Data Science - Practice 3 (R Basic II)

Make sure you not only just "write down" the R code but also "explain the answer with your own language". All answers without explanation will not be accepted.

Problem

Import 'country.RData'.

< Question 1 - country data >

(1) Use tapply to find the average GDP for each continent and (2) the average CO2 for each continent. (3) To compare both results, create a DataFrame as shown below.

Expected Result

(1)

```
     Africa          Asia        Europe North America       Oceania South America
   4913.677     21864.382     34015.457     16394.923     23803.250     15333.556
```

(2)

```
     Africa          Asia        Europe North America       Oceania South America
   35154.68     529585.44     154650.86     498421.38     112391.50     123327.78
```

(3)

```
                    GDP       CO2
Africa         4913.677  35154.68
Asia          21864.382 529585.44
Europe        34015.457 154650.86
North America 16394.923 498421.38
Oceania       23803.250 112391.50
South America 15333.556 123327.78
```

< Question 2 – country data>

(1) Use 'iterative and conditional statements' to create the variable called 'BTS_country_d' to 'country' DataFrame. For South Korea, the value of 'BTS_country' is 'BTS_Home', and the rest should be empty (NA). (2) Show the case of South Korea and China.

Expected Result

(1)

```
> country$BTS_country_d
  [1] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [10] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [19] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [28] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [37] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [46] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [55] NA          NA          NA          NA          NA          NA          NA          "BTS_Home"  NA
 [64] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [73] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [82] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [91] NA          NA          NA          NA          NA          NA          NA          NA          NA
[100] NA          NA          NA          NA          NA          NA          NA          NA          NA
[109] NA          NA          NA          NA          NA          NA          NA          NA          NA
[118] NA          NA          NA          NA          NA          NA          NA          NA          NA
```

(2)

```
    code country_name continent   GDP life_expect population     CO2 battle_death child.per.woman
27   chn        China      Asia 14369       76.69 1410000000 9710000        0.000            1.62
62   kor  South Korea      Asia 35020       82.52   51000000  618000        0.113            1.30
   programmable.aid BTS_country_d
27         1072.494          <NA>
62          850.567      BTS_Home
```

< Question 3 – country data>

(1) Do the same operation in Question 3, but use 'vectorized operation'. Here, name the variable 'BTS_country' (2) Show the case of South Korea and China.

Expected Result

(1)

```
> country$BTS_country
  [1] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [10] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [19] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [28] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [37] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [46] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [55] NA          NA          NA          NA          NA          NA          NA          "BTS_Home"  NA
 [64] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [73] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [82] NA          NA          NA          NA          NA          NA          NA          NA          NA
 [91] NA          NA          NA          NA          NA          NA          NA          NA          NA
[100] NA          NA          NA          NA          NA          NA          NA          NA          NA
[109] NA          NA          NA          NA          NA          NA          NA          NA          NA
[118] NA          NA          NA          NA          NA          NA          NA          NA          NA
```

(2)

```
    code country_name continent   GDP life_expect population     CO2 battle_death child.per.woman
27   chn        China      Asia 14369       76.69 1410000000 9710000        0.000            1.62
62   kor  South Korea      Asia 35020       82.52   51000000  618000        0.113            1.30
   programmable.aid BTS_country_d BTS_country
27         1072.494          <NA>        <NA>
62          850.567      BTS_Home    BTS_Home
```

< Question 4 – country data>

Replace NA value of 'BTS_country' to 'others' as shown below.

Expected Result

```
> country$BTS_country
  [1] "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"
 [10] "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"
 [19] "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"
 [28] "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"
 [37] "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"
 [46] "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"
 [55] "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "BTS_Home" "Others"
 [64] "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"
 [73] "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"
 [82] "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"
 [91] "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"
[100] "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"
[109] "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"
[118] "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"   "Others"
```

< Question 5 – country data>

Use 'vectorized operation' to create a new column called 'GDP_dummy'. 'GDP_dummy' is 'Low' if GDP is smaller than the average GDP, and 'high' if GDP is greater than average GDP. (1) Show the first 2 rows and (2) the number of cases.

Expected Result

(1)

```
  code country_name continent   GDP life_expect population  CO2 battle_death child.per.woman
1  afg  Afghanistan      Asia  1757       61.22   35400000 8660         9.45            4.64
2  alb      Albania    Europe 11357       78.12    2890000 4540         0.13            1.71
  programmable.aid BTS_country_d BTS_country GDP_dummy
1        3663.2516          <NA>      Others       Low
2         277.1891          <NA>      Others       Low
```

(2)

```
High  Low
  48   78
```

< Question 6 – country data>

(1) Create a new list called 'country_gdp' by using for statement. Here, each list element should contain different "GDP_dummy" value as shown below. Also write down a R code that returns first 6 rows of each list element. (2) Write down a R code that count the number of samples for each list element.

Expected Result

(1)

```
[[1]]
  code country_name     continent   GDP life_expect population    CO2 battle_death child.per.woman
1  afg  Afghanistan          Asia  1757       61.22   35400000   8660       9.4500            4.64
2  alb      Albania        Europe 11357       78.12    2890000   4540       0.1300            1.71
3  dza      Algeria        Africa 13940       77.40   40600000 148000       3.4100            2.78
4  arg    Argentina South America 18645       76.54   43500000 200000       0.0000            2.29
5  arm      Armenia          Asia  8159       75.37    2940000   5180       0.0000            1.63
8  aze   Azerbaijan          Asia 16132       70.62    9740000  37200       0.0726            2.08
  programmable.aid BTS_country_d BTS_country GDP_dummy
1       3663.25163          <NA>      Others       Low
2        277.18911          <NA>      Others       Low
3        108.27441          <NA>      Others       Low
4         59.06856          <NA>      Others       Low
5        373.09101          <NA>      Others       Low
8        182.79669          <NA>      Others       Low

[[2]]
   code country_name     continent   GDP life_expect population    CO2 battle_death child.per.woman
6   aus    Australia       Oceania 44606       82.50   24300000 413000       0.0000            1.85
7   aut      Austria        Europe 44671       81.69    8750000  67400       0.0000            1.49
9   bhr      Bahrain          Asia 43732       79.42    1430000  31500       0.0000            2.03
12  bel      Belgium        Europe 42214       81.24   11400000  98500       0.0000            1.79
17  brn       Brunei          Asia 72370       75.23     420000   7550       0.0000            1.87
22  can       Canada North America 43089       81.87   36400000 565000       0.0035            1.58
   programmable.aid BTS_country_d BTS_country GDP_dummy
6           850.567          <NA>      Others      High
7           850.567          <NA>      Others      High
9           850.567          <NA>      Others      High
12          850.567          <NA>      Others      High
17          850.567          <NA>      Others      High
22          850.567          <NA>      Others      High
```

(2)

```
[[1]]
[1] 78

[[2]]
[1] 48
```

< Question 7 – country data>

Imagine you want to find out how many countries have GDP above 30,000 and compare the number of those countries by continents. Using tapply and sapply, write down a R code that counts the number of countries with GDP above 30,000.

Expected Result

```
      Africa         Asia       Europe North America       Oceania South America
           0           10           17             2             2             0
```

< Question 8 – country data>

(1) Create a DataFrame called “country.data.type” as shown below. In this table, ‘data.type’ variable contains the information about the data type of variable. (2) Create a new variable called "numeric.dummy”, which tells us whether the variable is numeric or not as shown below.

Expected Result

(1)

```
> country.data.type
                 data.type
code             character
country_name     character
continent        character
GDP                integer
life_expect         double
population         integer
CO2                 double
battle_death        double
child.per.woman     double
programmable.aid    double
BTS_country_d    character
BTS_country      character
GDP_dummy        character
```

(2)

```
> country.data.type
                 data.type numeric.dummy
code             character   non-numeric
country_name     character   non-numeric
continent        character   non-numeric
GDP                integer       numeric
life_expect         double       numeric
population         integer       numeric
CO2                 double       numeric
battle_death        double       numeric
child.per.woman     double       numeric
programmable.aid    double       numeric
BTS_country_d    character   non-numeric
BTS_country      character   non-numeric
GDP_dummy        character   non-numeric
```

< Question 9 – country data>

(1) Create a function called “AvgPop”, which receives the name of continent and returns the average population of it. (2) Create a function called "AvgPopList", which receives the name of continent and returns the list of countries with GDP above average. Below is the example.

Expected Result

(1)

```
> AvgPop('Asia')
[1] 124441471
```

(2)

```
> AvgPopList('Asia')
 [1] "Bahrain"      "Brunei"       "Cyprus"       "Israel"
 [5] "Japan"        "Kazakhstan"   "South Korea"  "Kuwait"
 [9] "Malaysia"     "Oman"         "Saudi Arabia" "Singapore"
[13] "Turkey"
```

< Question 10 – country data>

Use tapply to create a Data.Frame called “gdp.pop.df” that compares average GDP between countries with population below average (avg.gdp.low.pop) and ones with population above average (avg.gdp.high.pop).

Expected Result

```
> gdp.pop.df
  avg.gdp.low.pop avg.gdp.high.pop
1         20291.8         19241.22
```