

```

-- Aufgabe 8 Umsatzanteil eines Artikels
-- Ermitteln Sie den Umsatzanteil eines Artikels innerhalb der
Warengruppe in Prozent.
-- Ergebnisspalten: ArtikelNr, Artikel, GruppenNr, Warengruppe, Umsatz,
WGUmsatz, Anteil; Ausgabe sortiert nach Warengruppe und ArtikelNr
-- Gehen Sie dabei schrittweise wie folgt vor:
-- a) Schreiben Sie eine Anfrage zur Ermittlung des Gesamtumsatzes
(Bestellmenge*Verkaufspreis) pro Artikel. Ergebnisspalten: ArtikelNr,
Artikel (Name), GruppenNr, Umsatz
SELECT
    Artikel.ArtikelNr,
    Artikel.Artikelname AS Artikel,
    Artikel.GruppenNr,
    COALESCE(SUM(Auftragspositionen.Bestellmenge *
Auftragspositionen.Verkaufspreis), 0) AS Umsatz
FROM Artikel
(LEFT JOIN Auftragspositionen ON Artikel.ArtikelNr =
Auftragspositionen.ArtikelNr
GROUP BY
    Artikel.ArtikelNr,
    Artikel.Artikelname,
    Artikel.GruppenNr;

-- b) Schreiben Sie eine Anfrage zur Ermittlung des Gesamtumsatzes pro
Warengruppe.
-- Ergebnisspalten: GruppenNr, Warengruppe, WGUmsatz.
SELECT
    Artikel.GruppenNr,
    Warengruppen.GruppenName AS Warengruppe,
    COALESCE(SUM(Auftragspositionen.Bestellmenge *
Auftragspositionen.Verkaufspreis), 0) AS WGUmsatz
FROM Artikel
JOIN Warengruppen ON Artikel.GruppenNr = Warengruppen.GruppenNr
(LEFT JOIN Auftragspositionen ON Artikel.ArtikelNr =
Auftragspositionen.ArtikelNr
GROUP BY
    Artikel.GruppenNr,
    Warengruppen.GruppenName;

-- c) Verwenden Sie diese beiden als Unterabfragen und verbinden Sie sie
mit einem Join über GruppenNr, um das Endergebnis zu berechnen.
-- Überprüfen Sie z.B. bei der Warengruppe "Trains", ob die Anteile der
Artikel sich auf 100% aufsummieren.
SELECT
    Artikel.ArtikelNr,
    Artikel.Artikelname AS Artikel,
    Artikel.GruppenNr,
    Warengruppen.GruppenName AS Warengruppe,
    COALESCE(SUM(Auftragspositionen.Bestellmenge *
Auftragspositionen.Verkaufspreis), 0) AS Umsatz,
    WarengruppenUmsatz.WGUmsatz,
    COALESCE((SUM(Auftragspositionen.Bestellmenge *
Auftragspositionen.Verkaufspreis) / WarengruppenUmsatz.WGUmsatz) * 100,
0) AS Anteil
FROM Artikel
(LEFT JOIN Auftragspositionen ON Artikel.ArtikelNr =
Auftragspositionen.ArtikelNr
JOIN Warengruppen ON Artikel.GruppenNr = Warengruppen.GruppenNr
JOIN
    (SELECT

```

```

        Artikel.GruppenNr,
        SUM(Auftragspositionen.Bestellmenge *
Auftragspositionen.Verkaufspreis) AS WGUmsatz
FROM
    Artikel
LEFT JOIN
    Auftragspositionen ON Artikel.ArtikelNr =
Auftragspositionen.ArtikelNr
GROUP BY
    Artikel.GruppenNr) AS WarengruppenUmsatz ON Artikel.GruppenNr =
WarengruppenUmsatz.GruppenNr
GROUP BY
    Artikel.ArtikelNr,
    Artikel.Artikelname,
    Artikel.GruppenNr,
    Warengruppen.GruppenName,
    WarengruppenUmsatz.WGUmsatz
ORDER BY
    Warengruppen.GruppenName,
    Artikel.ArtikelNr;

```

-- 2. Umgang mit Funktionen: Es soll eine Analyse zu Lieferzeiten gemacht werden.

-- a) Lassen Sie sich pro Auftrag folgende Angaben ausgeben: KundenNr, Auftragsdatum, Lieferdatum, Lieferzeit

-- Hinweis: Die Lieferzeit lässt sich mit Hilfe von DATEDIFF aus Zustell- und Auftragsdatum ermitteln.

```

SELECT
    Auftraege.KundenNr,
    Auftraege.Auftragsdatum,
    Auftraege.Lieferdatum,
    DATEDIFF(Auftraege.Lieferdatum, Auftraege.Auftragsdatum) AS
Lieferzeit
FROM
    Auftraege;

```

-- b) Lassen Sie sich eine weitere Spalte "Lieferklasse" ausgeben. In dieser Spalte soll "langsam" stehen, wenn die Lieferzeit 4 Tage und mehr beinhaltet; ansonsten "schnell".

-- Hinweis: Verwenden Sie einen CASE-Ausdruck.: Lassen Sie sich die durchschnittliche Lieferzeit in Tagen pro Lieferklasse, gerundet auf zwei Nachkommastellen ausgeben.

-- Hinweise: Verwenden Sie Ihr Abfrageergebnis aus der vorigen Aufgabe als Unterabfrage. Runden geht mit ROUND.

```

SELECT
    Auftraege.KundenNr,
    Auftraege.Auftragsdatum,
    Auftraege.Lieferdatum,
    DATEDIFF(Auftraege.Lieferdatum, Auftraege.Auftragsdatum) AS
Lieferzeit,
    CASE
        WHEN DATEDIFF(Auftraege.Lieferdatum, Auftraege.Auftragsdatum) >=
4 THEN 'langsam'
        ELSE 'schnell'
    END AS Lieferklasse
FROM
    Auftraege;

```

```

SELECT
    Lieferklasse,

```

```

        ROUND(AVG(Lieferzeit), 2) AS DurchschnittlicheLieferzeit
FROM
    (SELECT
        Auftraege.KundenNr,
        Auftraege.Auftragsdatum,
        Auftraege.Lieferdatum,
        DATEDIFF(Auftraege.Lieferdatum, Auftraege.Auftragsdatum) AS
Lieferzeit,
        CASE
            WHEN DATEDIFF(Auftraege.Lieferdatum, Auftraege.Auftragsdatum)
>= 4 THEN 'langsam'
            ELSE 'schnell'
        END AS Lieferklasse
    FROM
        Auftraege) AS Lieferzeiten
GROUP BY
    Lieferklasse;

```

-- 3. Ermitteln Sie mit einer SQL-Abfrage die Nummern der Kunden, die noch nicht alles bezahlt haben, d.h. bei denen der Gesamtwert aller Bestellungen (=Bestellmenge*Verkaufspreis) mit Status "in Bearbeitung" oder "zugestellt" größer ist als die Gesamtsumme aller Zahlungen. Ergebnisspalten: KundenNr, Bestellwert, Gesamtzahlung, Differenz (zwischen Bestellwert und Zahlung)

-- Hinweise: • Bauen Sie die Anfrage schrittweise auf: Erstellen Sie zwei Unterabfragen, eine für den Gesamtwert der Aufträge pro KundenNr und eine für die Gesamtsumme der Zahlungen pro KundenNr und verbinden Sie diese dann.

-- • Behandeln Sie den Fall, dass ein Kunde etwas bestellt, aber noch gar keine Zahlung geleistet hat, korrekt! Nullwerte einer numerischen Spalte können mit folgendem Ausdruck in numerische Werte umgewandelt werden: "COALESCE(<column>, 0)"

-- • Achtung: Es gibt in den Daten auch sehr viele Kunden, die mehr gezahlt haben als bestellt wurde. Diese Datensätze interessieren uns nicht.

```

-- Gesamtwert
SELECT Kunden.KundenNr,
    COALESCE(SUM(Auftragspositionen.Bestellmenge *
Auftragspositionen.Verkaufspreis),0) AS Bestellwert
FROM Kunden
JOIN Auftraege ON Kunden.KundenNr = Auftraege.KundenNr
JOIN Auftragspositionen ON Auftraege.AuftragsNr =
Auftragspositionen.AuftragsNr
WHERE Auftraege.Status IN ('in Bearbeitung', 'zugestellt')
GROUP BY Kunden.KundenNr;

```

```

-- Gesamtzahlung
SELECT Kunden.KundenNr,
    COALESCE(SUM(Bezahlungen.Zahlungsbetrag), 0) AS Gesamtzahlung
FROM Kunden
LEFT JOIN Bezahlungen ON Kunden.KundenNr = Bezahlungen.KundenNr
GROUP BY Kunden.KundenNr;

```

```

-- Alle Kunden mit fehlenden Zahlungen
SELECT Bestellwert.KundenNr, Bestellwert.Gesamtwert,
Gesamtwert.Gesamtzahlung,
(Bestellwert.Gesamtwert - Gesamtwert.Gesamtzahlung) as diffrenz

```

```

FROM ( SELECT Kunden.KundenNr,
Coalesce(SUM(Auftragspositionen.Bestellmenge*Auftragspositionen.Verkaufspreis),0) as Gesamtwert
FROM Kunden
JOIN Auftraege on Kunden.KundenNr = Auftraege.KundenNr AND
Auftraege.Status in ('in Bearbeitung', 'zugestellt')
JOIN Auftragspositionen ON Auftraege.AuftragsNr =
Auftragspositionen.AuftragsNr
GROUP BY Kunden.KundenNr
) AS Bestellwert
JOIN
( SELECT Kunden.KundenNr, COALESCE(SUM(Bezahlungen.zahlungsbetrag),0) as
Gesamtzahlung
From Kunden
LEFT JOIN Bezahlungen ON Kunden.KundenNr = Bezahlungen.KundenNr
GROUP BY Kunden.KundenNr) AS Gesamtwert ON Bestellwert.KundenNr =
Gesamtwert.KundenNr
WHERE Gesamtwert.Gesamtzahlung < Bestellwert.Gesamtwert;

```