## Task.

1)Create database from nation.sql script

2)Using the data from nation database create the table ‘countryRank’ that will display

Ranking of countries by its area from largest to smallest. Ranking shold be calculated for regions and continent.

a)National holiday field shouldn’t contain NULL values, put ‘no data insead’

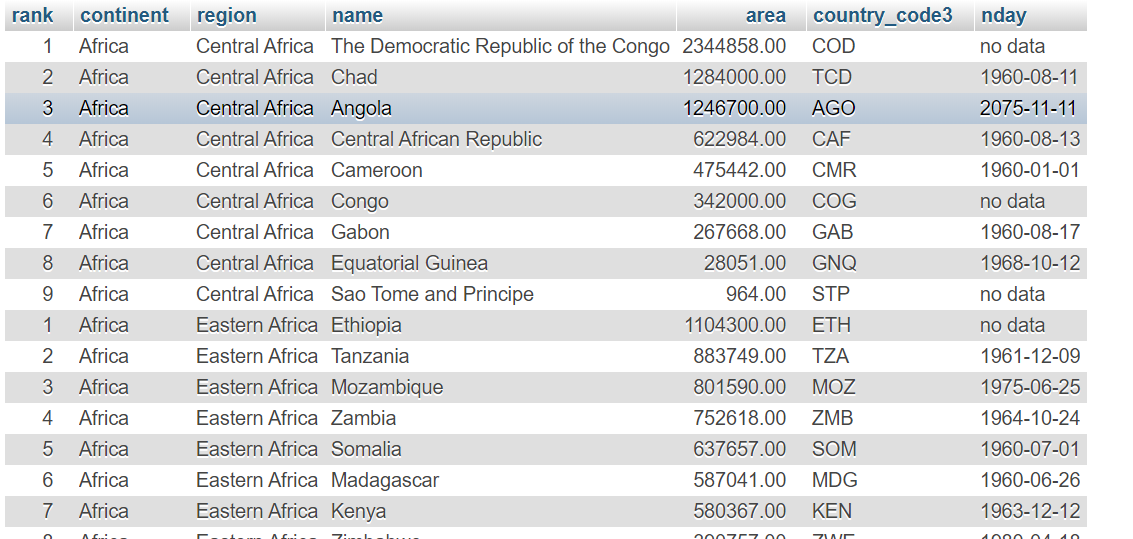
b)If notional holiday contains the date in future it should be changed to current date

c)Get rid of the duplicates

Provide sql scrpits and screenshots as evidence.

3)Answer the question – what can be done on a database level to avoid problems with duplicates and Null values?

The expected result should look like this.



**CREATE** **OR** **REPLACE** **TABLE** countryRank **AS**

**SELECT** **DISTINCT**

**RANK** () **OVER**(**PARTITION** **BY** r.region\_id, c2.continent\_id **ORDER** **BY** area **DESC**) **as** **rank**,

c2.name **as** continent, r.name **as** region, c.name , c.area ,c.country\_code3,

**CASE** **WHEN** c.national\_day **IS** **NULL** **THEN** 'no data'

**WHEN** c.national\_day >= **CURDATE**() **THEN** **CAST**(**CURDATE**() **as** **VARCHAR**(10))

**ELSE** **CAST**(c.national\_day **as** **VARCHAR**(10))

**END** **as** nday

**FROM** countries c

**JOIN** regions r **ON** c.region\_id = r.region\_id

**JOIN** continents c2 **ON** r.continent\_id = c2.continent\_id

**ORDER** **BY** c2.name , r.name , **rank  
  
A screenshot of a computer

Description automatically generated**

**Graphical user interface, application

Description automatically generated**

**CREATE** **OR** **REPLACE** **TABLE** countryRank

**SELECT**

**RANK** () **OVER**(**PARTITION** **BY** r.region\_id, c2.continent\_id **ORDER** **BY** area **DESC**) **as** **rank**,

c2.name **as** continent, r.name **as** region,c.name , c.area ,c.country\_code3 , **CAST**(c.national\_day **as** **VARCHAR**(10)) **as** nday

**FROM** countries c

**JOIN** regions r **ON** c.region\_id = r.region\_id

**JOIN** continents c2 **ON** r.continent\_id = c2.continent\_id

**ORDER** **BY** c2.name , r.name , **rank**;

**UPDATE** countryRank

**SET** nday = 'no data'

**WHERE** nday **IS** **NULL**;

**UPDATE** countryRank

**SET** nday = **CAST**(**CURDATE**() **as** **VARCHAR**(10))

**WHERE** nday != 'no data'

**AND** **CAST**(nday **AS** **DATE**) >= **CURDATE**();

**ALTER** **IGNORE** **TABLE** countryRank **ADD** **UNIQUE** **INDEX**(`name`);

**ALTER** **TABLE** nation.countryRank **MODIFY** **COLUMN** nday **varchar**(10) **DEFAULT** 'no data' **NOT** **NULL**;

**SELECT** \*

**FROM** countryRank cr;

Table

Description automatically generated

-2-

**CREATE OR REPLACE TABLE countryRank**

**SELECT**

**RANK () OVER(PARTITION BY r.region\_id, c2.continent\_id ORDER BY area DESC) as rank,**

**c2.name as continent, r.name as region,c.name , c.area ,c.country\_code3 , CAST(c.national\_day as VARCHAR(10)) as nday**

**FROM countries c**

**JOIN regions r ON c.region\_id = r.region\_id**

**JOIN continents c2 ON r.continent\_id = c2.continent\_id**

**ORDER BY c2.name , r.name , rank**

Graphical user interface, text, application, email

Description automatically generated

-a-

**UPDATE countryRank**

**SET nday = 'no data'**

**WHERE nday IS NULL**

Text, table

Description automatically generated with medium confidence

-b-

**UPDATE countryRank**

**SET nday = CAST(CURDATE() as VARCHAR(10))**

**WHERE nday != 'no data'**

**AND CAST(nday AS DATE) >= CURDATE()**

**Graphical user interface, text, application

Description automatically generated**

-c-

**ALTER IGNORE TABLE countryRank ADD UNIQUE INDEX(`name`);**

Graphical user interface, text, application

Description automatically generated

-3-

To avoid problems with duplicates we can set UNIQUE constraint or create a UNIQUE INDEX

To avoid problems with NULL values we should set NOT NULL constraint