

# Schema “Bibliothek” (LOTS)

autor (autorid, nachname, vornamen, zusatz)

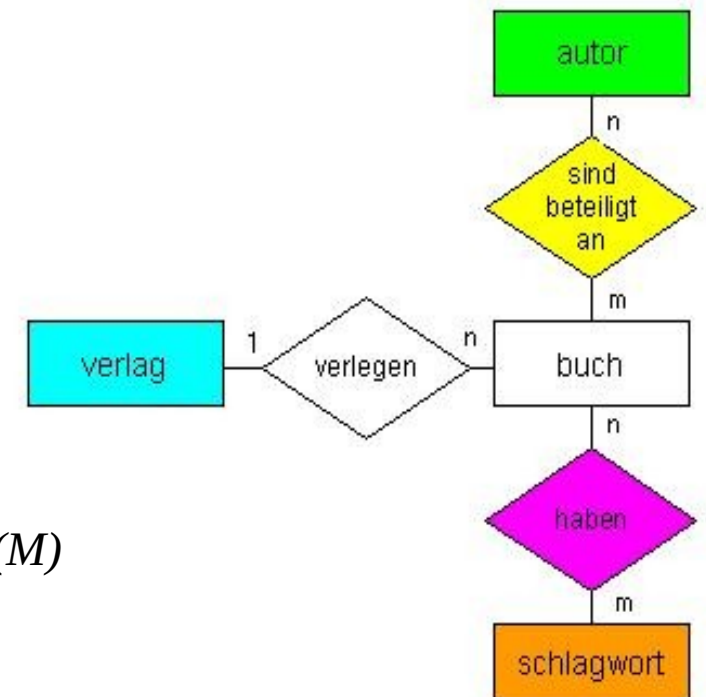
verlag (verlagsid, name, ort)

buch (buchid, titel, isbn, auflage, jahr, preis, signatur, verlagsid)

schlagwort (swid, schlagwort)

buch\_aut (buchid, autorid, rolle, rang)

buch\_sw (buchid, swid)



*Rolle in buch\_aut: Herausgeber (H), Verfasser (V), Übersetzer (U), Mitarbeiter (M)*

*Rang in buch\_aut: Position des Autors in Liste v. mehreren Autoren*

<http://lots.uni-leipzig.de/sql-training/>

# Frage B1

Welche Autoren (Name, Vornamen) waren bei ausschließlich in Berlin herausgegebenen Büchern als Herausgeber beteiligt?

Verhindern Sie Doppelausgaben und ordnen Sie die Ergebnismenge nach den Autornamen. (8)

## Frage B2

Welche Autoren (Name, Vornamen), deren Name mit 'Gold' beginnt, waren nie als Herausgeber an Büchern beteiligt?

Ordnen Sie die Ergebnismenge nach dem Autornamen und -vornamen. (9)

## Frage B3

Erstellen Sie eine Liste der Verlage - absteigend sortiert nach der Anzahl der im Verlag erschienen Bücher. (403)

# Frage B4

Welche Bücher haben im Schlagwort den String “Unix”? (4)

# Lösung B1

```
SELECT DISTINCT A.nachname, A.vornamen  
FROM autor A NATURAL JOIN buch_aut BA NATURAL JOIN buch B  
NATURAL JOIN V  
WHERE V.ort='Berlin'  
AND BA.rolle='h'  
ORDER BY A.nachname
```

# Lösung B2

```
SELECT A.nachname, A.vorname
FROM autor A
WHERE A.nachname LIKE 'Gold%'
AND NOT EXISTS (
  SELECT *
FROM buch_aut BA
WHERE rolle='h'
AND A.autorid= BA.autorid
)
ORDER BY A.nachname, A.vorname
```

# Lösung B3

```
SELECT verlag.name, COUNT(*)  
FROM verlag, buch  
WHERE verlag.verlagsid = buch.verlagsid  
GROUP BY verlag.name  
ORDER BY COUNT(*) desc
```



# Lösung B4

```
SELECT schlagwort, titel
FROM buch NATURAL JOIN buch_sw NATURAL JOIN schlagwort
WHERE lower(schlagwort) LIKE '%unix%'
```

# Schema “Parteien”

**Parteien**(Name, Mitgliederanz)

**Direktkandidaten**(SozVnr, Name, Partei)  
Partei FK auf Parteien(Name)

**Erststimmen**(Wahlbezirk, Jahr, Kandidat, AnzStimmen)  
Wahlbezirk FK auf Wahlbezirke  
Kandidat FK auf Direktkandidaten

**Zweitstimmen**(Wahlbezirk, Jahr, Partei, AnzStimmen)  
Wahlbezirk FK auf Wahlbezirke  
Partei FK auf Parteien

**Wahlbezirke**(Nr, AnzWahlberechtigte, Wahllokal, Wahlkreis)  
Wahlkreis FK auf Wahlkreise

**Wahlkreise**(Nr, Bezeichnung, Bundesland)  
Bundesland FK auf Bundeslaender

**Bundeslaender**(Name, AnzEinwohner)

# Frage P1

Welche Parteien haben in dem Jahr 2020 mehr Erststimmen als Zweitstimmen bekommen?

# Frage P2

Welche Kandidaten(Name) haben die meisten Erststimmen in ihren Wahlbezirken 2020 erhalten?

# Frage P3

Welche Parteien haben in allen Wahlbezirken 2020 Stimmen bekommen?

# Frage P4

Wie hoch ist der prozentuale Anteil der Wahlberechtigten für jedes Bundesland bzgl. der Gesamteinwohneranzahl?

# Lösung P1

```
SELECT dk.Partei
FROM Direktkandidaten dk,
(SELECT Partei, SUM(Anstimmen) AS gessstimmen
FROM Erststimmen JOIN Direktkandidaten ON Kandidat =
SozVnr
GROUP BY Partei
WHERE Jahr = 2020) es,
(SELECT Partei, SUM(Anstimmen) AS gessstimmen
FROM Zweitstimmen
GROUP BY Partei
WHERE Jahr = 2020) zs
WHERE dk.Partei = zs.Partei
AND es.gessstimmen > zs.gessstimmen
```

# Lösung P2

```
SELECT Name
FROM Direktkandidaten dk, Erststimmen es
(SELECT wahlbezirk, MAX(Anzstimmen) AS maxstimme
FROM Erststimmen
GROUP BY wahlbezirk) max
WHERE es.Anzstimmen = max.maxstimmen
AND es.wahlbezirk = max.wahlbezirk
AND es.kandidat = dk.SozVnr
```



# Lösung P3 (a)

```
SELECT Partei
FROM Zweitstimme zsl
WHERE zsl.Jahr = 2020 AND zsl.AnzStimmen > 0
AND NOT EXISTS (
  SELECT *
  FROM wahlbezirk
  WHERE Nr NOT IN
    (SELECT zsl.wahlbezirk
     FROM Zweitstimmen zsl
     WHERE zsl.Partei = zsl.Partei)
)
```

# Lösung P3 (b)

```
SELECT Partei  
FROM Zweitstimmen  
WHERE AnzStimmen > 0  
AND Jahr = 2020  
GROUP BY Partei  
HAVING COUNT(*) = (  
SELECT COUNT(*)  
FROM Wahlbezirke  
)
```

# Lösung P4

```
SELECT Bundesland, (SUM(Anzahlberechtigte)/CAST(Anzahlwohner  
AS double precision))  
FROM wahlbezirke wb, wahlkreise wk, Bundeslaender bl  
WHERE wb.wahlkreis = wk.Nr  
AND wk.Bundesland = bl.Name  
GROUP BY bl.Name
```