1. Create the following matrix of 5 rows

 [,1] [,2] [,3] [,4]

[1,]   6  112    5  545

[2,]  34  113    9  546

[3,] 923  114   34  547

[4,]   5  115   76  548

[5,]   0  116    2  549

**Function ‘apply’ on a simple matrix:**

**a.** Get the following matrix of 5 rows and call it ‘mymatrix’

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9 | mymatrix = matrix(data = c(6,34,923,5,0, 112:116, 5,9,34,76,2, 545:549), nrow = 5)  mymatrix        [,1] [,2] [,3] [,4]  [1,]   6  112    5  545  [2,]  34  113    9  546  [3,] 923  114   34  547  [4,]   5  115   76  548  [5,]   0  116    2  549 |

**b.** Get the mean of each row

|  |  |
| --- | --- |
| 1  2  3 | #expected result    [1] 167.00 175.50 404.50 186.00 166.75 |

**c.** Get the mean of each column

|  |  |
| --- | --- |
| 1  2  3 | #expected result    [1] 193.6 114.0 25.2 547.0 |

**d.** Sort the columns in ascending order

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | #expected result        [,1] [,2] [,3] [,4]  [1,]   0  112    2  545  [2,]   5  113    5  546  [3,]   6  114    9  547  [4,]  34  115   34  548  [5,] 923  116   76  549 |

**2. Using ‘lapply’ on a data.frame ‘mtcars’**

**a.** Use three ‘apply’ family functions to get the minimum values of each column of the ‘mtcars’ dataset (*hint:* ‘lapply’, ‘sapply’, ‘mapply’).

Store each output in a separate object (‘l’, ‘s’, ‘m’) and get the outputs.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43 | #expected result    >l  $mpg  [1] 10.4    $cyl  [1] 4    $disp  [1] 71.1    $hp  [1] 52    $drat  [1] 2.76    $wt  [1] 1.513    $qsec  [1] 14.5    $vs  [1] 0    $am  [1] 0    $gear  [1] 3    $carb  [1] 1    >s     mpg   cyl   disp     hp  drat    wt   qsec    vs    am  gear  carb  10.400 4.000 71.100 52.000 2.760 1.513 14.500 0.000 0.000 3.000 1.000    >m     mpg   cyl   disp     hp  drat    wt   qsec    vs    am  gear  carb  10.400 4.000 71.100 52.000 2.760 1.513 14.500 0.000 0.000 3.000 1.000 |

**b.** Put the three outputs ‘l’, ‘s’, ‘m’ in the list ‘listobjects’

**c.** Use a suitable ‘apply’ function to get the class of each of the three list elements in ‘listobjects’

**d.** Name the output classes for each of the three functions used in the exercise

**3. ‘mapply’**

**a.** Use ‘mapply’ to get a list of 10 elements. The list is an alteration of ‘A’ and ‘F’. The lengths of those 10 alternating elements decreases step by step from 10 to 1.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31 | #expected result    $A   [1] "A" "A" "A" "A" "A" "A" "A" "A" "A" "A"    $F  [1] "F" "F" "F" "F" "F" "F" "F" "F" "F"    $<NA>  [1] "A" "A" "A" "A" "A" "A" "A" "A"    $<NA>  [1] "F" "F" "F" "F" "F" "F" "F"    $<NA>  [1] "A" "A" "A" "A" "A" "A"    $<NA>  [1] "F" "F" "F" "F" "F"    $<NA>  [1] "A" "A" "A" "A"    $<NA>  [1] "F" "F" "F"    $<NA>  [1] "A" "A"    $<NA>  [1] "F" |

**b.** Tweak the function that you get proper element numbers (1 : 10) for the 10 list elements. *Hint:* argument USE.NAMES

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31 | #expected result    [[1]]   [1] "A" "A" "A" "A" "A" "A" "A" "A" "A" "A"    [[2]]  [1] "F" "F" "F" "F" "F" "F" "F" "F" "F"    [[3]]  [1] "A" "A" "A" "A" "A" "A" "A" "A"    [[4]]  [1] "F" "F" "F" "F" "F" "F" "F"    [[5]]  [1] "A" "A" "A" "A" "A" "A"    [[6]]  [1] "F" "F" "F" "F" "F"    [[7]]  [1] "A" "A" "A" "A"    [[8]]  [1] "F" "F" "F"    [[9]]  [1] "A" "A"    [[10]]  [1] "F" |

[Reveal solution](http://r-tutorials.com/r-exercises-apply-family-functions/#ac_441_collapse1)

**4. Titanic Casualties – Use the standard ‘Titanic’ dataset which is part of R Base**

**a.** Use an appropriate apply function to get the sum of males vs females aboard.

|  |  |
| --- | --- |
| 1  2  3  4 | #expected result    Male Female   1731 470 |

**b.** Get a table with the sum of survivors vs sex.

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | #expected result           Survived   Sex     No Yes   Male  1364 367   Female 126 344 |

**c.** Get a table with the sum of passengers by sex vs age.

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | #expected result               Sex     Age  Male Female   Child    64     45   Adult  1667    425 |

**5. Extracting elements from a list of matrices with ‘lapply’**

**a.** Create ‘listobj’ which is a list of four matrices – *see data*:

|  |  |
| --- | --- |
| 1  2  3  4  5 | first = matrix(38:66, 3)  second = matrix(56:91, 3)  third = matrix(82:145, 3)  fourth = matrix(46:93, 5)  listobj = list(first, second, third, fourth) |

**b.** Extract the second column from the list of matrices (from each single matrix).

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | #expected result    [[1]]  [1] 41 42 43    [[2]]  [1] 59 60 61    [[3]]  [1] 85 86 87    [[4]]  [1] 51 52 53 54 55 |

**c.** Extract the third row from the list of matrices.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | #expected result    [[1]]   [1] 40 43 46 49 52 55 58 61 64 38    [[2]]   [1] 58 61 64 67 70 73 76 79 82 85 88 91    [[3]]   [1] 84 87 90 93 96 99 102 105 108 111 114 117 120 123 126 129 132 135 138 141 144 83    [[4]]   [1] 48 53 58 63 68 73 78 83 88 93 |

<http://r-tutorials.com/r-exercises-apply-family-functions/>

<https://www.r-exercises.com/2017/01/17/multivariate-apply-exercises/>