

Aggregatfunktionen

ExerciselD: 194116 (category: Aggregatfunktionen, level: einfach)

Geben Sie die höchste Anzahl von Toren aus, die je in einer ersten Halbzeit während der regulären Spielzeit von einer Heimmannschaft (toret1) erzielt worden sind.

```
SELECT MAX(toret1) FROM spielstandabschnitt  
NATURAL JOIN spielabschnitt  
WHERE bezeichnung LIKE 'Erste Halbzeit';
```

ExerciselD: 194120 (category: Aggregatfunktionen, level: einfach)

Berechnen Sie die höchste Tordifferenz, die je am Ende eines Elfmeterschießens erzielt worden ist.

```
SELECT MAX(ABS(toret1 - toret2)) FROM spielstandabschnitt  
NATURAL JOIN spielabschnitt  
WHERE bezeichnung = 'Elfmeterschießen';
```

ExerciselD: 194119 (category: Aggregatfunktionen, level: einfach)

Geben Sie die Anzahl der Länder aus, die von der FIFA zum Kontinent 'Afrika' gezählt werden.

```
SELECT COUNT(*) FROM land  
JOIN fifakontinent ON kontinentid = fifakontinentid  
WHERE fifakontinentname = 'Afrika';
```

ExerciselD: 194115 (category: Aggregatfunktionen, level: einfach)

Geben Sie die Anzahl der Tore aus, die die Person mit dem Nachnamen 'Pelé' bisher in allen Weltmeisterschaften je erzielt hat.

```
SELECT COUNT(*) FROM tor  
JOIN person ON spielerid = personid  
WHERE name = 'Pelé';
```

ExerciselD: 194121 (category: Aggregatfunktionen, level: einfach)

Geben Sie die höchste Anzahl von Toren aus, die je innerhalb eines Spiels mit abschließendem Elfmeterschießen von einer Gastmannschaft (toret2) insgesamt erzielt worden sind.

```
SELECT MAX(toret2) FROM spielstandabschnitt  
NATURAL JOIN spielabschnitt  
WHERE bezeichnung LIKE 'Elfmeterschießen';
```

ExerciseID: 194114 (category: Aggregatfunktionen, level: einfach)

Geben Sie die Summe aller Tore aus, die in allen bisherigen Weltmeisterschaften in der regulären Spielzeit der ersten Halbzeit von beiden Mannschaften je erzielt worden sind.

```
SELECT SUM(toret1) + SUM(toret2) FROM spielstandabschnitt  
NATURAL JOIN spielabschnitt  
WHERE bezeichnung LIKE 'Erste Halbzeit';
```

ExerciseID: 194117 (category: Aggregatfunktionen, level: einfach)

Späßfrage! Berechnen Sie den Durchschnitt der Rückennummern aller Spieler, die jemals an einer Weltmeisterschaft teilgenommen haben.

```
SELECT AVG(rueckennummer) FROM personnominierung;
```

ExerciseID: 194118 (category: Aggregatfunktionen, level: einfach)

Geben Sie die Anzahl der Tore aus, die der Spieler mit dem Nachnamen 'Klose' bisher in allen Weltmeisterschaften je erzielt hat.

```
SELECT COUNT(*) FROM tor  
JOIN person ON tor.spielerid = person.personid  
WHERE name='Klose';
```

ExerciseID: 194122 (category: Aggregatfunktionen, level: einfach)

Geben Sie die höchste Anzahl von Toren aus, die je innerhalb eines Spiels mit abschließendem Elfmeterschießen von beiden Mannschaften zusammen erzielt worden sind.

```
SELECT MAX(toret1 + toret2) FROM spielstandabschnitt  
NATURAL JOIN spielabschnitt  
WHERE bezeichnung = 'Elfmeterschießen';
```

ExerciseID: 194123 (category: Aggregatfunktionen, level: einfach)

Geben Sie die Anzahl der Spiele aus, die durch Elfmeterschießen entschieden werden mussten.

```
SELECT COUNT(*) FROM spielstandabschnitt  
NATURAL JOIN spielabschnitt  
WHERE bezeichnung = 'Elfmeterschießen';
```

ExerciseID: 194112 (category: Aggregatfunktionen, level: mittel)

Geben Sie durchschnittliche Anzahl (mit Nachkommastellen) von Toren aus, die alle Spieler, die jemals bei einer Weltmeisterschaft ein Tor geschossen haben, erzielt haben. Benennen Sie die berechnete Spalte 'durchschnittliche_toranzahl'.

```
SELECT CAST(COUNT(*) AS FLOAT)/ CAST(COUNT (DISTINCT spielerid) AS FLOAT) AS  
'durchschnittliche_torzahl FROM tor;
```

Arithmetische Ausdrücke

ExerciseID: 194191 (category: Arithmetische Ausdr, level: einfach)

Geben Sie PersonId, Vorname, Nachname und Geburtsdatum aller Personen zusammen mit ihrem Alter (in Jahren) aus (es sei angenommen, dass alle Personen noch leben). Berechnen Sie das Jahr arithmetisch (unter der Annahme, dass jedes Jahr genau 365 Tage hat) und verwenden Sie *nicht* die AGE-Funktion. Benennen Sie die berechnete Spalte 'alter'.

```
SELECT personid, vorname, name, geburtsdatum, (current_date – geburtsdatum)  
/ 365 AS alter FROM person;
```

ExerciseID: 194193 (category: Arithmetische Ausdr, level: einfach)

Geben Sie alle Attribute der Tabelle Spielstandabschnitt zusammen mit der Summe der Tore eines jeden Spielabschnittes aus. Nennen Sie die berechnete Spalte 'torsumme'.

```
SELECT *, toret1 + toret2 AS torsumme FROM spielstandabschnitt;
```

ExerciseID: 194192 (category: Arithmetische Ausdr, level: einfach)

Geben Sie die Spalten Spielminute und Nachspielzeit der Tabelle Spielminute aus zusammen mit einer Spalte, die die Summe beider Werte angibt (beachten Sie, dass die Nachspielzeit NULL sein kann). Benennen Sie die berechnete Spalte 'effektive_spielminute'.

```
SELECT spielminute, nachspielzeit, spielminute + COALESCE(nachspielzeit,0)  
AS effektive_spielminute FROM spielminute;
```

DDL

ExerciseID: 194129 (category: DDL, level: einfach)

Löschen Sie die Tabelle 'person'.

```
DROP TABLE person;
```

ExerciseID: 194125 (category: DDL, level: einfach)

Erstellen Sie eine Tabelle 'verein', die die Attribute 'personid' (als Integer-Zahl), 'vereinname' (als Zeichenkette mit einer Länge von bis zu 30 Zeichen) sowie 'gage' (als Dezimalzahl mit 7 Vor- und 2 Nachkommastellen) enthält. (Bitte schlagen Sie die genaue Bedeutung der Parameter in Ihren Datentyp-Definitionen nach, bevor Sie einen BugReport senden!)

```
CREATE TABLE verein (personid INTEGER, vereinname VARCHAR(30), gage  
DECIMAL(9,2))
```

ExerciseID: 194131 (category: DDL, level: einfach)

Löschen Sie die Tabelle 'spielerwechsel'.

```
DROP TABLE spielerwechsel;
```

ExerciseID: 194130 (category: DDL, level: einfach)

Erzeugen Sie eine Tabelle 'foul' mit den Attributen 'aktionid', 'vonspielerid' und 'anspielerid'. Alle Attribute sollen vom Typ Integer und Nicht-Null sein. Attribut 'aktionid' soll darüber hinaus Primärschlüssel sein.

```
CREATE TABLE foul (aktionid INTEGER NOT NULL PRIMARY KEY, vonspielerid  
INTEGER NOT NULL, anspielerid INTEGER NOT NULL);
```

ExerciseID: 194126 (category: DDL, level: einfach)

Löschen Sie die Tabelle 'turnierausrichtung'.

```
DROP TABLE turnierausrichtung;
```

ExerciseID: 194128 (category: DDL, level: einfach)

Löschen Sie die Tabelle 'spiel'.

```
DROP TABLE spiel;
```

ExerciseID: 194124 (category: DDL, level: einfach)

Erzeugen Sie eine Tabelle 'stadion' mit den Attributen 'stadionid' (vom Typ integer), 'kapazitaet' (von Typ integer), 'stadt' (vom Typ varchar(30)) und 'landid' (vom Typ integer).

```
CREATE TABLE stadion (stadionid INTEGER, kapazitaet INTEGER, stadt
VARCHAR(30), landid INTEGER);
```

ExerciselD: 194127 (category: DDL, level: einfach)

Erstellen Sie eine Tabelle 'personattr', die (vorerst nur) die Attribute 'personid' (als Integer-Zahl), 'nachname' und 'vorname' (jeweils als Zeichenkette mit 31 Zeichen) sowie geschlecht (als char) enthält.

```
CREATE TABLE personattr (personid INTEGER, nachname VARCHAR(31), vorname
VARCHAR(31), geschlecht CHAR);
```

DML

ExerciselD: 194181 (category: DML, level: einfach)

Das in der Datenbank gespeicherte Geburtsdatum von Raul Cordoba ist nicht korrekt. Setzen Sie es auf den 14. April 1924.

```
UPDATE person SET geburtsdatum = DATE '14-04-1924' WHERE name = 'Cordoba'
AND vorname = 'Raul';
```

ExerciselD: 194183 (category: DML, level: einfach)

Ergänzen Sie die Tabelle 'tortyp' um ein Tor namens 'Wembley-Tor' (tortypid=1007).

```
INSERT INTO tortyp (bezeichnung, tortypid)VALUES('Wembley-Tor', 1007);
```

ExerciselD: 194177 (category: DML, level: einfach)

WM 1986. Mexico. Achtelfinale! Marokko-Deutschland. 87 Spielminute. Immer noch 0:0! Foul an Rummenigge. Torentfernung: 30m. Lothar Matthäus schießt. Tor! 0:1! Lothar Matthäus schießt Deutschland ins Viertelfinale... Tragen Sie diesen Treffer in die Tabelle 'tor' ein (aktionid=1007, spielerid=9878, tortypid=12).

```
INSERT INTO tor VALUES (1007, 9878, 12,0,1);
```

ExerciselD: 194180 (category: DML, level: einfach)

Ergänzen Sie die Tabelle 'strafetyp' um eine Strafe namens '20 Liegestützen' (strafetypid=1007).

```
INSERT INTO strafetyp VALUES (1007, '20 Liegestützen');
```

ExerciselD: 194179 (category: DML, level: einfach)

Günther Jauch wird für die nächste Weltmeisterschaft nominiert (er wurde am 13. Juli 1956 in Münster geboren). Tragen Sie ihn in die Tabelle 'person' ein (personid=1000000).

```
INSERT INTO person (personid, name, vorname, geburtsdatum) VALUES (1000000, 'Jauch', 'Günther', DATE('13-07-1956'));
```

ExerciselD: 194178 (category: DML, level: einfach)

Nepal hat sich für die Fußball-Weltmeisterschaft qualifiziert. Tragen Sie das Land in die Tabelle 'land' ein (landid=1007, kontinentid=17).

```
INSERT INTO land VALUES (1007, 'Nepal', 17);
```

ExerciselD: 194184 (category: DML, level: einfach)

Alle Spiele, die vor dem 15.06.1970 stattgefunden haben, sollen gelöscht werden.

```
DELETE FROM spiel WHERE anpfiff < DATE '1970-06-15';
```

ExerciselD: 194182 (category: DML, level: einfach)

Alle Personen, die vor dem 1.1.1950 geboren wurden, sollen aus der Tabelle 'person' gelöscht werden.

```
DELETE FROM person WHERE geburtsdatum < DATE '1950-01-01';
```

ExerciselD: 194185 (category: DML, level: mittel)

Erhöhen Sie die Rückennummern aller Spieler, die bei einer beliebigen WM ein Tor geschossen haben, um 10.

```
UPDATE personnominierung SET rueckennummer = rueckennummer + 10 WHERE personid IN (SELECT spielerid FROM tor);
```

ExerciselD: 194186 (category: DML, level: schwer)

In einem Vorrunden-Spiel der WM 2006 gegen Australien bekam der Kroatie Josip Simunic in der 90. Spielminute irrtümlich eine zweite Gelbe Karte (ehe er bei einem dritten Vergehen mit Gelb-Rot vom Platz gestellt wurde). Ändern Sie die zweite Gelbe Karte in eine Gelb-Rote Karte. Verwenden Sie SELECT-Anweisung um die entsprechenden IDs (strafetypid und aktionid) festzustellen.

```
UPDATE strafe SET strafetypid =  
    (SELECT strafetypid FROM strafetyp  
     WHERE bezeichnung = 'Gelb-Rote Karte')  
WHERE strafetypid IN  
    (SELECT strafetypid FROM strafetyp WHERE bezeichnung = 'Gelbe Karte')  
AND aktionid IN  
    (SELECT aktionid FROM aktion WHERE spielid =  
     (SELECT spielid FROM spiel WHERE spielid = 760) AND  
     spielminuteid IN  
        (SELECT spielminuteid FROM spielminute WHERE spielminute  
         = 90))  
AND spielerid IN (  
    SELECT personid FROM person WHERE name = 'Simunic' AND vorname =  
    'Josip');
```

Gruppierungen

ExerciselD: 194138 (category: Gruppierungen, level: einfach)

Geben Sie die Häufigkeit der Spielstände (zum Beispiel(!) 0:0, 0:1, 1:0, 3:1 etc.) an, die am Ende der ersten Halbzeit eines beliebigen Spiels einer beliebigen Weltmeisterschaft vorlagen. Die Ausgabe soll (toret1, toret2, count) lauten.

```
SELECT toret1, toret2, COUNT(*) AS count FROM spiel
NATURAL JOIN spielstandabschnitt
NATURAL JOIN spielabschnitt
WHERE bezeichnung = 'Erste Halbzeit'
GROUP BY toret1, toret2;
```

ExerciselD: 194136 (category: Gruppierungen, level: einfach)

Geben Sie PersonId, Nachname und Vorname einer jeden Person zusammen mit der Häufigkeit aus, mit der sie an einer Weltmeisterschaft (egal in welcher Rolle) teilgenommen hat. Das Ergebnis soll (personid, name, vorname, anzahl) lauten.

```
SELECT personid, name, vorname, COUNT(nominierungid) FROM person
NATURAL LEFT JOIN personnominierung
GROUP BY personid, name, vorname;
```

ExerciselD: 194139 (category: Gruppierungen, level: einfach)

Geben Sie die Häufigkeit der Vornamen aller Personen aus, die jemals an einer Weltmeisterschaft teilgenommen haben. Das Ergebnis soll aus den Spalten (vorname, anzahl) bestehen.

```
SELECT vorname, COUNT(*) FROM person
GROUP BY vorname;
```

ExerciselD: 194135 (category: Gruppierungen, level: einfach)

Geben Sie die SpielerId aller Spieler aus, die je bei einer Weltmeisterschaft ein Tor geschossen haben, zusammen mit der Gesamtanzahl der Tore, die sie bei Weltmeisterschaften insgesamt geschossen haben.

```
SELECT spielerid, COUNT(*) FROM person
JOIN tor ON tor.spielerid = person.personid
GROUP BY spielerid;
```

ExerciselD: 194132 (category: Gruppierungen, level: einfach)

Geben Sie die SpielerId aller Spieler aus, die je bei einer Weltmeisterschaft eine (Gelbe, Gelb-Rote oder Rote) Karte gesehen haben, zusammen mit der Gesamtanzahl der Karten, die sie bei Weltmeisterschaften insgesamt erhalten haben.

```
SELECT DISTINCT spielerid, COUNT(spielerid) AS count FROM strafe
GROUP BY spielerid;
```

ExerciselD: 194137 (category: Gruppierungen, level: einfach)

Geben Sie die Häufigkeit aller Rückennummern an, die jemals zu einer Weltmeisterschaft nominiert wurden. Das Ergebnis soll (rueckennummer, anzahl) lauten. Schließen Sie NULL-Werte aus Ihrer Anfrage aus.

```
SELECT rueckennummer, COUNT(*) AS anzahl
FROM personnominierung
WHERE rueckennummer NOTNULL
GROUP BY rueckennummer;
```

ExerciselD: 194133 (category: Gruppierungen, level: einfach)

Geben Sie die Häufigkeit der Geburtsdaten der Personen aus, die jemals an einer Weltmeisterschaft teilgenommen haben. Das Ergebnis soll aus (geburtsdatum, anzahl) bestehen. Schließen Sie NULL-Werte aus Ihrer Anfrage aus.

```
SELECT geburtsdatum, COUNT(*) FROM person
WHERE geburtsdatum NOTNULL
GROUP BY geburtsdatum;
```

ExerciselD: 194134 (category: Gruppierungen, level: einfach)

Geben Sie für jedes Land, in dem bisher eine Weltmeisterschaft stattgefunden hat, die Anzahl der Weltmeisterschaften an, die das Land bisher insgesamt ausgerichtet hat. Die Ausgabe soll (landname, count) lauten.

```
SELECT landname, COUNT(*) FROM turnierausrichtung
NATURAL JOIN land
GROUP BY landname;
```

ExerciselD: 194141 (category: Gruppierungen, level: mittel)

Geben Sie für jeden *Spieler* (!) an, wie oft er an einer Weltmeisterschaft teilgenommen hat. Das Ergebnis soll (personid, vorname, nachname, anzahl_nominierung) lauten.

```
SELECT personid, vorname, name, COUNT(*) AS anzahl_nominierung FROM person
NATURAL JOIN personnominierung
NATURAL JOIN personrolle
WHERE bezeichnung IN ('TW', 'VT', 'MI', 'ST')
GROUP BY personid, vorname, name;
```


ExercisID: 194142 (category: Gruppierungen, level: mittel)

Geben Sie SpielerId, Vornamen und Nachnamen aller Personen aus, die je bei einer Weltmeisterschaft eine Gelbe, Gelb-Rote oder Rote Karte gesehen haben. Geben Sie zudem - getrennt nach Kartenart- die Gesamtanzahl der Gelben, Gelb-Roten bzw. Roten Karten an, die die Personen bei Weltmeisterschaften insgesamt gesehen haben. Die Ausgabe soll (spielerid, vorname, name, bezeichnung, count) lauten.

```
SELECT spielerid, vorname, name, bezeichnung, COUNT(*) AS count FROM strafe
NATURAL JOIN strafetyp
JOIN person ON strafe.spielerid = person.personid
GROUP BY spielerid, vorname, name, bezeichnung;
```

ExercisID: 194145 (category: Gruppierungen, level: mittel)

Geben Sie PersonId, Vorname und Nachname aller Personen aus, die an mehr als drei Weltmeisterschaften teilgenommen haben.

```
SELECT personid, vorname, name FROM person
NATURAL JOIN personnominierung
GROUP BY personid, vorname, name
HAVING COUNT(*) > 3;
```

ExercisID: 194146 (category: Gruppierungen, level: mittel)

Erstellen Sie eine Liste aus Alter (in Jahren) und Anzahl der Personen, die dieses Alter haben. Nehmen Sie an, dass alle Personen noch leben. Verwenden Sie die AGE-Funktion. Das Ergebnis soll (alter, anzahl) lauten. Lassen Sie bei Ihren Berechnungen NULL-Werte außer Acht.

```
SELECT EXTRACT(YEAR FROM AGE(geburtsdatum)) AS alter, COUNT(*) FROM person
WHERE geburtsdatum NOTNULL
GROUP BY alter;
```

ExercisID: 194143 (category: Gruppierungen, level: mittel)

Geben Sie die Häufigkeit der Spielstände (z.B. 1:0, 3:1, etc.) an, die am Ende der zweiten Halbzeit eines beliebigen Spiels einer beliebigen Weltmeisterschaft vorlagen und bei denen die Heimmannschaft mit mindestens einem Tor Vorsprung führte. Die Ausgabe soll (toret1, toret2, count) lauten.

```
SELECT toret1, toret2, COUNT(*) FROM spielstandabschnitt
NATURAL JOIN spielabschnitt
WHERE toret1 - toret2 >= 1 AND bezeichnung = 'Zweite Halbzeit'
GROUP BY toret1, toret2;
```

ExerciseID: 194147 (category: Gruppierungen, level: mittel)

Geben Sie für jeden Spieler die Häufigkeit an, mit der er während eines beliebigen Spiels bei einer beliebigen Weltmeisterschaft eine Gelbe Karte erhalten hat (beschränken Sie Ihre Ausgabe *der Einfachheit halber* auf jene Spieler, die bisher -bei einer beliebigen Weltmeisterschaft- mindestens einmal eine Gelbe Karte erhalten haben). Das Ergebnis soll (spielerid, anzahl_gelb) lauten.

```
SELECT spielerid, COUNT(*) FROM person
JOIN strafe ON spielerid = personid
NATURAL JOIN strafetyp
WHERE bezeichnung = 'Gelbe Karte'
GROUP BY spielerid;
```

ExerciseID: 194144 (category: Gruppierungen, level: mittel)

Geben Sie für jedes Spiel und für jede daran beteiligte Mannschaft die Anzahl der von den Mannschaften nominierten Spieler aus. Das Ergebnis soll (spielid, teamid, count) lauten.

ExerciseID: 194140 (category: Gruppierungen, level: mittel)

Geben Sie alle Tore aus, die im selben Spiel in der selben Spielminute erzielt worden sind. Lassen Sie dabei Nachspielminuten sowie Tore, deren Spielminute nicht erfasst wurde (d.h. spielminute=0), außer Acht. Die Ausgabe soll (spielid, spielminute) lauten.

ExerciseID: 194148 (category: Gruppierungen, level: schwer)

Geben Sie PersonID, Vorname und Nachname aller Personen an, die mehr Tore geschossen haben als Sandor Kocsis - durchschnittlich gesehen über alle seine WM-Spiele, bei denen er nominiert wurde.

```
SELECT personid, vorname, name FROM person
INNER JOIN tor ON person.personid = tor.spielerid
GROUP BY personid, vorname, name
HAVING COUNT(*) >
    (SELECT AVG(t) FROM
        (SELECT COUNT(tore) AS t FROM person
        LEFT JOIN spielnominierung ON spielnominierung .spielerid =
        person.personid
        NATURAL JOIN spiel
        NATURAL JOIN aktion
        NATURAL LEFT JOIN
            (SELECT * FROM tor
            INNER JOIN person ON tor.spielerid = person.personid
            WHERE UPPER(vorname) = 'SANDOR' AND
            UPPER(name) = 'KOCISIS') as tore
        WHERE UPPER(vorname) = 'SANDOR' AND UPPER(name) = 'KOCISIS'
        GROUP BY spielid)
    AS b);
```

Komplexe Anfragen

ExerciseID: 194189 (category: Komplexe Anfragen, level: einfach)

Geben Sie den Namen des Landes an, dessen Team die meisten WM-Spiele bestritten hat. Geben Sie zudem die Anzahl der Spiele an, die das Land bestritten hat. Die Ausgabe soll (landid, landname, count) lauten.

```
SELECT landid, landname, count FROM (
    SELECT landid, landname, COUNT(*) AS count FROM land
    NATURAL JOIN teamland
    NATURAL JOIN team
    INNER JOIN spiel AS s1 ON s1.t1id = team.teamid OR s1.t2id =
    team.teamid
    GROUP BY landid, landname) AS t1
WHERE count =
    (SELECT MAX(count) FROM (SELECT landid, landname, COUNT(*) AS count
    FROM land
    NATURAL JOIN teamland
    NATURAL JOIN team
    INNER JOIN spiel AS s1 ON s1.t1id = team.teamid OR s1.t2id =
    team.teamid
    GROUP BY landid, landname) AS t2);
```

ExerciseID: 194188 (category: Komplexe Anfragen, level: einfach)

Geben Sie den Namen des Landes an, dessen Team am häufigsten an einer WM teilgenommen hat. Geben Sie zudem die Anzahl der Turniere an, an denen das Land teilgenommen hat. Die Ausgabe soll (landid, landname, count) lauten.

```
SELECT landid, landname, count FROM
    (SELECT landid, landname, COUNT(*) AS count
    FROM land NATURAL JOIN teamland NATURAL JOIN teamnominierung
    NATURAL JOIN nominierung NATURAL JOIN turnier
    GROUP BY landid, landname) AS t1
WHERE count = (SELECT MAX(count) FROM
    (SELECT landid, landname, COUNT(*) AS count
    FROM land NATURAL JOIN teamland NATURAL JOIN teamnominierung
    NATURAL JOIN nominierung NATURAL JOIN turnier
    GROUP BY landid, landname) AS t2);
```

ExercisID: 194216 (category: Komplexe Anfragen, level: mittel)

Geben Sie PersonID, Nachname und Vorname aller Spieler an, die bei der Weltmeisterschaft 1998 mehr Tore geschossen haben als die durchschnittliche Anzahl von Toren (pro Spiel und mit Nachkommastellen!), die in allen Spielen der zweiten Runde aller bisheriger Weltmeisterschaften geschossen wurden.

```
SELECT personid, name, vorname FROM person
INNER JOIN tor ON personid = spielerid
NATURAL JOIN aktion
NATURAL JOIN spiel
NATURAL JOIN turnier
WHERE jahr = 1998
GROUP BY personid, name, vorname
HAVING COUNT(*) > (
    SELECT AVG(tore) FROM (SELECT spielid, MAX(toret1) + MAX(toret2) AS
        tore
        FROM spiel NATURAL JOIN spielstandabschnitt
        WHERE phase = 'Zweite Runde'
        GROUP BY spielid) AS t);
```

ExercisID: 194190 (category: Komplexe Anfragen, level: schwer)

Geben Sie die Namen aller Teams aus, die während einer(!) Weltmeisterschaft mehr als einmal gegeneinander gespielt haben. Geben Sie außerdem Namen des Turniers und Phase an, innerhalb derer die Teams aufeinander getroffen sind, sowie das Datum und die Gruppe des Spiels. Die Ausgabe soll (turniername, phase, gruppe, tag, heimmannschaft, gastmannschaft) lauten. (Bitte beachten Sie, dass es bei früheren Weltmeisterschaften bei unentschiedenem Endspielstand Wiederholungsspiele gab, die mit auszugeben sind!)

```
SELECT turniername, phase, gruppe, anpfiff AS tag, t1.name AS
heimmannschaft, t2.name AS gastmannschaft
FROM spiel
INNER JOIN team AS t1 ON spiel.t1id = t1.teamid
INNER JOIN team AS t2 ON spiel.t2id = t2.teamid
NATURAL JOIN turnier
INNER JOIN
    (SELECT turnierid,
        CASE WHEN t1id < t2id THEN t1id ELSE t2id END AS t1,
        CASE WHEN t1id < t2id THEN t2id ELSE t1id END AS t2
    FROM spiel
    GROUP BY turnierid, t1, t2
    HAVING COUNT(spielid) > 1) AS multi
ON (spiel.turnierid = multi.turnierid AND
    ((spiel.t1id = multi.t1 AND spiel.t2id = multi.t2) OR
    (spiel.t1id = multi.t2 AND spiel.t2id = multi.t1)))
ORDER BY spielid;
```

ExercisID: 194187 (category: Komplexe Anfragen, level: einfach)

Bei welcher Weltmeisterschaft wurden die meisten Tore geschossen? Das Ergebnis soll (turniername, jahr, tore) lauten.

Mengen

ExerciselD: 194168 (category: Mengen, level: einfach)

Geben Sie (ohne Duplikate) SpielerId, Vorname und Nachname aller Spieler aus, die in einer(!) jener Spielminuten ein Tor erzielten, in denen auch der Allzeit-Torschützenkönig Just Fontaine ein Tor erzielte. Geben Sie zudem die Spielminute aus, in der der Torschuss erfolgt ist.

```
SELECT DISTINCT spielerid, vorname, name, spielminute FROM person
JOIN tor ON person.personid = tor.spielerid
NATURAL JOIN aktion
NATURAL JOIN spielminute
WHERE spielminute IN (
    SELECT DISTINCT spielminute FROM person
    JOIN tor ON person.personid = tor.spielerid
    NATURAL JOIN aktion
    NATURAL JOIN spielminute
    WHERE name = 'Fontaine' AND vorname = 'Just');
```

ExerciselD: 194172 (category: Mengen, level: einfach)

Zeigen Sie (ohne Duplikate) Personid, Nachname und Vorname aller *Spieler* (!) an, die zwar an einer Weltmeisterschaft teilgenommen haben, aber nie in einem Weltmeisterschaftsspiel tatsächlich gespielt haben (d.h. die nie in der Anfangsaufstellung waren und auch nie im Verlaufe eines Spiels eingewechselt wurden).

```
SELECT DISTINCT personid, name, vorname FROM person
NATURAL JOIN personnominierung
NATURAL JOIN personrolle
WHERE bezeichnung IN('TW','MI','VT','ST')
AND personid NOT IN
    (SELECT spielerid FROM spielnominierung
    WHERE instartelf
    UNION
    SELECT reinspielerid FROM spielerwechsel);
```

ExerciselD: 194167 (category: Mengen, level: einfach)

Geben Sie (ohne Duplikate) die PersonId aller *Spieler* (!) aus, die noch nie ein Tor geschossen haben.

```
SELECT DISTINCT personid FROM person
NATURAL JOIN personnominierung
NATURAL JOIN personrolle
WHERE bezeichnung IN('TW','MI','VT','ST')
EXCEPT
    SELECT personid FROM person
    JOIN tor ON tor.spielerid=person.personid;
```

ExerciselD: 194173 (category: Mengen, level: einfach)

Geben Sie die PersonId aller Spieler aus, deren *letzte*(!) Nominierung zu einem Weltmeisterschafts*spiel* mehr als 10 Jahre zurückliegt. (Musterlösung aktualisiert am 06.06.2011, 17h04!)

ExerciseID: 194175 (category: Mengen, level: schwer)

Welche Spieler haben sowohl in dem Spiel mit der SpielId 787 als auch in dem Spiel mit der SpielId 837 *gespielt* (!) (d.h. waren gleich in der Anfangsaufstellung dabei oder wurden im Verlauf des Spiels eingewechselt)? Das Ergebnis soll (personid, name, vorname) laute

ExerciseID: 194169 (category: Mengen, level: einfach)

Welche Spieler wurden für das Spiel mit der SpielId 1034, aber nicht für das Spiel mit der SpielId 1129 nominiert? Das Ergebnis soll (personid, name, vorname) lauten.

```
SELECT personid, name, vorname FROM person
LEFT JOIN spielnominierung ON spielnominierung.spielerid = person.personid
WHERE spielid = 1034
EXCEPT
SELECT personid, name, vorname FROM person
LEFT JOIN spielnominierung ON spielnominierung.spielerid = person.personid
WHERE spielid = 1129;
```

ExerciseID: 194166 (category: Mengen, level: einfach)

Geben Sie (ohne Duplikate) PersonId, Vorname und Nachname aller Spieler an, die nur an Spielen der zweiten Runde teilgenommen und dort mindestens ein Tor geschossen haben.

```
SELECT DISTINCT personid, vorname, name FROM person
INNER JOIN tor ON person.personid = tor.spielerid
NATURAL JOIN aktion
NATURAL JOIN spiel
WHERE phase = 'Zweite Runde'
EXCEPT
SELECT DISTINCT personid, vorname, name FROM person
LEFT JOIN spielnominierung ON person.personid = spielnominierung.spielerid
NATURAL JOIN spiel
WHERE phase = 'Gruppenspiele';
```

ExerciseID: 194171 (category: Mengen, level: einfach)

Geben Sie (ohne Duplikate) Personid, Nachname und Vorname aller Spieler aus, die nicht nur an einer Weltmeisterschaft teilgenommen haben, sondern auch in einem Weltmeisterschaftsspiel tatsächlich gespielt haben.

```
SELECT personid, name, vorname FROM person
INNER JOIN spielnominierung ON person.personid = spielnominierung.spielerid
WHERE instartelf = true
UNION (SELECT personid, name, vorname FROM person
INNER JOIN spielerwechsel ON person.personid =
spielerwechsel.reinspielerid);
```

ExerciseID: 194170 (category: Mengen, level: einfach)

Geben Sie (ohne Duplikate) *SpielerId*(!), Nachname und Vorname aller Spieler aus, deren -bei irgendeiner Weltmeisterschaft- erzielten Tore ausnahmslos Eigentore waren. (Beachten Sie, dass "normale" Tore keinen Tortyp haben, d.h. dass das Attribut 'tortypid' in der Tabelle 'tor' 0 bzw. NULL ist.)

```

SELECT spielerid, name, vorname FROM person
JOIN tor ON tor.spielerid = person.personid
EXCEPT
SELECT spielerid, name, vorname FROM person
JOIN tor ON tor.spielerid = person.personid
WHERE COALESCE(tortypid,0 ) != 11;

```

ExerciseID: 194174 (category: Mengen, level: schwer)

Geben Sie Namen und Vornamen aller Trainer aus, die auch als Spieler an einer Weltmeisterschaft teilgenommen haben. Geben Sie zudem die Namen der Turniere aus, an denen die betreffenden Personen als Trainer mitgewirkt haben. Das Ergebnis soll (name, vorname, turniername) sein. (TIP: Joinen Sie über Vor- und Nachnamen in Großbuchstaben und geben Sie Vor- und Nachnamen in Großbuchstaben aus!)

```

SELECT t2.name, t2.vorname, t2.turniername FROM (
SELECT UPPER(name) AS name , UPPER(vorname) as vorname FROM person
NATURAL JOIN personnominierung
NATURAL JOIN personrolle
WHERE bezeichnung = 'Trainer'
INTERSECT(
    SELECT UPPER(name) AS name , UPPER(vorname) as vorname FROM person
    NATURAL JOIN personnominierung
    NATURAL JOIN personrolle
    WHERE bezeichnung != 'Trainer')) AS t1
INNER JOIN
    (SELECT UPPER(name) AS name , UPPER(vorname) as vorname,
    turniername FROM person
    NATURAL JOIN personnominierung
    NATURAL JOIN personrolle
    NATURAL JOIN nominierung
    NATURAL JOIN turnier
    WHERE bezeichnung = 'Trainer')
    AS t2
ON t1.name = t2.name AND t1.vorname = t2.vorname;

```

ExerciseID: 194176 (category: Mengen, level: schwer)

Geben Sie Namen und Vornamen aller Spieler aus, die auch als Trainer an einer Weltmeisterschaft teilgenommen haben. Geben Sie zudem die Namen der Turniere aus, an denen die betreffenden Personen als Spieler teilgenommen haben. Das Ergebnis soll (name, vorname, turniername) sein. (TIP: Joinen Sie über Vor- und Nachnamen in Großbuchstaben und geben Sie Vor- und Nachnamen in Großbuchstaben aus!)

```

SELECT t2.name, t2.vorname, t2.turniername FROM (
SELECT UPPER(name) AS name, UPPER(vorname) AS vorname FROM person
NATURAL JOIN personnominierung
NATURAL JOIN personrolle
WHERE bezeichnung LIKE 'Trainer'
INTERSECT
    SELECT UPPER(name) AS name, UPPER(vorname) AS vorname FROM person
    NATURAL JOIN personnominierung
    NATURAL JOIN personrolle
    WHERE bezeichnung NOT LIKE 'Trainer') AS t1
INNER JOIN
    (SELECT UPPER(name) AS name, UPPER(vorname) AS vorname,
    turniername FROM person

```

```

        NATURAL JOIN personnominierung
        NATURAL JOIN personrolle
        NATURAL JOIN nominierung
        NATURAL JOIN turnier
        WHERE bezeichnung NOT LIKE 'Trainer')
    AS t2
ON t1.name = t2.name AND t1.vorname = t2.vorname;

```

SFW

ExerciseID: 194202 (category: SFW, level: einfach)

Geben Sie alle Tage aus (ohne Duplikate), an denen eine Person geboren worden ist. Lassen Sie dabei NULL-Marken außer Acht.

```
SELECT DISTINCT geburtsdatum FROM person WHERE geburtsdatum NOTNULL;
```

ExerciseID: 194195 (category: SFW, level: einfach)

Geben Sie Name, Vorname und Geburtsdatum aller Personen aus, deren Nachname 'Ivanov' lautet

```
SELECT name, vorname, geburtsdatum FROM person WHERE name = 'Ivanov';
```

ExerciseID: 194196 (category: SFW, level: einfach)

Geben Sie Name, Vorname und Geburtsdatum aller Personen aus, die mit Vornamen 'Jozsef' und mit Nachnamen 'Toth' heißen.

```
SELECT name, vorname, geburtsdatum FROM person
WHERE vorname = 'Jozsef' AND name = 'Toth';
```

ExerciseID: 194200 (category: SFW, level: einfach)

Geben Sie Vorname, Nachname und Geburtsdatum aller Personen aus, deren Vorname weder 'Paul', noch 'Paulo', noch 'Paolo' lautet.

```
SELECT vorname, name, geburtsdatum FROM person
WHERE vorname NOT IN ('Paul', 'Paulo', 'Paolo');
```

ExerciseID: 194199 (category: SFW, level: einfach)

Geben Sie Vorname, Nachname und Geburtsdatum aller Personen aus, deren Vorname weder 'John', noch 'Juan', noch 'Jean' und noch 'Johannes' lautet.

```
SELECT vorname, name, geburtsdatum FROM person
WHERE vorname NOT IN ('John', 'Juan', 'Jean', 'Johannes');
```

ExerciseID: 194198 (category: SFW, level: einfach)

Geben Sie alle Attribute der Personen aus, deren Vorname leer (d.h. '') ist.

```
SELECT * FROM person WHERE vorname = '';
```

ExerciseID: 194204 (category: SFW, level: einfach)

Geben Sie die PersonId aller Spieler aus, die (bei einer beliebigen WM) die Rückennummer 1 getragen haben.

```
SELECT personid FROM personnominierung WHERE rueckennummer = 1;
```

ExerciseID: 194203 (category: SFW, level: einfach)

Geben Sie alle Turniere (mit allen ihren Attributen) aus.

```
SELECT * FROM turnier;
```

ExerciseID: 194197 (category: SFW, level: einfach)

Geben Sie alle Nachnamen der Personen aus (ohne Duplikate).

```
SELECT DISTINCT name FROM person;
```

ExerciseID: 194194 (category: SFW, level: einfach)

Geben Sie die Spielabschnitte (mit allen ihren Attributen) aus, die in der Weltmeisterschaftsdatenbank gespeichert sind.

```
SELECT * FROM spielabschnitt;
```

ExerciseID: 194201 (category: SFW, level: einfach)

Geben Sie Name, Vorname und Geburtsdatum aller Personen aus, deren Nachname 'Cho' lautet.

```
SELECT name, vorname, geburtsdatum FROM person WHERE name = 'Cho';
```

ExerciseID: 194209 (category: SFW, level: mittel)

Geben Sie Name, Vorname und Geburtsdatum aller Personen aus, deren Nachname die Buchstabenkombination 'aou' beinhaltet

```
SELECT name, vorname, geburtsdatum FROM person WHERE name LIKE '%aou%';
```

ExerciseID: 194211 (category: SFW, level: mittel)

Geben Sie Vorname und Nachname aller Personen aus, die im Februar 1967 Geburtstag haben?

```
SELECT vorname, name FROM person
WHERE geburtsdatum BETWEEN '1-2-1967' AND '28-2-1967';
```

ExerciselD: 194212 (category: SFW, level: mittel)

Geben Sie Vornamen und Nachnamen aller Personen aus, deren Vorname die Buchstabenkombination 'no' beinhaltet und deren Nachname die Buchstabenkombination 'si' beinhaltet.

```
SELECT vorname, name FROM person
WHERE vorname LIKE '%no%' AND name LIKE '%si%';
```

ExerciselD: 194214 (category: SFW, level: mittel)

Geben Sie Vornamen und Nachnamen aller Personen aus, die vor dem 01.01.1979 geboren wurden?

```
SELECT vorname, name FROM person WHERE geburtsdatum < '1-1-1979';
```

ExerciselD: 194215 (category: SFW, level: mittel)

Geben Sie alle Attribute aller Gruppenspiele aus, die an einem Montag stattgefunden haben.

```
SELECT * FROM spiel
WHERE phase = 'Gruppenspiele' AND EXTRACT( DOW FROM anpfiff) = 1;
```

ExerciselD: 194213 (category: SFW, level: mittel)

Geben Sie (ohne Duplikate) den Namen aller Länder aus, die auf 'reich' enden.

```
SELECT DISTINCT landname FROM land WHERE landname LIKE '%reich';
```

ExerciselD: 194206 (category: SFW, level: mittel)

Geben Sie Vorname, Nachname und Geburtsdatum aller Personen aus, die nach dem 01.01.1988 geboren wurden.

```
SELECT vorname, name, geburtsdatum FROM person
WHERE geburtsdatum > DATE('01-01-1988');
```

ExerciselD: 194207 (category: SFW, level: mittel)

Geben Sie alle Attribute der Spiele aller Weltmeisterschaften aus, die nicht länger als 30 Jahre zurückliegen (benutzen Sie die AGE Funktion)

```
SELECT * FROM spiel WHERE EXTRACT(YEAR FROM AGE(anpfiff))< 30;
```

ExerciselD: 194205 (category: SFW, level: mittel)

Geben Sie (ohne Duplikate) den Namen aller Länder aus, die mit 'K' beginnen.

```
SELECT DISTINCT landname FROM land WHERE landname LIKE 'K%';
```

ExerciselD: 194210 (category: SFW, level: mittel)

Geben Sie alle Attribute aller Personen aus, deren Nachnamen nur aus Großbuchstaben besteht.

```
SELECT * FROM person WHERE name = UPPER(name);
```

ExerciselD: 194208 (category: SFW, level: mittel)

Geben Sie Vorname, Nachname und Geburtsdatum aller Personen aus, die in den Jahren 1960 bis 1982 geboren wurden.

```
SELECT vorname, name, geburtsdatum FROM person  
WHERE geburtsdatum BETWEEN DATE('1960-01-01') AND DATE('1982-12-31');
```

ExerciselD: 194113 (category: SFW, level: schwer)

Geben Sie (ohne Duplikate) Vornamen und Nachnamen aller Personen aus, deren Nachname aus höchstens 5 Buchstaben besteht und deren Vornamen ein 'r' beinhaltet und mit einem 'P' beginnt.

```
SELECT DISTINCT vorname, name FROM person  
WHERE LENGTH(name) <= 5 AND vorname LIKE 'P%r%';
```

Verbund

ExerciselD: 194150 (category: Verbund, level: einfach)

Geben Sie Vorname, Nachname und Landname aller Spieler aus, die bei einer Weltmeisterschaft je die Rückennummer 11 getragen haben. Geben Sie zudem den Turniernamen der jeweiligen Weltmeisterschaft aus.

```
SELECT vorname, name, landname, turniername FROM person  
NATURAL JOIN personnominierung  
NATURAL JOIN nominierung  
NATURAL JOIN turnier  
JOIN teamnominierung ON personnominierung.teamnominierungid =  
teamnominierung.nominierungid  
NATURAL JOIN teamland  
NATURAL JOIN land  
WHERE rueckennummer = 11;
```

ExerciseID: 194149 (category: Verbund, level: einfach)

Geben Sie den Turniernamen aller Weltmeisterschaften aus, an denen Henryk Kaspercak teilgenommen hat. Geben Sie zudem die Rollenbezeichnung aus sowie den Namen des Landes, für das Henryk Kaspercak an der Weltmeisterschaft teilgenommen hat. (Tip: Verwenden Sie die upper(...) -Funktion, um Vor- und Nachnamen in Großbuchstaben umzuwandeln, und vergleichen Sie mit HENRYK KASPERCZAK.)

```
SELECT turniername, bezeichnung, landname FROM turnier
NATURAL JOIN nominierung
NATURAL JOIN personnominierung
NATURAL JOIN person
NATURAL JOIN personrolle
LEFT JOIN teamnominierung ON personnominierung.teamnominierungid =
teamnominierung.nominierungid
NATURAL JOIN teamland
NATURAL JOIN land
WHERE UPPER(vorname) = 'HENRYK' AND UPPER(name) = 'KASPERCZAK';
```

ExerciseID: 194155 (category: Verbund, level: einfach)

Geben Sie (ohne Duplikate) den Vornamen und Nachnamen aller Trainer aus, die je an einer Weltmeisterschaft teilgenommen haben.

```
SELECT DISTINCT vorname, name FROM person
NATURAL JOIN personnominierung
NATURAL JOIN personrolle
WHERE bezeichnung = 'Trainer';
```

ExerciseID: 194154 (category: Verbund, level: einfach)

Geben Sie (mit Duplikaten) Vorname und Nachname aller Personen aus, die je in einer Weltmeisterschaft eine Gelbe, Gelb-Rote oder Rote Karte gesehen haben. Geben Sie die Bezeichnung der Karte, die die Spieler gesehen haben, mit aus.

```
SELECT vorname, name, bezeichnung FROM strafe
JOIN person ON strafe.spielerid = person.personid
NATURAL JOIN strafetyp;
```

ExerciseID: 194152 (category: Verbund, level: einfach)

Geben Sie Vorname, Nachname und Land aller Personen aus, die an der Weltmeisterschaft des Jahres 2006 teilgenommen haben.

```
SELECT vorname, name, landname FROM person
NATURAL JOIN personnominierung
NATURAL JOIN nominierung
NATURAL JOIN turnier
JOIN teamnominierung ON teamnominierung.nominierungid =
personnominierung.teamnominierungid
NATURAL JOIN teamland
```

```
NATURAL JOIN land
WHERE turnier.jahr = 2006;
```

ExerciseID: 194151 (category: Verbund, level: einfach)

Geben Sie die Namen aller Länder aus (ohne Duplikate), in denen eine Weltmeisterschaft ausgerichtet wurde.

```
SELECT DISTINCT landname FROM turnierausrichtung
NATURAL JOIN turnier
NATURAL JOIN land;
```

ExerciseID: 194153 (category: Verbund, level: einfach)

Geben Sie Vorname und Nachname aller Spieler aus, die bei einer Weltmeisterschaft je die Rückennummer 13 getragen haben.

```
SELECT vorname, name FROM person
NATURAL JOIN personnominierung
WHERE rueckennummer = 13;
```

ExerciseID: 194156 (category: Verbund, level: einfach)

Geben Sie (ohne Duplikate) den Vornamen, Nachnamen und das Geburtsdatum aller Stürmer aus, die je an einer Weltmeisterschaft teilgenommen haben.

```
SELECT DISTINCT vorname, name, geburtsdatum FROM person
NATURAL JOIN personnominierung
NATURAL JOIN personrolle
WHERE bezeichnung = 'ST';
```

ExerciseID: 194158 (category: Verbund, level: mittel)

Geben Sie die PersonID aller Torwarte aus, die im Juli zwischen 1965 und 1985 geboren worden sind.

```
SELECT personid FROM person
NATURAL JOIN personnominierung
NATURAL JOIN personrolle
WHERE bezeichnung = 'TW' AND EXTRACT(MONTH FROM geburtsdatum) = 7 AND
EXTRACT (YEAR FROM geburtsdatum) BETWEEN 1965 AND 1985;
```

ExerciseID: 194159 (category: Verbund, level: mittel)

Geben Sie die PersonID aller *Spieler* (!) aus, die im Mai des Jahres 1977 bzw. 1982 geboren worden sind.

```
SELECT personid FROM personnominierung
NATURAL JOIN person
NATURAL JOIN personrolle
WHERE bezeichnung != 'Trainer' AND EXTRACT(MONTH FROM geburtsdatum) = 5 AND
EXTRACT (YEAR FROM geburtsdatum) IN (1977, 1982);
```

ExerciselD: 194161 (category: Verbund, level: mittel)

Geben Sie (mit Duplikaten) Vorname und Nachname aller Spieler aus, die ihr Team in einer Weltmeisterschaft mit 1:0 (bzw. mit 0:1) in Führung geschossen haben (d.h. Eigentore sind nicht zu beachten).

```
SELECT vorname, name FROM tor
JOIN person ON tor.spielerid = person.personid
LEFT JOIN tortyp ON tor.tortypid = tortyp.tortypid
WHERE ((toret1 = 0 AND toret2 = 1) OR (toret1 = 1 AND toret2 = 0))
AND COALESCE(bezeichnung, '0') NOT LIKE 'Eigentor';
```

ExerciselD: 194163 (category: Verbund, level: mittel)

Geben Sie den Turniernamen aller Weltmeisterschaften aus, an denen Ernst Happel teilgenommen hat. Geben Sie zudem die Rollenbezeichnung aus sowie den Namen des Landes, für das Ernst Happel an der Weltmeisterschaft teilgenommen hat. (Tip: Verwenden Sie die UPPER-Funktion, um Vor- und Nachnamen in Großbuchstaben umzuwandeln.)

```
SELECT turniername, bezeichnung, landname FROM turnier
NATURAL JOIN nominierung
NATURAL JOIN personnominierung
NATURAL JOIN personrolle
NATURAL JOIN person
JOIN teamnominierung ON teamnominierungid = teamnominierung.nominierungid
NATURAL JOIN teamland
NATURAL JOIN land
WHERE UPPER(vorname) = 'ERNST' AND UPPER(name) = 'HAPPEL';
```

ExerciselD: 194164 (category: Verbund, level: mittel)

Geben Sie (mit Duplikaten) das (heutige) Alter (in ganzen Jahren!) aller *Spieler* (!) an, die mindestens 30 Jahre alt sind (nehmen Sie an, dass alle Spieler noch leben und verwenden Sie die AGE Funktion). Das Ergebnis soll (vorname, nachname, alter) lauten.

```
SELECT vorname, name AS nachname, CAST(EXTRACT(YEAR FROM AGE(geburtsdatum))
AS INT) AS alter FROM person
NATURAL JOIN personnominierung
NATURAL JOIN personrolle
WHERE bezeichnung IN ('TW', 'VT', 'ST', 'MI')
AND AGE(geburtsdatum) >= INTERVAL '30 YEAR';
```

ExerciselD: 194162 (category: Verbund, level: mittel)

Geben Sie (ohne Duplikate) für alle Spiele aller Weltmeisterschaften die Spielminute (ggf. zzgl. der Minuten der Nachspielzeit) an, in der die Heimmannschaft mit 1:0 in Führung gegangen ist. Geben Sie zudem den Nachnamen und den Ländernamen des Torschützen aus. Benennen Sie die berechnete Spalte 'spielminute'.

ExercisID: 194160 (category: Verbund, level: mittel)

Geben Sie (mit Duplikaten) Vorname und Nachname aller Spieler aus, die ihr Team in einer Weltmeisterschaft mit 2:0 (bzw. mit 0:2) in Führung geschossen haben (d.h. Eigentore sind nicht zu beachten). Geben Sie zudem die SpielId, die Namen der gegnerischen Teams (benennen Sie die Spalten 'heimmannschaft' bzw. 'gastmannschaft') sowie den Namen des betreffenden Turniers aus.

ExercisID: 194165 (category: Verbund, level: schwer)

Geben Sie das Jahr und den Turniernamen aller Weltmeisterschaften aus. Geben Sie zudem (wenn vorhanden) das Datum und die Teilnehmer der Finalspiele der jeweiligen Weltmeisterschaft aus. Das Ergebnis soll (jahr, turniername, datum, heimmannschaft_name, gastmannschaft_name) lauten. (Beachten Sie, dass es im Jahr 1950 kein "Finale" gab; dennoch soll auch diese Weltmeisterschaft - gemäß des ersten Satzes dieser Aufgabe - Bestandteil der Ausgabe sein.)

```
SELECT jahr, turniername, datum, heimmannschaft_name, gastmannschaft_name
FROM turnier
NATURAL LEFT JOIN
  (SELECT turnierid, anpfiff AS datum, t1.name AS heimmannschaft_name,
    t2.name AS gastmannschaft_name FROM spiel
  LEFT JOIN team AS t1 ON spiel.t1id = t1.teamid
  LEFT JOIN team AS t2 ON spiel.t2id = t2.teamid
  WHERE gruppe = 'Finale') AS finale;
```