Lab work 1 Report

Lei Liu, 9669373

November 19, 2015

1 Task 1: Tuple-relational calculus expression

(1) Return the name of any country that has a lake.

$$\{A|\ \exists L \in geo\_lake\ \exists C \in country\ (L.country = C.code\ \wedge\ A.name = C.name)\}$$

(2)Return all the information available on cities whose population is between 3 and 5 million inhabitants.

$$\{C|\ C \in city\ (C.populatoin > 3000000\ \wedge\ C.population < 5000000)\}$$

(3)Return the country code and the continent of every country not in Europe or in Australia/Oceania.

$$\{A|\exists C \in country\ \exists cont \in continent(C.code = cont.country$$

$$\wedge\ cont.continent \neq'\ Europe'\ \wedge\ cont.continent \neq'\ Australia/Oceania')\}$$

(4)Return the names of countries that also give their name a province.

$$\{A|\ \exists C \in country\ \exists P \in province(C.code = P.country\ \wedge\ C.name = P.name\ \wedge\ A.name = C.name)\}$$

(5)Return countries that are not landlocked (i.e., have a sea coast).

$$\{A|\ \exists C \in country\ \exists S \in geo\_sea(C.code = S.country\ \wedge\ A.name = C.name)\}$$

2 Task 2: Relational-algebraic expression

(6)Return countries that are not landlocked (i.e., have a sea coast).

$$\backslash project\_\{name\}\ (\backslash project\_\{name,\ code\}\ country\ \backslash join\_\{country = code\}\ geo\_sea);$$

(7)Return the names of all lakes, rivers and seas.

$$\backslash rename\_\{L\_R\_S\_names\}(\backslash project\_\{name\}\ lake)$$

$$\backslash union(\backslash project\_\{name\}\ river)\ \backslash union\ (\backslash project\_\{name\}\ sea);$$

(8)Return the average length of a river.

$$\backslash project\_\{avg(length)\}\ river;$$

(9)Return the name of countries that have more than 10 islands.

$$\pi_{name,\ num\_island}\ (\sigma_{code=country\ \wedge\ num\_island>10}$$

$$({}_{country}\gamma_{count(island)\rightarrow num\_island}(geo\_island)\ \bowtie\ country))$$

(10)Return the length of all rivers in Great Britain.

$$\backslash project\_\{rname,\ length\}\backslash select\_\{country =' GB'\}$$

$$(\backslash rename\_\{rname,\ length\}\ \backslash project\_\{name,\ length\}\ river$$

$$\backslash join$$

$$\backslash rename\_\{rname,\ country\}\backslash project\_\{river,\ country\}\ geo\_river);$$

(11)Return the name of the countries that have the 10 longest total length of rivers.

$$\pi_{countryname\ limit\ 10}($$

$$\tau_{length}({}_{countryname}\gamma_{sum(length)\rightarrow length}($$

$$\sigma_{name=river}(river \bowtie$$

$$\pi_{country.name\rightarrow countryname,river}($$

$$\sigma_{country=code}(geo\_river \bowtie country))))))$$

Question (6),(7),(8), and (10) were executed via the RA software, and their output were printed in file: *"Lei_Liu_LW1_RA_Result.txt"*.

3 Task 3: SQL expression

(12)Return the names of up to 10 countries and the value corresponding to half the countrys population.

$$select\ name,\ (population/2)\ as\ half\_of\_population\ from\ country\ limit\ 10;$$

(13)Return all the information available about cities whose name is Manchester.

$$select\ *\ from\ city\ where\ name\ =\ 'Manchester';$$

(14)Return the name of cities whose name starts with the substring 'Man.

$$select\ name\ from\ city\ where\ name\ like\ 'Man\%';$$

(15)Return the name of both countries with Buddhist populations and organizations, established after 1st December 1994, that the country is a member of.

$$select\ c.name\ as\ country,\ organization\ from\ country\ c\ join$$

$$(select\ distinct\ m.country\ as\ country\_code,\ o.name\ as\ organization$$

$$from\ organization\ o\ join\ ismember\ m\ on\ m.organization = o.abbreviation$$

$$where\ o.established >' 1994-12-01'\ and\ m.country\ in$$

$$(\ select\ country\ from\ religion\ where\ name =' Buddhist'))mem$$

$$onc.code = mem.country\_code;$$

(16)Return the name of each country with the number of islands in it.

$$select\ c.name,\ count(geo.island)\ as\ num\_island\ from\ country\ c$$

$$join\ geo\_island\ geo\ on\ c.code = geo.country\ group\ by\ geo.island;$$

Executed result were logged in file: *Lei_Liu_LW1_SQL.log*
Also, the sql script file is: *Lei_Liu_LW1_SQL.sql*