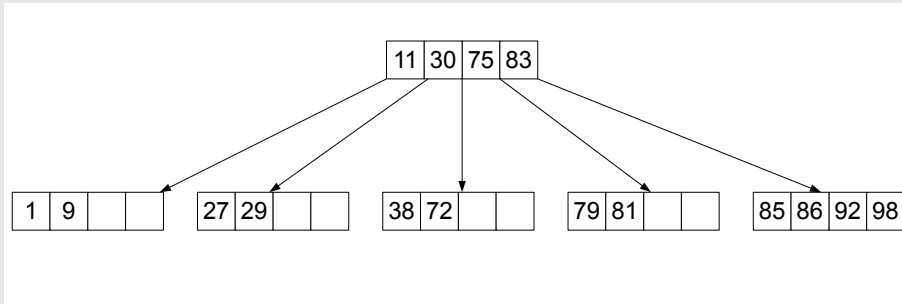
Schlüsselwert 79 (einfaches Einfügen)  
 Schlüsselwert 85 (einfaches Einfügen)  
 Schlüsselwert 81 (Splitten)



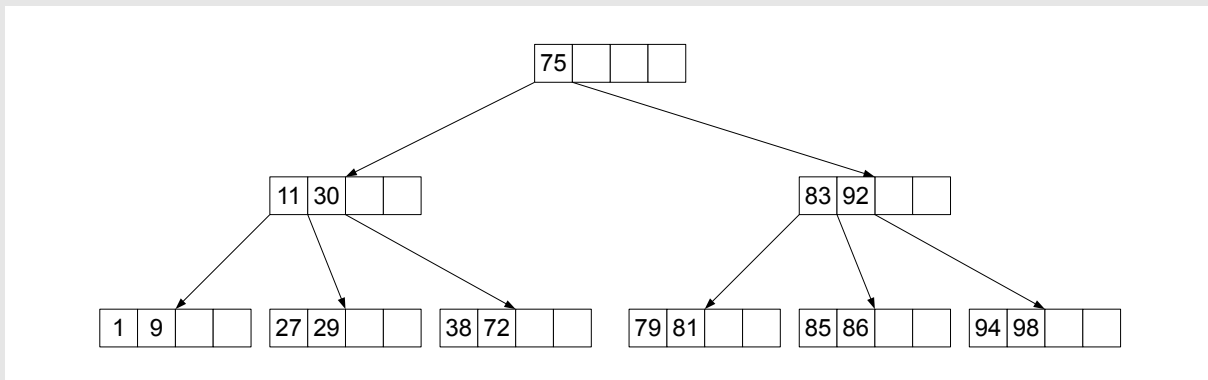
Schlüsselwert 1 (einfaches Einfügen)  
 Schlüsselwert 29 (einfaches Einfügen)  
 Schlüsselwert 92 (einfaches Einfügen)  
 Schlüsselwert 9 (Splitten)



Lehrveranstaltung	<b>Grundlagen von Datenbanken</b>		WS 2019/20
Aufgabenzettel	<b>6 (Lösungsvorschläge)</b>		
Gesamtpunktzahl	<b>30</b>		
Ausgabe	<b>Mi. 08.01.2020</b>	Abgabe	<b>Fr. 24.01.2020</b>



Schlüsselwert 94 (Splitten)

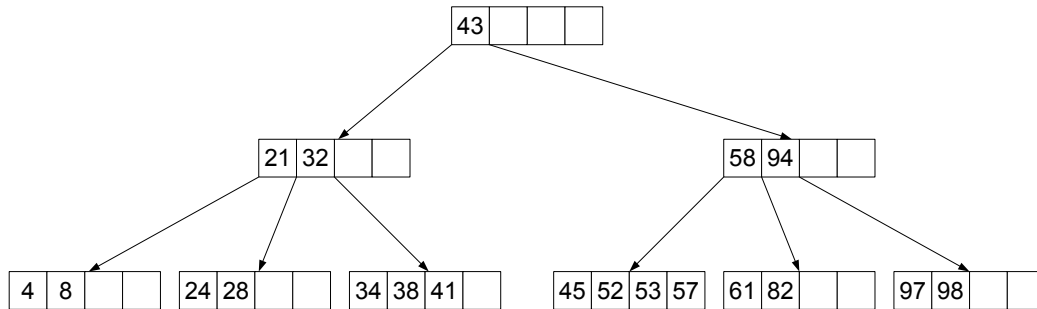


## 2 Präsenzaufgabe: Löschen in B-Bäumen

Gegeben ist der unten vereinfacht dargestellte **B-Baum** der Klasse  $\tau(2, h)$ . Löschen Sie die (Datensätze mit den) Schlüsselwerte(n) **52, 34, 97, 41, 94 und 28** in dieser Reihenfolge. Geben Sie in jedem Löschriff die verwendete Maßnahme (einfaches Löschen, Mischen, Ausgleichen) an und zeichnen Sie den Baum nach jeder Veränderung der Knotenstruktur (Mischen, Ausgleichen) neu. Für Ausgleichoperationen sollen nur unmittelbare Nachbarknoten herangezogen werden.

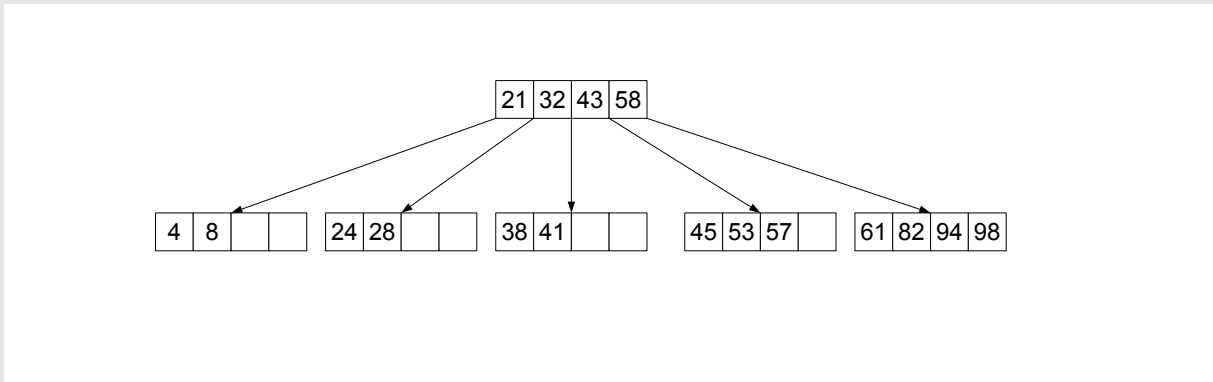


Lehrveranstaltung	<b>Grundlagen von Datenbanken</b>		WS 2019/20
Aufgabenzettel	<b>6 (Lösungsvorschläge)</b>		
Gesamtpunktzahl	<b>30</b>		
Ausgabe	<b>Mi. 08.01.2020</b>	Abgabe	<b>Fr. 24.01.2020</b>




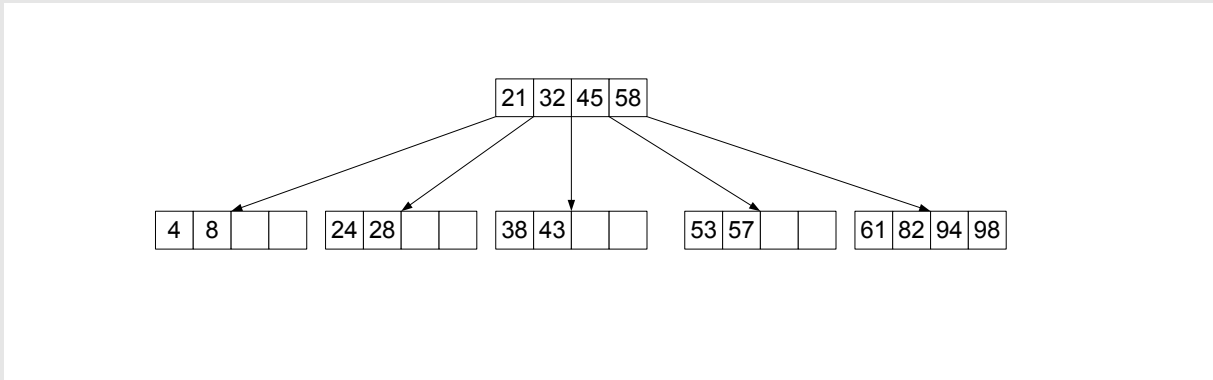
### Lösungsvorschlag:

- Schlüsselwert 52 (einfaches Löschen)
- Schlüsselwert 34 (einfaches Löschen)
- Schlüsselwert 97 (Mischen)

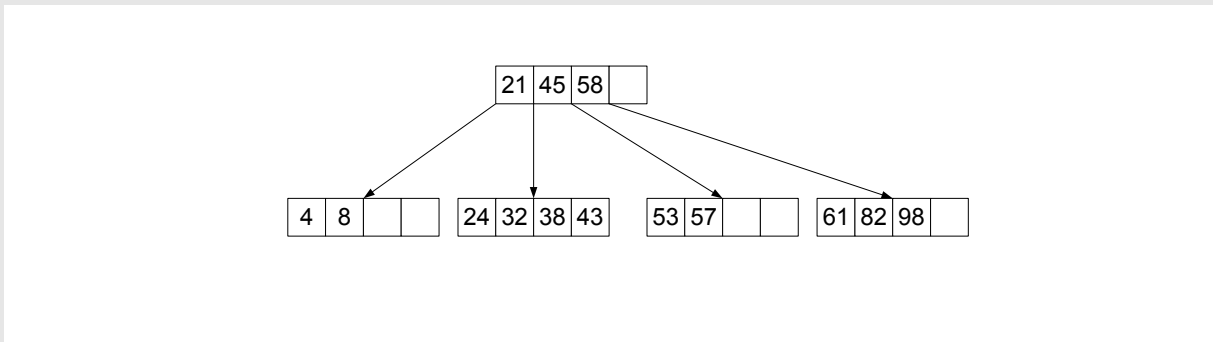


Schlüsselwert 41 (Ausgleichen)

	Lehrveranstaltung	Grundlagen von Datenbanken		WS 2019/20
	Aufgabenzettel	6 (Lösungsvorschläge)		
	Gesamtpunktzahl	30		
	Ausgabe	Mi. 08.01.2020	Abgabe	Fr. 24.01.2020



Schlüsselwert 94 (einfaches Löschen)  
 Schlüsselwert 28 (Mischen)




### 3 Präsenzaufgabe: Berechnungen in B-Bäumen

Ein B-Baum der Klasse  $\tau(4, h)$  enthält 60 Datensätze. Welche Werte kann  $h$  annehmen? Geben Sie den Rechenweg an.

#### Lösungsvorschlag:


$h \in \{2, 3\}$ , denn für einen maximal befüllten B-Baum der Klasse  $\tau(4, h)$  gilt:

1. Ebene: 1 Wurzel mit 8 Einträgen (insgesamt 1 Knoten und 8 Einträge)
  2. Ebene: 9 Knoten mit je 8 Einträgen (insgesamt  $1 + 9 = 10$  Knoten und 80 Einträge)
- $\Rightarrow h_{min} = 2$

	Lehrveranstaltung	<b>Grundlagen von Datenbanken</b>		WS 2019/20
	Aufgabenzettel	<b>6 (Lösungsvorschläge)</b>		
	Gesamtpunktzahl	<b>30</b>		
	Ausgabe	<b>Mi. 08.01.2020</b>	Abgabe	<b>Fr. 24.01.2020</b>

Und für einen minimal befüllten B-Baum der Klasse  $\tau(4, h)$  gilt:

1. Ebene: 1 Wurzel mit 1 Eintrag
  2. Ebene: 2 Knoten mit je 4 Einträgen (8 Einträge; insgesamt: 9 Einträge)
  3. Ebene: 10 Knoten mit je 4 Einträgen (40 Einträge; insgesamt: 49 Einträge)
  4. Ebene: 50 Knoten mit je 4 Einträgen (200 Einträge; insgesamt: 249 Einträge)
- $\Rightarrow h_{max} = 3$

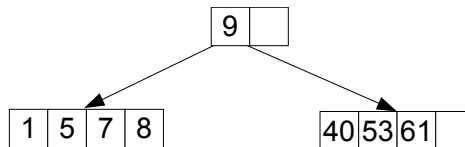
	Lehrveranstaltung	Grundlagen von Datenbanken		WS 2019/20
	Aufgabenzettel	6 (Lösungsvorschläge)		
	Gesamtpunktzahl	30		
	Ausgabe	Mi. 08.01.2020	Abgabe	Fr. 24.01.2020

#### 4 Übungsaufgabe: Einfügen und Löschen in B\*-Bäumen

[10 P.]

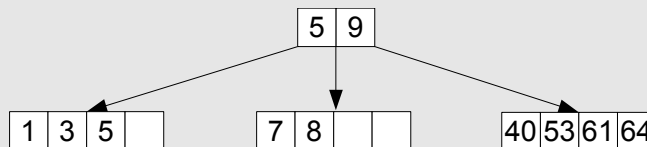
[5 P.]

- a) Gegeben ist der unten vereinfacht dargestellte **B\*-Baum** der Klasse  $\tau(1, 2, h)$ . Fügen Sie die (Datensätze mit den) Schlüsselwerte(n) **64, 3, 6, 80, 4 und 63** in dieser Reihenfolge ein. Geben Sie in jedem Einfügeschritt die verwendete Maßnahme (einfaches Einfügen in einen Knoten, Splitten) an und zeichnen Sie den Baum nach jedem Split-Vorgang neu. Als Splitfaktor wird dabei  $m = 1$  gewählt.

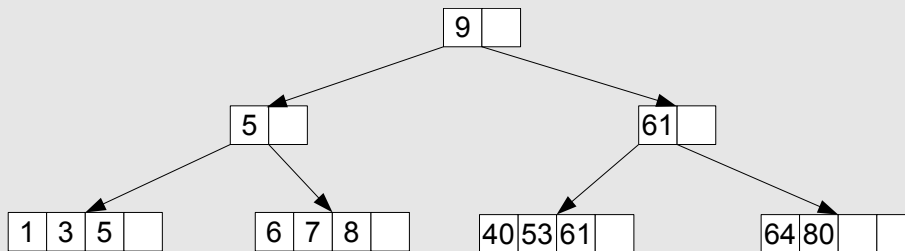


#### Lösungsvorschlag:

Schlüsselwert 64 (einfaches Einfügen)  
Schlüsselwert 3 (Splitten)



Schlüsselwert 6 (einfaches Einfügen)  
Schlüsselwert 80 (Splitten)



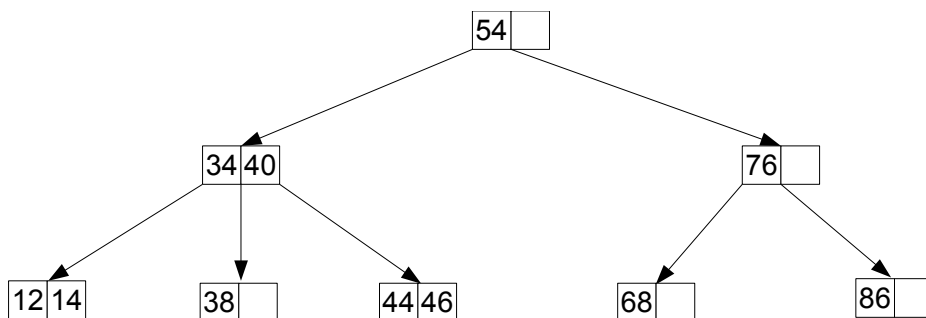
	Lehrveranstaltung	<b>Grundlagen von Datenbanken</b>		WS 2019/20
	Aufgabenzettel	<b>6 (Lösungsvorschläge)</b>		
	Gesamtpunktzahl	<b>30</b>		
	Ausgabe	<b>Mi. 08.01.2020</b>	Abgabe	<b>Fr. 24.01.2020</b>

Schlüsselwert 4 (einfaches Einfügen)  
 Schlüsselwert 63 (einfaches Einfügen)



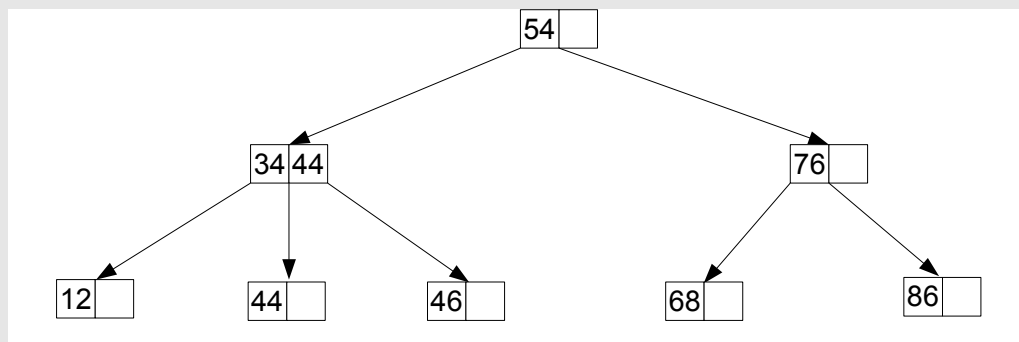
Lehrveranstaltung	Grundlagen von Datenbanken		WS 2019/20
Aufgabenzettel	6 (Lösungsvorschläge)		
Gesamtpunktzahl	30		
Ausgabe	Mi. 08.01.2020	Abgabe	Fr. 24.01.2020

- b) Gegeben ist der unten vereinfacht dargestellte **B\*-Baum** der Klasse  $\tau(1, 1, h)$ . Löschen Sie die (Datensätze mit den) Schlüsselwert(n) **14, 38, 12 und 44** in dieser Reihenfolge. Geben Sie in jedem Löschschritt die verwendete Maßnahme (einfaches Löschen, Mischen, Ausgleichen) an und zeichnen Sie den Baum nach jeder Veränderung der Knotenstruktur (Mischen, Ausgleichen) neu. Für Ausgleichsoperationen sollen nur unmittelbare Nachbarknoten herangezogen werden. [5 P.]




#### Lösungsvorschlag:

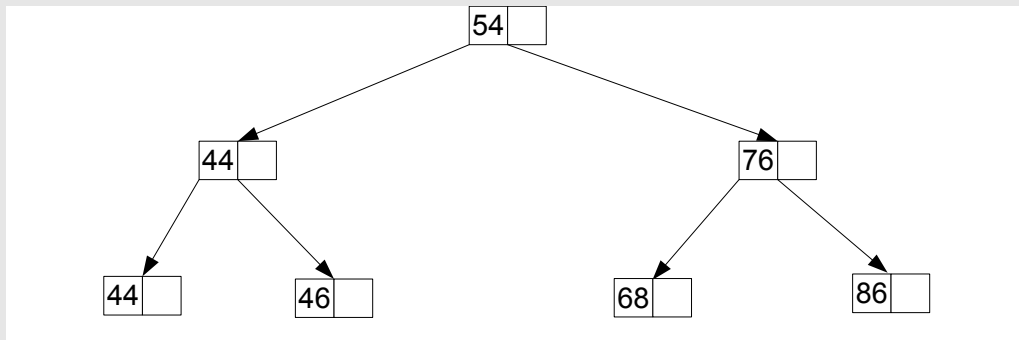
Schlüsselwert 14 (einfaches Löschen)  
Schlüsselwert 38 (Ausgleichen)



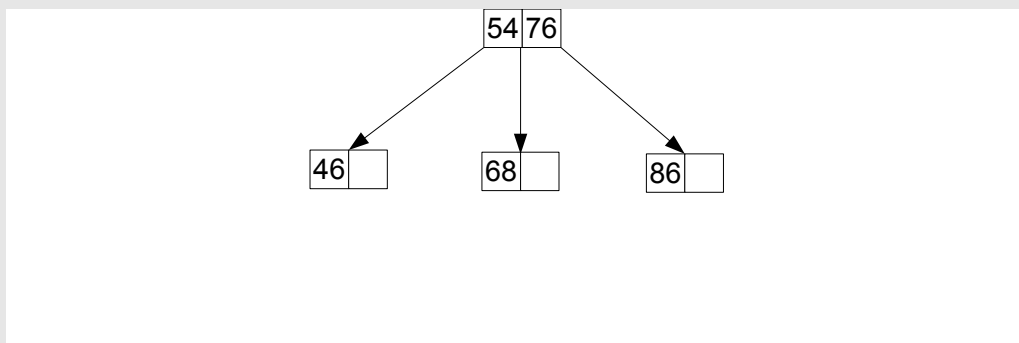


	Lehrveranstaltung	<b>Grundlagen von Datenbanken</b>		WS 2019/20
	Aufgabenzettel	<b>6 (Lösungsvorschläge)</b>		
	Gesamtpunktzahl	<b>30</b>		
	Ausgabe	<b>Mi. 08.01.2020</b>	Abgabe	<b>Fr. 24.01.2020</b>

Schlüsselwert 12 (Mischen)



Schlüsselwert 44 (Mischen auf unterer Ebene und Mischen auf nächsthöherer Ebene)



## 5 Übungsaufgabe: Berechnungen in B\*-Bäumen

[5 P.]


a) Gegeben ist ein B\*-Baum der Klasse  $\tau(k, k^*, h^*)$  mit  $k = 3$ ,  $k^* = 5$  und  $h^* = 4$ .

[4 P.]

i) Wieviele Datensätze kann dieser Baum maximal haben? Geben Sie den Rechenweg an.

**Lösungsvorschlag:**

1. Ebene: 1 Wurzel mit 6 Einträgen und 7 Kindern
2. Ebene: 7 Knoten mit jeweils 6 Einträgen und 7 Kindern

	Lehrveranstaltung	<b>Grundlagen von Datenbanken</b>		WS 2019/20
	Aufgabenzettel	<b>6 (Lösungsvorschläge)</b>		
	Gesamtpunktzahl	<b>30</b>		
	Ausgabe	<b>Mi. 08.01.2020</b>	Abgabe	<b>Fr. 24.01.2020</b>

3. Ebene:  $7 \cdot 7 = 49$  Knoten mit jeweils 6 Einträgen und 7 Kindern  
 4. Ebene:  $49 \cdot 7 = 343$  Blattknoten mit jeweils 10 Einträgen  
 $\Rightarrow d_{max} = 343 \cdot 10 = 3430$  Datensätze

ii) Wieviele Datensätze muss dieser Baum mindestens haben? Geben Sie den Rechenweg an.

#### Lösungsvorschlag:

1. Ebene: 1 Wurzel mit 1 Eintrag und 2 Kindern  
 2. Ebene: 2 Knoten mit jeweils 3 Einträgen und 4 Kindern  
 3. Ebene:  $2 \cdot 4 = 8$  Knoten mit jeweils 3 Einträgen und 4 Kindern  
 4. Ebene:  $8 \cdot 4 = 32$  Blattknoten mit jeweils 5 Einträgen  
 $\Rightarrow d_{min} = 32 \cdot 5 = 160$  Datensätze

- b) Ein B\*-Baum der Klasse  $\tau(k, k^*, h^*)$  mit  $k = 3$  und  $k^* = 1$  enthält genau 42 Blattknoten. Alle Blattknoten sind vollbesetzt. Wieviele Datensätze enthält der Baum? [1 P.]

#### Lösungsvorschlag:

Datensätze sind nur in den Blattknoten zu finden. Der Baum enthält folglich  $2 \cdot 42 = 84$  Datensätze.

## 6 Übungsaufgabe: Normalformenlehre

[7 P.]

Gegeben ist die Relation R mit den Attributen A, B, C, D und E, sowie der Menge F an funktionalen Abhängigkeiten

$$F = \{FA_1, FA_2, FA_3\}.$$

Die Wertebereiche der Attribute sind alle atomar.

$$R(A, B, C, D, E)$$


$$FA_1 = A \rightarrow B, C, D$$

$$FA_2 = C \rightarrow A, E$$

$$FA_3 = D \rightarrow A, C$$

- a) Bestimmen Sie die Schlüsselkandidaten von R bezüglich F.

[2 P.]

	Lehrveranstaltung	<b>Grundlagen von Datenbanken</b>		WS 2019/20
	Aufgabenzettel	<b>6 (Lösungsvorschläge)</b>		
	Gesamtpunktzahl	<b>30</b>		
	Ausgabe	<b>Mi. 08.01.2020</b>	Abgabe	<b>Fr. 24.01.2020</b>

**Lösungsvorschlag:**

Schlüsselkandidat 1: A  
 Schlüsselkandidat 2: C  
 Schlüsselkandidat 3: D

- b) Bestimmen Sie die Nicht-Primärattribute (Nicht-Schlüsselattribute) von R bezüglich F. [2 P.]

**Lösungsvorschlag:**

Nicht-Primärattribut 1: B  
 Nicht-Primärattribut 2: E

- c) Nehmen Sie an, dass einer der in Aufgabenteil a) ermittelten Schlüsselkandidaten als Primärschlüssel verwendet wird. In welchen Normalformen befindet sich das Relationenschema R bezüglich F? Begründen Sie Ihre Antwort, indem Sie darlegen, warum sich das Relationenschema in genau diesen Normalformen befindet und warum die anderen Normalformen nicht vorliegen. [3 P.]

**Hinweis:** Betrachten Sie dabei lediglich die 1., 2. und 3. Normalform.

**Lösungsvorschlag:**


Das Relationenschema befindet sich in der 3. Normalform, denn:

- Attributwerte sind atomar => 1. NF
- Keines der Nicht-Primärattribute B und E hängt partiell von einem der Schlüsselkandidaten ab; dass es keine partiellen Abhängigkeiten geben kann, wird bereits dadurch offensichtlich, dass es keinen Schlüsselkandidaten mit mehr als einem Attribut gibt => 2. NF
- Keines der Nicht-Primärattribute B und E hängt transitiv von einem der Schlüsselkandidaten ab => 3. NF

**7 Übungsaufgabe:  
 SQL-Anfragen**

[8 P.]

Gegeben sei das folgende Formel 1 Datenbankschema inklusive Instanz:

	Lehrveranstaltung	<b>Grundlagen von Datenbanken</b>		WS 2019/20
	Aufgabenzettel	<b>6 (Lösungsvorschläge)</b>		
	Gesamtpunktzahl	<b>30</b>		
	Ausgabe	<b>Mi. 08.01.2020</b>	Abgabe	<b>Fr. 24.01.2020</b>

<u>RennfahrerIn</u>	<u>RID</u>	Vorname	Nachname	Geburt	Wohnort	<u>Rennstall</u>
	4	Sebastian	Vettel	1987-07-03	Kemmental (Schweiz)	2
	6	Fernando	Alonso	1981-07-29	Lugano (Schweiz)	5
	8	Marc	Webber	1976-08-27	Aston Clinton (UK)	2
	9	Lewis	Hamilton	1985-01-07	Genf (Schweiz)	31
	20	Jenson	Button	1980-01-19	Monte Carlo (Monaco)	31
	21	Felipe	Massa	1982-04-25	São Paulo (Brasilien)	5
	44	Brendon	Hartley	1989-11-10	Palmerston North (Neuseeland)	2

Rennstall → Rennstall.RSID


<u>Rennstall</u>	<u>RSID</u>	Name	TeamchefIn	Budget
	2	Red Bull	Christian Horner	120
	5	Ferrari	Stefano Domenicali	220
	31	McLaren	Martin Whitmarsh	220

<u>Rennort</u>	<u>OID</u>	Name	Strecke
	4	Brasilien GP	Autodromo Jose Carlos Pace, Interlagos
	15	Abu Dhabi GP	Yas Marina Circuit
	21	Großbritannien GP	Silverstone Grand Prix Circuit

<u>Platzierung</u>	<u>RID</u>	<u>OID</u>	Platz
	8	4	2
	4	15	1
	20	15	3
	4	4	1
	6	4	3
	8	15	8
	6	21	14
	9	15	2
	9	4	4
	21	15	10
	20	4	5
	21	4	15
	6	15	7

RID → RennfahrerIn.RID, OID → Rennorte.OID

Formulieren Sie entsprechende SQL-Anweisungen für die in den nachfolgenden Teilaufgaben angeführten natürlichsprachlich formulierten Mengenbeschreibungen. **Verwenden Sie den in der Vorlesung verwendeten**

	Lehrveranstaltung	<b>Grundlagen von Datenbanken</b>		WS 2019/20
	Aufgabenzettel	<b>6 (Lösungsvorschläge)</b>		
	Gesamtpunktzahl	<b>30</b>		
	Ausgabe	<b>Mi. 08.01.2020</b>	Abgabe	<b>Fr. 24.01.2020</b>

**SQL-Standard.** Das SQL-Schlüsselwort JOIN darf dabei aber nur zur Spezifizierung eines äusseren Verbundes verwendet werden. Testen Sie die SQL-Ausdrücke auf der Übungsdatenbank.

**Hinweis:** Die zum Testen benötigten Schema- und Instanzdaten können mit dem auf der Veranstaltungsseite bereitgestellten SQL-Skript erstellt bzw. eingefügt werden.

- a) Zu jedem Rennstall dessen Namen und die Gesamtzahl der zu diesem Rennstall gehörenden FahrerInnen.

**Lösungsvorschlag:**

```
SELECT Name, COUNT(*) AS Anzahl_FahrerInnen
FROM RennfahrerIn, Rennstall
WHERE RSID = Rennstall
GROUP BY RSID, Name;
```

- b) Vor- und Nachnamen der RennfahrerInnen, die keine Platzierung erlangt haben.

**Lösungsvorschlag:**

```
SELECT R.Vorname, R.Nachname
FROM RennfahrerIn R
WHERE R.RID NOT IN(SELECT P.RID
FROM Platzierung P);
```

- c) Zu jeder RennfahrerIn (egal, ob jemals platziert oder nicht), deren Nachnamen und deren bester erreichter Platz, nach Nachname aufsteigend sortiert.

**Lösungsvorschlag:**

```
SELECT R.Nachname, MIN(P.Platz) AS Bester_Platz
FROM RennfahrerIn R LEFT OUTER JOIN Platzierung P ON P.RID = R.RID
GROUP BY R.RID, R.Nachname
ORDER BY R.Nachname ASC;
```

- d) Alle Informationen zu RennfahrerInnen, die zwischen dem 01.01.1980 und dem 01.01.1985 geboren wurden und bei deren Nachname der zweite Buchstabe ein 'a' ist.

**Lösungsvorschlag:**

```
SELECT *
FROM RennfahrerIn
WHERE Geburt BETWEEN '1980-01-01' AND '1985-01-01'
AND Nachname LIKE '_a%';
```