q1.R

ASUS

2023-12-04

```r
#part 01

#read data into a variable
library(readr)
stu_performance <- read_csv(file = "E:/22Second Sem/Laboratory 2/R/practical - 02/archive/Student Performance new.csv")
```

```
## New names:
## Rows: 1000 Columns: 9
## ── Column specification
## ──────────────────────────────────────────────────────── Delimiter: "," chr
## (5): race/ethnicity, parental level of education, lunch, test preparatio... dbl
## (4): ...1, math percentage, reading score percentage, writing score perc...
## ℹ Use `spec()` to retrieve the full column specification for this data. ℹ
## Specify the column types or set `show_col_types = FALSE` to quiet this message.
## • `` -> `...1`
```

```r
stu_performance
```

```
## # A tibble: 1,000 × 9
##     ...1 `race/ethnicity` parental level of educa…¹ lunch test preparation cou…²
##    <dbl> <chr>            <chr>                     <chr> <chr>
##  1     0 group B          bachelor's degree         stan… none
##  2     1 group C          some college              stan… completed
##  3     2 group B          master's degree           stan… none
##  4     3 group A          associate's degree        free… none
##  5     4 group C          some college              stan… none
##  6     5 group B          associate's degree        stan… none
##  7     6 group B          some college              stan… completed
##  8     7 group B          some college              free… none
##  9     8 group D          high school               free… completed
## 10     9 group B          high school               free… none
## # ℹ 990 more rows
## # ℹ abbreviated names: ¹`parental level of education`,
## #   ²`test preparation course`
## # ℹ 4 more variables: `math percentage` <dbl>,
## #   `reading score percentage` <dbl>, `writing score percentage` <dbl>,
## #   sex <chr>
```

```r
# assigning new names to the columns of the data frame
colnames(stu_performance) <- c('id','race','edu', 'lunch', 'prep', 'math', 'read', 'write', 'gen')
stu_performance
```

```
## # A tibble: 1,000 × 9
##       id race    edu                lunch        prep     math  read write gen
##    <dbl> <chr>   <chr>              <chr>        <chr>   <dbl> <dbl> <dbl> <chr>
##  1     0 group B bachelor's degree  standard     none     0.72  0.72  0.74 F
##  2     1 group C some college       standard     comple…  0.69  0.9   0.88 F
##  3     2 group B master's degree    standard     none     0.9   0.95  0.93 F
##  4     3 group A associate's degree free/reduced none     0.47  0.57  0.44 M
##  5     4 group C some college       standard     none     0.76  0.78  0.75 M
##  6     5 group B associate's degree standard     none     0.71  0.83  0.78 F
##  7     6 group B some college       standard     comple…  0.88  0.95  0.92 F
##  8     7 group B some college       free/reduced none     0.4   0.43  0.39 M
##  9     8 group D high school        free/reduced comple…  0.64  0.64  0.67 M
## 10     9 group B high school        free/reduced none     0.38  0.6   0.5  F
## # ℹ 990 more rows
```

```r
#Change the math, read and write variables to whole number (Multiply by 100)
stu_performance[c('math')] <- stu_performance[c('math')] * 100
stu_performance[c('read')] <- stu_performance[c('read')] * 100
stu_performance[c('write')] <- stu_performance[c('write')] * 100
stu_performance
```

```
## # A tibble: 1,000 × 9
##       id race    edu                lunch        prep     math  read write gen
##    <dbl> <chr>   <chr>              <chr>        <chr>   <dbl> <dbl> <dbl> <chr>
##  1     0 group B bachelor's degree  standard     none       72    72    74 F
##  2     1 group C some college       standard     comple…    69    90    88 F
##  3     2 group B master's degree    standard     none       90    95    93 F
##  4     3 group A associate's degree free/reduced none       47    57    44 M
##  5     4 group C some college       standard     none       76    78    75 M
##  6     5 group B associate's degree standard     none       71    83    78 F
##  7     6 group B some college       standard     comple…    88    95    92 F
##  8     7 group B some college       free/reduced none       40    43    39 M
##  9     8 group D high school        free/reduced comple…    64    64    67 M
## 10     9 group B high school        free/reduced none       38    60    50 F
## # ℹ 990 more rows
```

```r
#Create a new attribute average (average of math,read and write)
attr(stu_performance, "average of math") <- mean(stu_performance$math)
attr(stu_performance, "average of read") <- mean(stu_performance$read)
attr(stu_performance, "average of write") <- mean(stu_performance$write)
attributes(stu_performance)
```

```
## $row.names
##    [1]    1    2    3    4    5    6    7    8    9   10   11   12   13   14
##   [15]   15   16   17   18   19   20   21   22   23   24   25   26   27   28
##   [29]   29   30   31   32   33   34   35   36   37   38   39   40   41   42
##   [43]   43   44   45   46   47   48   49   50   51   52   53   54   55   56
##   [57]   57   58   59   60   61   62   63   64   65   66   67   68   69   70
##   [71]   71   72   73   74   75   76   77   78   79   80   81   82   83   84
##   [85]   85   86   87   88   89   90   91   92   93   94   95   96   97   98
##   [99]   99  100  101  102  103  104  105  106  107  108  109  110  111  112
##  [113]  113  114  115  116  117  118  119  120  121  122  123  124  125  126
##  [127]  127  128  129  130  131  132  133  134  135  136  137  138  139  140
##  [141]  141  142  143  144  145  146  147  148  149  150  151  152  153  154
##  [155]  155  156  157  158  159  160  161  162  163  164  165  166  167  168
##  [169]  169  170  171  172  173  174  175  176  177  178  179  180  181  182
##  [183]  183  184  185  186  187  188  189  190  191  192  193  194  195  196
##  [197]  197  198  199  200  201  202  203  204  205  206  207  208  209  210
##  [211]  211  212  213  214  215  216  217  218  219  220  221  222  223  224
##  [225]  225  226  227  228  229  230  231  232  233  234  235  236  237  238
##  [239]  239  240  241  242  243  244  245  246  247  248  249  250  251  252
##  [253]  253  254  255  256  257  258  259  260  261  262  263  264  265  266
##  [267]  267  268  269  270  271  272  273  274  275  276  277  278  279  280
##  [281]  281  282  283  284  285  286  287  288  289  290  291  292  293  294
##  [295]  295  296  297  298  299  300  301  302  303  304  305  306  307  308
##  [309]  309  310  311  312  313  314  315  316  317  318  319  320  321  322
##  [323]  323  324  325  326  327  328  329  330  331  332  333  334  335  336
##  [337]  337  338  339  340  341  342  343  344  345  346  347  348  349  350
##  [351]  351  352  353  354  355  356  357  358  359  360  361  362  363  364
##  [365]  365  366  367  368  369  370  371  372  373  374  375  376  377  378
##  [379]  379  380  381  382  383  384  385  386  387  388  389  390  391  392
##  [393]  393  394  395  396  397  398  399  400  401  402  403  404  405  406
##  [407]  407  408  409  410  411  412  413  414  415  416  417  418  419  420
##  [421]  421  422  423  424  425  426  427  428  429  430  431  432  433  434
##  [435]  435  436  437  438  439  440  441  442  443  444  445  446  447  448
##  [449]  449  450  451  452  453  454  455  456  457  458  459  460  461  462
##  [463]  463  464  465  466  467  468  469  470  471  472  473  474  475  476
##  [477]  477  478  479  480  481  482  483  484  485  486  487  488  489  490
##  [491]  491  492  493  494  495  496  497  498  499  500  501  502  503  504
##  [505]  505  506  507  508  509  510  511  512  513  514  515  516  517  518
##  [519]  519  520  521  522  523  524  525  526  527  528  529  530  531  532
##  [533]  533  534  535  536  537  538  539  540  541  542  543  544  545  546
##  [547]  547  548  549  550  551  552  553  554  555  556  557  558  559  560
##  [561]  561  562  563  564  565  566  567  568  569  570  571  572  573  574
##  [575]  575  576  577  578  579  580  581  582  583  584  585  586  587  588
##  [589]  589  590  591  592  593  594  595  596  597  598  599  600  601  602
##  [603]  603  604  605  606  607  608  609  610  611  612  613  614  615  616
##  [617]  617  618  619  620  621  622  623  624  625  626  627  628  629  630
##  [631]  631  632  633  634  635  636  637  638  639  640  641  642  643  644
##  [645]  645  646  647  648  649  650  651  652  653  654  655  656  657  658
##  [659]  659  660  661  662  663  664  665  666  667  668  669  670  671  672
##  [673]  673  674  675  676  677  678  679  680  681  682  683  684  685  686
##  [687]  687  688  689  690  691  692  693  694  695  696  697  698  699  700
##  [701]  701  702  703  704  705  706  707  708  709  710  711  712  713  714
##  [715]  715  716  717  718  719  720  721  722  723  724  725  726  727  728
##  [729]  729  730  731  732  733  734  735  736  737  738  739  740  741  742
##  [743]  743  744  745  746  747  748  749  750  751  752  753  754  755  756
##  [757]  757  758  759  760  761  762  763  764  765  766  767  768  769  770
##  [771]  771  772  773  774  775  776  777  778  779  780  781  782  783  784
##  [785]  785  786  787  788  789  790  791  792  793  794  795  796  797  798
##  [799]  799  800  801  802  803  804  805  806  807  808  809  810  811  812
##  [813]  813  814  815  816  817  818  819  820  821  822  823  824  825  826
##  [827]  827  828  829  830  831  832  833  834  835  836  837  838  839  840
##  [841]  841  842  843  844  845  846  847  848  849  850  851  852  853  854
##  [855]  855  856  857  858  859  860  861  862  863  864  865  866  867  868
##  [869]  869  870  871  872  873  874  875  876  877  878  879  880  881  882
##  [883]  883  884  885  886  887  888  889  890  891  892  893  894  895  896
##  [897]  897  898  899  900  901  902  903  904  905  906  907  908  909  910
##  [911]  911  912  913  914  915  916  917  918  919  920  921  922  923  924
##  [925]  925  926  927  928  929  930  931  932  933  934  935  936  937  938
##  [939]  939  940  941  942  943  944  945  946  947  948  949  950  951  952
##  [953]  953  954  955  956  957  958  959  960  961  962  963  964  965  966
##  [967]  967  968  969  970  971  972  973  974  975  976  977  978  979  980
##  [981]  981  982  983  984  985  986  987  988  989  990  991  992  993  994
##  [995]  995  996  997  998  999 1000
##
## $names
## [1] "id"    "race"  "edu"   "lunch" "prep"  "math"  "read"  "write" "gen"
##
## $spec
## cols(
##   ...1 = col_double(),
##   `race/ethnicity` = col_character(),
##   `parental level of education` = col_character(),
##   lunch = col_character(),
##   `test preparation course` = col_character(),
##   `math percentage` = col_double(),
##   `reading score percentage` = col_double(),
##   `writing score percentage` = col_double(),
##   sex = col_character()
## )
##
## $problems
## <pointer: 0x000001fa0d219100>
##
## $class
## [1] "spec_tbl_df" "tbl_df"      "tbl"         "data.frame"
##
## $`average of math`
## [1] 66.089
##
## $`average of read`
## [1] 69.169
##
## $`average of write`
## [1] 68.054
```

```r
#Find the summary for each mark (math,read and write)
math_sumary = summary(stu_performance$math)
math_sumary
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00   57.00   66.00   66.09   77.00  100.00
```

```r
read_sumary = summary(stu_performance$read)
read_sumary
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   17.00   59.00   70.00   69.17   79.00  100.00
```

```r
write_sumary = summary(stu_performance$write)
write_sumary
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   10.00   57.75   69.00   68.05   79.00  100.00
```

```r
#Find unique values for attributes a.
#a. race
race_unique = unique(stu_performance$race)
race_unique
```

```
## [1] "group B" "group C" "group A" "group D" "group E"
```

```r
#b. edu
edu_unique = unique(stu_performance$edu)
edu_unique
```

```
## [1] "bachelor's degree"  "some college"       "master's degree"
## [4] "associate's degree" "high school"        "some high school"
```

```r
#c. lunch
lunch_unique = unique(stu_performance$lunch)
lunch_unique
```

```
## [1] "standard"     "free/reduced"
```

```r
#d. prep
prep_unique = unique(stu_performance$prep)
prep_unique
```

```
## [1] "none"      "completed"
```

```r
#e. gen
gen_unique = unique(stu_performance$gen)
gen_unique
```

```
## [1] "F" "M"
```

```r
#8. Find average mark for
#a. race
avg_math_by_race <- aggregate(math ~ race, data = stu_performance, FUN = mean)
avg_read_by_race <- aggregate(read ~ race, data = stu_performance, FUN = mean)
avg_write_by_race <- aggregate(write ~ race, data = stu_performance, FUN = mean)

print(avg_math_by_race)
```

```
##      race     math
## 1 group A 61.62921
## 2 group B 63.45263
## 3 group C 64.46395
## 4 group D 67.36260
## 5 group E 73.82143
```

```r
print(avg_read_by_race)
```

```
##      race     read
## 1 group A 64.67416
## 2 group B 67.35263
## 3 group C 69.10345
## 4 group D 70.03053
## 5 group E 73.02857
```

```r
print(avg_write_by_race)
```

```
##      race    write
## 1 group A 62.67416
## 2 group B 65.60000
## 3 group C 67.82759
## 4 group D 70.14504
## 5 group E 71.40714
```

```r
#b. edu
avg_math_by_edu <- aggregate(math ~ edu, data = stu_performance, FUN = mean)
avg_read_by_edu <- aggregate(read ~ edu, data = stu_performance, FUN = mean)
avg_write_by_edu <- aggregate(write ~ edu, data = stu_performance, FUN = mean)

print(avg_math_by_edu)
```

```
##                  edu     math
## 1 associate's degree 67.88288
## 2  bachelor's degree 69.38983
## 3        high school 62.13776
## 4    master's degree 69.74576
## 5       some college 67.12832
## 6   some high school 63.49721
```

```r
print(avg_read_by_edu)
```

```
##                  edu     read
## 1 associate's degree 70.92793
## 2  bachelor's degree 73.00000
## 3        high school 64.70408
## 4    master's degree 75.37288
## 5       some college 69.46018
## 6   some high school 66.93855
```

```r
print(avg_write_by_edu)
```

```
##                  edu    write
## 1 associate's degree 69.89640
## 2  bachelor's degree 73.38136
## 3        high school 62.44898
## 4    master's degree 75.67797
## 5       some college 68.84071
## 6   some high school 64.88827
```

```r
#c. lunch
avg_math_by_lunch <- aggregate(math ~ lunch, data = stu_performance, FUN = mean)
avg_read_by_lunch <- aggregate(read ~ lunch, data = stu_performance, FUN = mean)
avg_write_by_lunch <- aggregate(write ~ lunch, data = stu_performance, FUN = mean)

print(avg_math_by_lunch)
```

```
##          lunch     math
## 1 free/reduced 58.92113
## 2     standard 70.03411
```

```r
print(avg_read_by_lunch)
```

```
##          lunch     read
## 1 free/reduced 64.65352
## 2     standard 71.65426
```

```r
print(avg_write_by_lunch)
```

```
##          lunch    write
## 1 free/reduced 63.02254
## 2     standard 70.82326
```

```r
#d. prep
avg_math_by_prep <- aggregate(math ~ prep, data = stu_performance, FUN = mean)
avg_read_by_prep <- aggregate(read ~ prep, data = stu_performance, FUN = mean)
avg_write_by_prep <- aggregate(write ~ prep, data = stu_performance, FUN = mean)

print(avg_math_by_prep)
```

```
##        prep     math
## 1 completed 69.69553
## 2      none 64.07788
```

```r
print(avg_read_by_prep)
```

```
##        prep     read
## 1 completed 73.89385
## 2      none 66.53427
```

```r
print(avg_write_by_prep)
```

```
##        prep    write
## 1 completed 74.41899
## 2      none 64.50467
```

```r
#e. Gen
avg_math_by_gen <- aggregate(math ~ gen, data = stu_performance, FUN = mean)
avg_read_by_gen <- aggregate(read ~ gen, data = stu_performance, FUN = mean)
avg_write_by_gen <- aggregate(write ~ gen, data = stu_performance, FUN = mean)

print(avg_math_by_gen)
```

```
##   gen     math
## 1   F 63.63320
## 2   M 68.72822
```

```r
print(avg_read_by_gen)
```

```
##   gen     read
## 1   F 72.60811
## 2   M 65.47303
```

```r
print(avg_write_by_gen)
```

```
##   gen    write
## 1   F 72.46718
## 2   M 63.31120
```

```r
#part 02
Melbourne_Housing_Snapshot <- read_csv(file = "E:/22Second Sem/Laboratory 2/R/practical - 02/archive- 2/melb_data.csv")
```

```
## Rows: 13580 Columns: 21
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr  (8): Suburb, Address, Type, Method, SellerG, Date, CouncilArea, Regionname
## dbl (13): Rooms, Price, Distance, Postcode, Bedroom2, Bathroom, Car, Landsiz...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
Melbourne_Housing_Snapshot
```

```
## # A tibble: 13,580 × 21
##    Suburb     Address  Rooms Type   Price Method SellerG Date  Distance Postcode
##    <chr>      <chr>    <dbl> <chr>  <dbl> <chr>  <chr>   <chr>    <dbl>    <dbl>
##  1 Abbotsford 85 Turn…     2 h     1.48e6 S      Biggin  3/12…      2.5     3067
##  2 Abbotsford 25 Bloo…     2 h     1.03e6 S      Biggin  4/02…      2.5     3067
##  3 Abbotsford 5 Charl…     3 h     1.46e6 SP     Biggin  4/03…      2.5     3067
##  4 Abbotsford 40 Fede…     3 h     8.5 e5 PI     Biggin  4/03…      2.5     3067
##  5 Abbotsford 55a Par…     4 h     1.6 e6 VB     Nelson  4/06…      2.5     3067
##  6 Abbotsford 129 Cha…     2 h     9.41e5 S      Jellis  7/05…      2.5     3067
##  7 Abbotsford 124 Yar…     3 h     1.88e6 S      Nelson  7/05…      2.5     3067
##  8 Abbotsford 98 Char…     2 h     1.64e6 S      Nelson  8/10…      2.5     3067
##  9 Abbotsford 6/241 N…     1 u     3   e5 S      Biggin  8/10…      2.5     3067
## 10 Abbotsford 10 Vali…     2 h     1.10e6 S      Biggin  8/10…      2.5     3067
## # ℹ 13,570 more rows
## # ℹ 11 more variables: Bedroom2 <dbl>, Bathroom <dbl>, Car <dbl>,
## #   Landsize <dbl>, BuildingArea <dbl>, YearBuilt <dbl>, CouncilArea <chr>,
## #   Lattitude <dbl>, Longtitude <dbl>, Regionname <chr>, Propertycount <dbl>
```

```r
#Print first few values of the dataset
print(head(Melbourne_Housing_Snapshot))
```

```
## # A tibble: 6 × 21
##   Suburb     Address   Rooms Type   Price Method SellerG Date  Distance Postcode
##   <chr>      <chr>     <dbl> <chr>  <dbl> <chr>  <chr>   <chr>    <dbl>    <dbl>
## 1 Abbotsford 85 Turne…     2 h     1.48e6 S      Biggin  3/12…      2.5     3067
## 2 Abbotsford 25 Bloom…     2 h     1.03e6 S      Biggin  4/02…      2.5     3067
## 3 Abbotsford 5 Charle…     3 h     1.46e6 SP     Biggin  4/03…      2.5     3067
## 4 Abbotsford 40 Feder…     3 h     8.5 e5 PI     Biggin  4/03…      2.5     3067
## 5 Abbotsford 55a Park…     4 h     1.6 e6 VB     Nelson  4/06…      2.5     3067
## 6 Abbotsford 129 Char…     2 h     9.41e5 S      Jellis  7/05…      2.5     3067
## # ℹ 11 more variables: Bedroom2 <dbl>, Bathroom <dbl>, Car <dbl>,
## #   Landsize <dbl>, BuildingArea <dbl>, YearBuilt <dbl>, CouncilArea <chr>,
## #   Lattitude <dbl>, Longtitude <dbl>, Regionname <chr>, Propertycount <dbl>
```

```r
# Count the number of missing values in each attribute
missing_values <- colSums(is.na(Melbourne_Housing_Snapshot))
print(missing_values)
```

```
##        Suburb       Address         Rooms          Type         Price
##             0             0             0             0             0
##        Method       SellerG          Date      Distance      Postcode
##             0             0             0             0             0
##      Bedroom2      Bathroom           Car      Landsize  BuildingArea
##             0             0            62             0          6450
##     YearBuilt   CouncilArea     Lattitude    Longtitude    Regionname
##          5375          1369             0             0             0
## Propertycount
##             0
```

```r
#Find the mean value for “YearBuilt”
meanYearBuilt <- mean(Melbourne_Housing_Snapshot$YearBuilt, na.rm = TRUE)
print(meanYearBuilt)
```

```
## [1] 1964.684
```

q2.R

ASUS

2023-12-04

```
# Create the dataframe
emp_sal <- data.frame(
  Emp_ID = c(11, 12, 13, 14, 15),
  Dep = c("Sales", "HR", "Sales", "HR", "Sales"),
  Basic = c(25450, 22500, 21000, 23500, 15000),
  Allowances = c(5200, 4500, 3100, 2600, 1800)
)

print(emp_sal)
```

```
##   Emp_ID   Dep Basic Allowances
## 1     11 Sales 25450       5200
## 2     12    HR 22500       4500
## 3     13 Sales 21000       3100
## 4     14    HR 23500       2600
## 5     15 Sales 15000       1800
```

```
#Store net salary in new column named “net_sal”
emp_sal$net_salary <- emp_sal$Basic + emp_sal$Allowances
print(emp_sal$net_salary)
```

```
## [1] 30650 27000 24100 26100 16800
```

```
#Obtain employee IDs of employees whose net salary is above 25000
high_salary_employees <- emp_sal$Emp_ID[emp_sal$net_sal > 25000]
print(high_salary_employees)
```

```
## [1] 11 12 14
```

```
#Obtain employee IDs of employees attached to HR Department whose net salary is below 25000
HR_high_salary_employees <- emp_sal$Emp_ID[emp_sal$net_sal > 25000 & emp_sal$Dep == 'HR']
print(HR_high_salary_employees)
```

```
## [1] 12 14
```