

```

1  -- Kevin Hance
2  -- DBMS (CPSC 321)
3  -- HW6 (hw6.sql)
4
5  DROP TABLE IF EXISTS border;
6  DROP TABLE IF EXISTS city;
7  DROP TABLE IF EXISTS province;
8  DROP TABLE IF EXISTS country;
9
10 CREATE TABLE country(
11     country_code VARCHAR(10),
12     country_name VARCHAR(50),
13     gdp INT UNSIGNED,
14     inflation FLOAT,
15     PRIMARY KEY (country_code)
16 );
17
18 CREATE TABLE province(
19     province_name VARCHAR(50),
20     country_code VARCHAR(10),
21     area FLOAT UNSIGNED,
22     PRIMARY KEY (province_name, country_code),
23     FOREIGN KEY (country_code) REFERENCES country (country_code)
24 );
25
26 CREATE TABLE city(
27     city_name VARCHAR(50),
28     province_name VARCHAR(50),
29     country_code VARCHAR(10),
30     population INT UNSIGNED,
31     PRIMARY KEY (city_name, province_name, country_code),
32     FOREIGN KEY (province_name, country_code) REFERENCES province (province_name,
33     country_code)
34 );
35
36 CREATE TABLE border(
37     country_code_1 VARCHAR(10),
38     country_code_2 VARCHAR(10),
39     border_length FLOAT UNSIGNED,
40     PRIMARY KEY (country_code_1, country_code_2),
41     FOREIGN KEY (country_code_1) REFERENCES country (country_code),
42     FOREIGN KEY (country_code_2) REFERENCES country (country_code)
43 );
44
45 INSERT INTO country VALUES ('OS', 'Oswaldo', 78000, 6.7);
46 INSERT INTO country VALUES ('RL', 'Renlandia', 54000, 1.7);
47 INSERT INTO country VALUES ('GN', 'Geneva', 65000, 2.1);
48
49 INSERT INTO province VALUES ('Oslodo', 'OS', 8712);
50 INSERT INTO province VALUES ('St. Janice', 'OS', 76650);
51
52 INSERT INTO province VALUES ('Huport', 'RL', 91030);
53 INSERT INTO province VALUES ('Flaubury', 'RL', 5690);
54
55 INSERT INTO province VALUES ('Antalens', 'GN', 72003);
56 INSERT INTO province VALUES ('Huport', 'GN', 54041);
57
58 INSERT INTO city VALUES ('Tesa', 'Oslodo', 'OS', 964);
59 INSERT INTO city VALUES ('Sluurgan', 'Oslodo', 'OS', 5919);
60 INSERT INTO city VALUES ('Belogonia', 'St. Janice', 'OS', 97635);
61 INSERT INTO city VALUES ('Britano', 'St. Janice', 'OS', 33434);
62
63 INSERT INTO city VALUES ('Piolas', 'Huport', 'RL', 46626);
64 INSERT INTO city VALUES ('Stombus', 'Huport', 'RL', 49384);
65 INSERT INTO city VALUES ('Outling', 'Flaubury', 'RL', 127);
66 INSERT INTO city VALUES ('Whita', 'Flaubury', 'RL', 52743);
67
68 INSERT INTO city VALUES ('Blumore', 'Antalens', 'GN', 54000);
69 INSERT INTO city VALUES ('Juefbert', 'Antalens', 'GN', 120);

```

```

69 INSERT INTO city VALUES ('Tesa', 'Huport', 'GN', 11043);
70 INSERT INTO city VALUES ('Stombus', 'Huport', 'GN', 13299);
71
72 INSERT INTO border VALUES ('OS', 'GN', 4800);
73 INSERT INTO border VALUES ('GN', 'RL', 4500);
74 INSERT INTO border VALUES ('RL', 'OS', 1200);
75 INSERT INTO border VALUES ('GN', 'OS', 4800);
76 INSERT INTO border VALUES ('RL', 'GN', 4500);
77 INSERT INTO border VALUES ('OS', 'GN', 1200);
78
79
80 -- group concat question
81 SELECT * FROM city;
82 SELECT GROUP_CONCAT(city_name) FROM city WHERE population > 1000;
83
84 -- stddev pop question
85 SELECT * FROM city;
86 SELECT STDDEV_POP(population), AVG(population) FROM city WHERE population > 2000;
87
88 -- union vs union all question
89 SELECT * FROM city;
90 (SELECT * FROM city WHERE population > 1000)
91 UNION
92 (SELECT * FROM city WHERE population < 20000);
93
94 (SELECT * FROM city WHERE population > 1000)
95 UNION ALL
96 (SELECT * FROM city WHERE population < 20000);
97
98 -- TECHNICAL WORK
99
100 -- question 1
101 -- all create/insert statements
102
103 -- question 2
104 SET @gdp = 60000;
105 SET @inflation = 3.0;
106 SET @area = 10000;
107
108 SELECT DISTINCT c.country_name, c.country_code, c.gdp, c.inflation
109 FROM province p, country c
110 WHERE c.country_code = p.country_code
111 AND c.gdp > @gdp
112 AND c.inflation > @inflation
113 AND p.area < @area;
114
115 -- question 3
116 SELECT DISTINCT c.country_name, c.country_code, c.gdp, c.inflation
117 FROM province p INNER JOIN country c ON c.country_code = p.country_code
118 WHERE c.gdp > @gdp
119 AND c.inflation > @inflation
120 AND p.area < @area;
121
122 --question 4
123 SET @population = 1000;
124
125 SELECT p.country_code, c.country_name, p.province_name, ct.city_name, ct.population,
126 p.area
127 FROM country c, province p, city ct
128 WHERE p.country_code = c.country_code
129 AND p.province_name = ct.province_name
130 AND ct.population < @population;
131
132 --question 5
133 SELECT p.country_code, c.country_name, p.province_name, ct.city_name, ct.population,
134 p.area
135 FROM country c
136 INNER JOIN province p ON p.country_code = c.country_code

```

```
136 INNER JOIN city ct ON p.province_name = ct.province_name
137 WHERE ct.population < @population;
138
139 -- question 6
140 SELECT SUM(area) FROM province;
141
142 -- question 7
143 SELECT MIN(gdp), MAX(gdp), AVG(gdp), MIN(inflation), MAX(inflation), AVG(inflation)
144 FROM country;
145
146 -- question 8
147 SET @country_code = 'OS';
148 SELECT COUNT(*), AVG(population) FROM city WHERE country_code = @country_code;
149
150 -- question 9
151 SET @province_name = "Oslo";
152 SELECT AVG(population) FROM city WHERE province_name = @province_name;
153
```