Aufgabe 1:

SELECT f\_kategorie , o\_land , sum ( VK\_Umsatz )

FROM Fahrzeug as F , Ort as O , Verkauf as V

WHERE F. F\_ID = V. F\_ID

AND O . O\_ID = V . O\_ID

GROUP BY CUBE ( f\_kategorie , o\_land )

ORDER BY f\_kategorie , o\_land ;

* Sum(VK\_Umsatz) berechnet den Gesamtumsatz für Kombinationen aus Fahrzeugkategorien f\_kategorie und Ländern o\_land
* FROM Fahrzeug as F , Ort as O , Verkauf as V

WHERE F. F\_ID = V. F\_ID

AND O . O\_ID = V . O\_ID ist äquivalent zu   
FROM FAHRZEUG as F join Verkauf as V on F.F\_ID = V.F\_ID  
JOIN ORT as O on O.O\_ID = V.O\_ID

* GROUP BY CUBE (f\_kategorie, o\_land) erstellt Gruppierungssätze für
  + Jeden Wert von f\_kategorie einzeln
  + Jeden Wert von o\_land einzeln
  + Die kombination von f\_kategorie und o\_land
  + Die Gesamtsumme, aggregiert beide f\_kategorie und o\_land
* ORDER BY f\_kategorie, o\_land ordnen die Tabelle nach f\_kategorie und dann o\_land

A screenshot of a computer

Description automatically generated  
  
CUBE-Operator duch ROLLUP-Operator ersetzen:

SELECT f\_kategorie, o\_land, sum(VK\_Umsatz)

FROM Fahrzeug as F, Ort as O, Verkauf as V

WHERE F.F\_ID = V.F\_ID AND O.O\_ID = V.O\_ID

GROUP BY ROLLUP (f\_kategorie, o\_land)

ORDER BY f\_kategorie, o\_land;

A screenshot of a computer

Description automatically generated

Mit ROLLUP erhält man

* Umsatz pro Kombination von Fahrzeugkategorie und Land
* Umsatz pro Fahrzeugkategorie (mit allen Ländern zusammengefasst)
* Gesamtumsatz ohne Berücksichtigung der Kategorie oder des Landes

Aufgabe 2:

1. SELECT z\_jahr, f\_kategorie, o\_land, sum(VK\_Umsatz)

FROM Fahrzeug as F, Ort as O, Verkauf as V, Zeit as z

WHERE F.F\_ID = V.F\_ID AND O.O\_ID = V.O\_ID AND z\_jahr < 2021

GROUP BY ROLLUP ((z\_jahr, f\_kategorie), o\_land)

ORDER BY z\_jahr, f\_kategorie, o\_land;

1. SELECT z\_jahr, f\_kategorie, o\_land, sum(VK\_Umsatz)

FROM Fahrzeug as F, Ort as O, Verkauf as V, Zeit as z

WHERE F.F\_ID = V.F\_ID AND O.O\_ID = V.O\_ID AND z\_jahr < 2021

GROUP BY CUBE (z\_jahr, f\_kategorie, o\_land)

ORDER BY z\_jahr, f\_kategorie, o\_land;

Aufgabe 3:

Base:

SELECT f\_id , (z. z\_tag || '. ' || z. z\_monat || '. ' || z. z\_jahr ) as datum ,

SUM ( VK\_Umsatz ) AS Summe\_am\_Bestelldatum

FROM Verkauf as v , zeit as z

WHERE f\_id =12 and z. z\_id = v. z\_id

GROUP BY ( f\_id , datum )

ORDER BY f\_id , datum DESC ;

Geordnet:  
  
SELECT f\_id , (z. z\_tag || '. ' || z. z\_monat || '. ' || z. z\_jahr ) as datum ,

SUM ( VK\_Umsatz ) AS Summe\_am\_Bestelldatum

FROM Verkauf as v , zeit as z

WHERE f\_id =12 and z. z\_id = v. z\_id

GROUP BY ( f\_id , datum , z.z\_jahr, z.z\_monat, z.z\_tag)

ORDER BY f\_id , z.z\_jahr, z.z\_monat, z.z\_tag DESC ;

Beispiel partition:

select v.f\_id,f.f\_bezeichnung, v.vk\_umsatz ,z.z\_jahr, z.z\_monat, z.z\_tag, SUM (v.vk\_umsatz) OVER (PARTITION BY z.z\_jahr, z.z\_monat)

from verkauf as v join zeit as z on v.z\_id = z.z\_id

join fahrzeug as f on v.f\_id = f.f\_id

order by z.z\_jahr, z.z\_monat, z.z\_tag asc;

SELECT f\_id, (z.z\_tag || '. ' || z.z\_monat || '. ' || z.z\_jahr) AS datum,

SUM(VK\_Umsatz) AS Summe\_am\_Bestelldatum,

SUM(SUM(VK\_Umsatz)) OVER(PARTITION BY f\_id) AS Gesamtsumme\_pro\_Fahrzeug

FROM Verkauf AS v, zeit AS z

WHERE f\_id = 12 AND z.z\_id = v.z\_id

GROUP BY f\_id, datum,

z.z\_jahr, z.z\_monat, z.z\_tag

ORDER BY f\_id, z.z\_jahr, z.z\_monat, z.z\_tag DESC;

Aufgabe 4

1. SELECT F\_ID,

SUM(VK\_UMSATZ) AS Gesamtumsatz,

COUNT(VK\_ID) AS Anzahl\_Verkaeufe

FROM VERKAUF

GROUP BY F\_ID

ORDER BY Gesamtumsatz DESC

LIMIT 1;

1. SELECT O\_LAND, SUM(VK\_UMSATZ) AS Gesamtumsatz

FROM VERKAUF

JOIN ORT ON VERKAUF.O\_ID = ORT.O\_IDGROUP BY O\_LAND

ORDER BY Gesamtumsatz DESC

LIMIT 1;

1. SELECT V.F\_ID, F.F\_BEZEICHNUNG,

SUM(VK\_UMSATZ) / (SELECT SUM(VK\_UMSATZ) FROM VERKAUF) \* 100 AS Umsatzanteil

FROM VERKAUF as V JOIN FAHRZEUG as F on V.F\_ID = F.F\_ID

GROUP BY V.F\_ID, F.F\_BEZEICHNUNG

ORDER BY Umsatzanteil ASC;

Soll mit window funktion machen

1. SELECT V\_ID, SUM(VK\_UMSATZ) / (SELECT SUM(VK\_UMSATZ) FROM VERKAUF) \* 100 AS Umsatzanteil

FROM VERKAUF

GROUP BY V\_ID;

1. SELECT

FahrzeugKategorie,

Monat, SUM(MonatlicherUmsatz) OVER (PARTITION BY FahrzeugKategorie ORDER BY Monat

ROWS BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW) AS Kumulierter\_Umsatz

FROM

(SELECT

FAHRZEUG.F\_KATEGORIE AS FahrzeugKategorie,

ZEIT.Z\_MONAT AS Monat,

SUM(VERKAUF.VK\_UMSATZ) AS MonatlicherUmsatz

FROM VERKAUF

JOIN FAHRZEUG ON VERKAUF.F\_ID = FAHRZEUG.F\_ID

JOIN ZEIT ON VERKAUF.Z\_ID = ZEIT.Z\_ID

WHERE ZEIT.Z\_JAHR = 2020

GROUP BY FAHRZEUG.F\_KATEGORIE, ZEIT.Z\_MONAT) AS MonatlicheUmsaetze

ORDER BY FahrzeugKategorie, Monat;

1. WITH KundenUmsatz AS (

SELECT O.O\_FILIALE, K.K\_ID, K.K\_NAME,

SUM(V.VK\_UMSATZ) AS Gesamtumsatz,

STRING\_AGG(F.F\_BEZEICHNUNG, ', ') AS GekaufteFahrzeuge,

ROW\_NUMBER() OVER (PARTITION BY O.O\_FILIALE ORDER BY SUM(V.VK\_UMSATZ) DESC) AS Rang

FROM VERKAUF V

JOIN KUNDE K ON V.K\_ID = K.K\_ID

JOIN ORT O ON V.O\_ID = O.O\_ID

JOIN FAHRZEUG F ON V.F\_ID = F.F\_ID

GROUP BY O.O\_FILIALE, K.K\_ID, K.K\_NAME)

SELECT O\_FILIALE, K\_ID, K\_NAME, Gesamtumsatz, GekaufteFahrzeuge, Rang

FROM KundenUmsatz

WHERE Rang <= 4

ORDER BY O\_FILIALE, Rang;

1. SELECT Z\_MONAT, O\_FILIALE, SUM(VK\_UMSATZ) AS Monatsumsatz,RANK() OVER (PARTITION BY Z\_MONAT ORDER BY SUM(VK\_UMSATZ) DESC) AS Rang

FROM VERKAUF

JOIN ORT ON VERKAUF.O\_ID = ORT.O\_ID

JOIN ZEIT ON VERKAUF.Z\_ID = ZEIT.Z\_ID

GROUP BY Z\_MONAT, O\_FILIALE;

1. WITH VerkaeufeProFiliale AS (

SELECT O.O\_FILIALE, COUNT(VK\_ID) AS AnzahlVerkaeufe

FROM VERKAUF V

JOIN ORT O ON V.O\_ID = O.O\_ID

GROUP BY O.O\_FILIALE),

VerkaeuferProFiliale AS (

SELECT O.O\_FILIALE, COUNT(DISTINCT V.V\_ID) AS AnzahlVerkaeufer

FROM VERKAUF V

JOIN ORT O ON V.O\_ID = O.O\_ID

GROUP BY O.O\_FILIALE)

SELECT

VPF.O\_FILIALE,

VPF.AnzahlVerkaeufer / VFF.AnzahlVerkaeufe AS Verhaeltnis

FROM VerkaeuferProFiliale VPF

JOIN VerkaeufeProFiliale VFF ON VPF.O\_FILIALE = VFF.O\_FILIALE

ORDER BY Verhaeltnis DESC

LIMIT 1;

1. SELECT Z\_JAHR, Z\_MONAT,

SUM(VK\_UMSATZ) / (SELECT SUM(VK\_UMSATZ)

FROM VERKAUF

JOIN ZEIT ON VERKAUF.Z\_ID = ZEIT.Z\_ID

WHERE Z\_JAHR = ZEIT.Z\_JAHR) \* 100 AS Umsatzanteil

FROM VERKAUF

JOIN ZEIT ON VERKAUF.Z\_ID = ZEIT.Z\_ID

WHERE Z\_JAHR IN (2020, 2021)

GROUP BY Z\_JAHR, Z\_MONAT;

OPTINAL: KLAUSUR