1.

SELECT COUNTRY.NAME

FROM COUNTRY

JOIN ECONOMY

ON COUNTRY.CODE=ECONOMY.COUNTRY

WHERE ECONOMY.AGRICULTURE > 50;

2.

SELECT NAME FROM (SELECT NAME, (POPULATION \* POWER((1+NVL(POPULATION\_GROWTH,0)),5)) AS AFTER\_5 FROM (SELECT COUNTRY.POPULATION,POPULATION.POPULATION\_GROWTH,COUNTRY.NAME

FROM (COUNTRY

JOIN POPULATION

ON COUNTRY.CODE=POPULATION.COUNTRY) ) ORDER BY AFTER\_5 DESC ) WHERE ROWNUM <= 5 ;

3.

4.

SELECT C.NAME

FROM (COUNTRY C

JOIN RELIGION R

ON C.CODE=R.COUNTRY) WHERE PERCENTAGE > 80 AND C.NAME IN (

SELECT CNAME FROM (SELECT NAME AS CNAME, COUNT(DISTINCT RELNAME) AS RELNUM FROM (SELECT COUNTRY.NAME,RELIGION.NAME AS RELNAME,RELIGION.PERCENTAGE

FROM (COUNTRY

JOIN RELIGION

ON COUNTRY.CODE=RELIGION.COUNTRY)) GROUP BY NAME) WHERE RELNUM > 4

)

5.

SELECT

SUM(length)

FROM

(

SELECT

\*

FROM

borders

WHERE

country1 IN (

SELECT

code

FROM

country

WHERE

name = 'China'

)

OR country2 IN (

SELECT

code

FROM

country

WHERE

name = 'China'

)

);

7.

SELECT DISTINCT NAME FROM (SELECT L.NAME,L.ELEVATION,G.COUNTRY

FROM GEO\_LAKE G

JOIN LAKE L

ON G.LAKE= L.NAME

WHERE G.COUNTRY IN (

SELECT CODE FROM COUNTRY WHERE NAME = 'United States'

)) WHERE ELEVATION > (SELECT AVG(ELEVATION) FROM LAKE) ;

8.

SELECT \* FROM (SELECT T.NAME AS MOUNTAIN\_NAME ,T.PROVINCE,J.POPULATION\_DENSITY FROM (SELECT M.NAME, M.TYPE,G.PROVINCE, G.COUNTRY FROM

MOUNTAIN M JOIN GEO\_MOUNTAIN G ON M.NAME = G.MOUNTAIN

WHERE TYPE='volcano') T JOIN

(SELECT NAME, (POPULATION / AREA) AS POPULATION\_DENSITY FROM PROVINCE) J

ON T.PROVINCE = J.NAME

ORDER BY POPULATION\_DENSITY DESC) WHERE POPULATION\_DENSITY IN (

(SELECT MAX(POPULATION\_DENSITY) FROM

(SELECT T.NAME AS MOUNTAIN\_NAME ,T.PROVINCE,J.POPULATION\_DENSITY FROM (SELECT M.NAME, M.TYPE,G.PROVINCE, G.COUNTRY FROM

MOUNTAIN M JOIN GEO\_MOUNTAIN G ON M.NAME = G.MOUNTAIN

WHERE TYPE='volcano') T JOIN

(SELECT NAME, (POPULATION / AREA) AS POPULATION\_DENSITY FROM PROVINCE) J

ON T.PROVINCE = J.NAME

ORDER BY POPULATION\_DENSITY DESC))

)

9.

SELECT \* FROM (SELECT T2.PROVINCE, T2.ISLAND\_COUNT, T2.COUNTRY, E.GDP FROM (

SELECT T.PROVINCE, T.ISLAND\_COUNT, P.COUNTRY FROM (

SELECT \* FROM (SELECT PROVINCE, COUNT(ISLAND) AS ISLAND\_COUNT FROM GEO\_ISLAND GROUP BY PROVINCE) WHERE ISLAND\_COUNT > 2

) T JOIN

PROVINCE P ON

T.PROVINCE = P.NAME

) T2 JOIN

ECONOMY E ON

T2.COUNTRY = E.COUNTRY) WHERE GDP > 1000000

10.

SELECT NAME, LENGTH FROM (SELECT \* FROM (SELECT \* FROM RIVER WHERE SEA = 'Atlantic Ocean') R JOIN

(SELECT RIVER, COUNT(NAME) AS LAKE\_COUNT FROM LAKE GROUP BY RIVER) L ON

R.NAME = L.RIVER ORDER BY R.LENGTH DESC) WHERE ROWNUM <= 2

11.

SELECT NAME FROM COUNTRY WHERE CODE IN (

SELECT DISTINCT R\_COUNTRY FROM (SELECT LAKE, COUNTRY AS L\_COUNTRY FROM GEO\_LAKE WHERE LAKE IN (

SELECT LAKE FROM (SELECT LAKE, COUNT(PROVINCE) AS PROVINCE\_COUNT FROM GEO\_LAKE GROUP BY LAKE) WHERE PROVINCE\_COUNT > 3

)) L JOIN

(SELECT \* FROM (SELECT COUNTRY AS R\_COUNTRY, COUNT(RIVER) AS RIVER\_COUNT FROM GEO\_RIVER GROUP BY COUNTRY ) WHERE RIVER\_COUNT > 3) R ON

L.L\_COUNTRY = R.R\_COUNTRY

)

13.

SELECT CONTINENT, MAX(HIGHEST\_PEAK) AS HIGHEST\_ELEVATION FROM (SELECT \* FROM

(SELECT COUNTRY, MAX(ELEVATION)AS HIGHEST\_PEAK FROM

(SELECT NAME, ELEVATION, COUNTRY FROM MOUNTAIN M JOIN GEO\_MOUNTAIN GM ON M.NAME = GM.MOUNTAIN)

GROUP BY COUNTRY) P JOIN

(SELECT COUNTRY,CONTINENT FROM ENCOMPASSES) C ON

P.COUNTRY = C.COUNTRY

) GROUP BY CONTINENT

14.

SELECT C.NAME, TEMP.MAX\_ELEVATION, TEMP.MAX\_DEPTH FROM

COUNTRY C JOIN

(SELECT MOUNTAIN.COUNTRY, MOUNTAIN.MAX\_ELEVATION, SEA.MAX\_DEPTH FROM

((SELECT COUNTRY, MAX(ELEVATION) AS MAX\_ELEVATION FROM

(SELECT \* FROM MOUNTAIN M JOIN GEO\_MOUNTAIN G ON M.NAME = G.MOUNTAIN)

GROUP BY COUNTRY

) MOUNTAIN JOIN

(

SELECT COUNTRY, MAX(DEPTH) AS MAX\_DEPTH FROM

(SELECT \* FROM SEA S JOIN GEO\_SEA GS ON S.NAME = GS.SEA)

GROUP BY COUNTRY

) SEA

ON MOUNTAIN.COUNTRY = SEA.COUNTRY

) WHERE MOUNTAIN.MAX\_ELEVATION > SEA.MAX\_DEPTH

) TEMP

ON C.CODE = TEMP.COUNTRY

25.

SELECT NAME FROM ORGANIZATION WHERE ABBREVIATION IN

(SELECT ABBREVIATION FROM (SELECT DISTINCT ABBREVIATION FROM (SELECT ABBREVIATION, NAME FROM ORGANIZATION) O

JOIN

(SELECT \* FROM ISMEMBER WHERE COUNTRY IN

(SELECT COUNTRY FROM ENCOMPASSES WHERE CONTINENT = 'Asia')

) N

ON O.ABBREVIATION = N.ORGANIZATION

) WHERE ABBREVIATION NOT IN (

SELECT NON\_ASIA.ABBREVIATION FROM (SELECT DISTINCT ABBREVIATION FROM (SELECT ABBREVIATION, NAME FROM ORGANIZATION) O

JOIN

(SELECT \* FROM ISMEMBER WHERE COUNTRY IN

(SELECT COUNTRY FROM ENCOMPASSES WHERE CONTINENT = 'Asia')

) N

ON O.ABBREVIATION = N.ORGANIZATION)ASIA

JOIN

(SELECT DISTINCT ABBREVIATION FROM (SELECT ABBREVIATION, NAME FROM ORGANIZATION) O

JOIN

(SELECT \* FROM ISMEMBER WHERE COUNTRY IN

(SELECT COUNTRY FROM ENCOMPASSES WHERE CONTINENT != 'Asia')

) N

ON O.ABBREVIATION = N.ORGANIZATION) NON\_ASIA

ON ASIA.ABBREVIATION = NON\_ASIA.ABBREVIATION

))

--NAME OF ORGANIZATONS HAVING ASIAN MEMBER COUNTRIES

24.

SELECT NAME AS "Country Name", POPULATION\_DENSITY AS "Population Density", ((POPULATION / WP)\*100) AS "Percentage" FROM

(SELECT NAME, POPULATION, (POPULATION / AREA) AS Population\_Density, (SELECT SUM(POPULATION) AS WORLD\_POP FROM COUNTRY) AS WP

FROM COUNTRY

ORDER BY (POPULATION\_DENSITY) DESC ) WHERE ROWNUM <= 10

23.

SELECT \* FROM

(SELECT DISTINCT I.PROVINCE, I.COUNT\_ISLANDS, G.COUNTRY AS COUNTRY\_CODE FROM

(SELECT COUNT(ISLAND)AS COUNT\_ISLANDS, PROVINCE FROM GEO\_ISLAND GROUP BY PROVINCE ORDER BY COUNT\_ISLANDS DESC) I

JOIN GEO\_ISLAND G

ON I.PROVINCE = G.PROVINCE

ORDER BY COUNT\_ISLANDS DESC)

WHERE COUNT\_ISLANDS = (SELECT MAX(COUNT\_ISLANDS)FROM

(SELECT COUNT(ISLAND)AS COUNT\_ISLANDS, PROVINCE FROM GEO\_ISLAND GROUP BY PROVINCE ORDER BY COUNT\_ISLANDS DESC)

)

22.

SELECT RIVER\_NAME,LENGTH FROM

(SELECT DISTINCT G.RIVER AS RIVER\_NAME, G.COUNTRY, R.LENGTH FROM

((SELECT RIVER, COUNTRY FROM GEO\_RIVER)G

JOIN

(SELECT NAME, LENGTH FROM RIVER) R

ON G.RIVER = R.NAME) WHERE

COUNTRY IN (SELECT COUNTRY FROM ENCOMPASSES WHERE CONTINENT = 'America')

ORDER BY R.LENGTH DESC

)

WHERE LENGTH = (

SELECT MAX (R.LENGTH) FROM

((SELECT RIVER, COUNTRY FROM GEO\_RIVER)G

JOIN

(SELECT NAME, LENGTH FROM RIVER) R

ON G.RIVER = R.NAME) WHERE

COUNTRY IN (SELECT COUNTRY FROM ENCOMPASSES WHERE CONTINENT = 'America')

)

21.

SELECT RIVER FROM

(SELECT RIVER,COUNT(\*) AS PROV\_COUNT FROM GEO\_RIVER GROUP BY RIVER,COUNTRY)

WHERE PROV\_COUNT >= 12

20.

SELECT T.NAME,G.GDP FROM

(SELECT COUNTRY,GDP FROM ECONOMY WHERE COUNTRY IN (

SELECT COUNTRY FROM RELIGION WHERE COUNTRY IN (

SELECT COUNTRY FROM ISMEMBER WHERE ORGANIZATION = 'NATO'

) AND NAME = 'Muslim' AND PERCENTAGE > 5

)

) G JOIN

(SELECT NAME,CODE FROM COUNTRY) T

ON G.COUNTRY = T.CODE

19.

SELECT I.NAME,I.AREA FROM

(SELECT NAME,AREA FROM ISLAND WHERE AREA > 1000) I JOIN

(SELECT \* FROM GEO\_ISLAND WHERE COUNTRY IN (SELECT COUNTRY FROM ENCOMPASSES WHERE CONTINENT = 'Africa' )) G

ON I.NAME = G.ISLAND

ORDER BY I.AREA DESC

18.

SELECT NAME FROM COUNTRY WHERE CODE IN (

SELECT COUNTRY FROM (SELECT COUNTRY, COUNT(CONTINENT) AS CONT\_COUNT FROM ENCOMPASSES GROUP BY COUNTRY) WHERE CONT\_COUNT > 1

)

17.

SELECT

(SELECT SUM(AREA) AS "top10" FROM (SELECT NAME,AREA FROM COUNTRY ORDER BY AREA DESC) WHERE ROWNUM <= 10 ) "top10",

(SELECT SUM(AREA) AS "rest\_world" FROM (SELECT NAME,AREA FROM COUNTRY ORDER BY AREA ASC) WHERE ROWNUM < (SELECT COUNT(\*) FROM

(SELECT NAME,AREA FROM COUNTRY ORDER BY AREA DESC)

) - 9 ) "rest\_world",

((SELECT SUM(AREA) AS "top10" FROM (SELECT NAME,AREA FROM COUNTRY ORDER BY AREA DESC) WHERE ROWNUM <= 10 )-

((SELECT SUM(AREA) AS "rest\_world" FROM (SELECT NAME,AREA FROM COUNTRY ORDER BY AREA ASC) WHERE ROWNUM < (SELECT COUNT(\*) FROM

(SELECT NAME,AREA FROM COUNTRY ORDER BY AREA DESC)

) - 9 ))) "difference"

FROM DUAL

16.

SELECT NAME FROM COUNTRY WHERE CODE IN (

SELECT COUNTRY FROM CITY WHERE NAME IN (

SELECT CAPITAL FROM COUNTRY

) AND LATITUDE > 0 AND POPULATION < 10000

)

15.

SELECT NAME,CONTINENT FROM

(SELECT CL1.NAME, CL1.LATITUDE,CO1.CONTINENT FROM

(SELECT NAME,LATITUDE FROM CITY WHERE COUNTRY NOT IN

(SELECT COUNTRY AS COUNTRY\_ENCP FROM ENCOMPASSES WHERE CONTINENT = 'Asia')

) CL1

JOIN

(SELECT CONTINENT,MAX(LATITUDE) AS MAX\_LATITUDE FROM

(SELECT CI.COUNTRY,CI.NAME AS CITY\_NAME, C.CONTINENT,CI.LATITUDE FROM

(

SELECT COUNTRY AS COUNTRY\_ENCP,CONTINENT FROM ENCOMPASSES WHERE CONTINENT != 'Asia' AND COUNTRY NOT IN (

SELECT COUNTRY AS COUNTRY\_ENCP FROM ENCOMPASSES WHERE CONTINENT = 'Asia'

)) C

JOIN

(SELECT \* FROM CITY WHERE COUNTRY IN (

SELECT COUNTRY FROM ENCOMPASSES WHERE CONTINENT != 'Asia' AND COUNTRY NOT IN (

SELECT COUNTRY AS COUNTRY\_ENCP FROM ENCOMPASSES WHERE CONTINENT = 'Asia'

)

)) CI

ON C.COUNTRY\_ENCP = CI.COUNTRY)

GROUP BY CONTINENT) CO1

ON CL1.LATITUDE = CO1.MAX\_LATITUDE )

ORDER BY LATITUDE DESC