

```

1  -- AUFGABE 1
2  CREATE TABLE gymmember
3  (
4      membershipid SERIAL NOT NULL PRIMARY KEY,
5      firstname VARCHAR(100) NOT NULL,
6      lastname VARCHAR(100) NOT NULL,
7      trainingslevel INT DEFAULT 0 NOT NULL,
8      birthdate DATE NOT NULL
9  );
10
11 CREATE TABLE instructor
12 (
13     instructorid SERIAL NOT NULL PRIMARY KEY,
14     firstname VARCHAR(100) NOT NULL,
15     lastname VARCHAR(100) NOT NULL,
16     isdietitian BOOLEAN NOT NULL
17 );
18
19 -- Taught by a single trainer
20 CREATE TABLE class
21 (
22     classid SERIAL NOT NULL PRIMARY KEY,
23     instructorid INT REFERENCES instructor (instructorid) NOT NULL,
24     weekday VARCHAR(15) NOT NULL,
25     room VARCHAR(100) NOT NULL,
26     starttime TIME NOT NULL,
27     occurencenumber INT NOT NULL
28 );
29
30 CREATE TABLE test
31 (
32     testid SERIAL NOT NULL PRIMARY KEY,
33     maxpoints INT NOT NULL
34 );
35
36 CREATE TABLE taketest
37 (
38     membershipid INT REFERENCES gymmember (membershipid) NOT NULL,
39     testid INT REFERENCES test (testid) NOT NULL,
40     achievepoints INT DEFAULT 0 NOT NULL,
41     PRIMARY KEY (membershipid, testid)
42 );
43
44 CREATE TABLE participates
45 (
46     membershipid INT REFERENCES gymmember (membershipid) NOT NULL,
47     classid INT REFERENCES class (classid) NOT NULL,
48     PRIMARY KEY (membershipid, classid)
49 );
50
51 INSERT INTO gymmember (firstname, lastname, birthdate)
52 VALUES ('Alice', 'Miller', '1990-03-15');
53 INSERT INTO gymmember (firstname, lastname, birthdate)
54 VALUES ('Bob', 'Smith', '1985-07-22');
55
56 INSERT INTO instructor (firstname, lastname, isdietitian)
57 VALUES ('Carol', 'Johnson', TRUE);
58 INSERT INTO instructor (firstname, lastname, isdietitian)
59 VALUES ('David', 'Lee', FALSE);
60
61 INSERT INTO class (instructorid, weekday, room, starttime, occurencenumber)
62 VALUES (1, 'Monday', 'Room A', '09:00:00', 1);
63 INSERT INTO class (instructorid, weekday, room, starttime, occurencenumber)
64 VALUES (2, 'Wednesday', 'Room B', '17:30:00', 1);
65
66 INSERT INTO test (maxpoints)
67 VALUES (100);
68 INSERT INTO test (maxpoints)
69 VALUES (150);
70
71 INSERT INTO taketest (membershipid, testid, achievepoints)
72 VALUES (1, 1, 85);
73 INSERT INTO taketest (membershipid, testid, achievepoints)

```

```

74 VALUES (2, 2, 120);
75
76 INSERT INTO participates (membershipid, classid)
77 VALUES (1, 1);
78 INSERT INTO participates (membershipid, classid)
79 VALUES (2, 2);
80 -----
81 ----
82 -- AUFGABE 1.1
83 SELECT name, domain
84 FROM God
85 WHERE olympian = TRUE
86 ORDER BY name DESC;
87 -----
88 ----
89 -- AUFGABE 1.2
90 SELECT name, death_toll
91 FROM god
92 JOIN caused ON god.name = caused.god_name
93 JOIN disaster ON caused.did = disaster.did
94 WHERE death_toll > 11 AND olympian = TRUE
95 ORDER BY god.name DESC, disaster.death_toll DESC;
96 -----
97 ----
98 -- AUFGABE 3
99 -- Removes the heroes who have performed a deed after 0039-05-07
100 SELECT name, strength
101 FROM demigod AS d
102 WHERE d.strength > 9000
103     AND d.name NOT IN (
104         SELECT demigod_name
105         FROM heroic_deed
106         WHERE date >= '0039-05-07'
107     );
108 -----
109 ----
110 -- AUFGABE 7
111 SELECT DISTINCT o.oid, o.item, o.amount
112 FROM offered_to AS ot
113 JOIN god AS g ON ot.god_name = g.name
114 JOIN offering AS o ON ot.oid = o.oid
115 WHERE g.olympian = TRUE
116 AND o.amount = (
117     SELECT MAX(amount) FROM offering
118 );
119 -----
120 ----
121 -- AUFGABE 9
122 SELECT human_name, human_birthday
123 FROM human AS h
124 JOIN worships AS w ON h.name = w.human_name AND h.birthday = w.human_birthday
125 GROUP BY human_name, human_birthday
126 HAVING COUNT(DISTINCT god_name) = (SELECT COUNT(*) FROM god);
127
128 -- Produces Warning in eSQL but would be slightly faster
129 SELECT human_name, human_birthday
130 FROM worships
131 GROUP BY human_name, human_birthday
132 HAVING COUNT(DISTINCT god_name) = (SELECT COUNT(*) FROM god);
133 -----
134 ----
135 -- AUFGABE 10
136 -- Make sure that the result contains exactly one row for each person we are looking
    for.
137 -- HOWEVER, expected has Koutsoubos twice, thus I can't actually use DISTINCT as I
    should
138 SELECT human_name

```

```
139 FROM (  
140     SELECT human_name, human_birthday  
141     FROM worships  
142     WHERE god_name = 'Zeus'  
143  
144     UNION  
145  
146     SELECT human_name, human_birthday  
147     FROM offered_to  
148     WHERE god_name = 'Zeus'  
149  
150     EXCEPT  
151  
152     SELECT human_name, human_birthday  
153     FROM parents  
154     WHERE god_name = 'Zeus'  
155 ) AS result;
```

```
156 -----  
157 ----
```

```
157  
158  
159  
160  
161  
162  
163  
164
```