|  |
| --- |
| R Test + Business Case |

Question 1

What type of object(s) should be used to store a table that contains categorical and numeric values?

1. Vector
2. Factor
3. Matrix
4. Data Frame
5. List
6. Only IV
7. I and II
8. III, IV and V
9. IV and V
10. Only V

Question 2

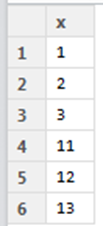
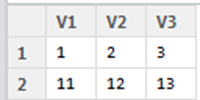
Which function from the “apply family” should be used to calculate the mean for each of the columns in a table?

1. Apply
2. Sapply
3. Lapply
4. Mapply
5. Tapply

Question 3

How would you transform Table 1 into Table 2? (You may pick more than 1 alternative)

Table 1 Table 2



1. c(1,2,3,11,12,13)
2. as.matrix(c(1,2,3, 11,12,13))
3. matrix(c(1,2,3, 11,12,13), nrow = 2, ncol = 3, byrow = TRUE)
4. matrix(c(1,2,3, 11,12,13), nrow = 2, ncol = 3, byrow = FALSE)
5. rbind(c(1,2,3),c(11,12,13))

Question 4

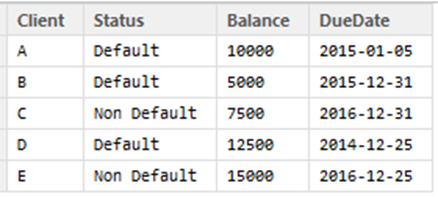
Create a function that returns (x^2 + 2\*x) for every entry of the object which the function is being applied on.

fx = function(x){(x^2 + 2\*x)}

Question 5

Given the Dataset 2, write the commands for the required subsets.

Dataset 2



5.1

Clients which have not fallen into default.

subset(Dataset.2, Status != ‘Default’, select = Client)

5.2

Clients and Balances with Default Status and Balance higher than 10000.

subset(Dataset.2, Status == ‘Default’ & Balance > 10000, select = c(Client, Balance))

5.3

 Due date of clients who have Non Default Status and Balance higher than 10000.

subset(Dataset.2, Status == “Non Default” & Balance > 10000, select = DueDate)

Question 6

Given the following R Code. Why the output table does not correspond with the code?

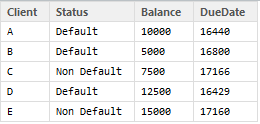
**Client<-c("A","B","C","D","E")**

**Status<-c("Default","Default","Non Default","Default","Non Default")**

**Balance<-c(10000,5000,7500,12500,15000)**

**DueDate<-as.Date(c("01-05-2015","12-31-2015","12-31-2016","12-25-2014","12-25-2016"),"%m-%d-%Y")**

**Total<-cbind(Client,Status,Balance,DueDate,deparse.level=0)**

****

1. The format given to DueDate field is not correct
2. The output is a matrix, so it cannot contain dates and characters
3. Deparse.level option should be 1 instead of 0.
4. The output is correct. It is just that the visualization retrieves wrong dates, but once you calculate something will work properly.
5. None of the above

Question 7

A t-test is applied using the following code.

**treeVolume <- c(rnorm(7500, mean = 39500, sd = 3000))**

**t.test(treeVolume, mu = 38500,alternative="greater")**

Interpret the information retrieved by the command t.test described below.

data: treeVolume

t = 27.6918, df = 7499, p-value < 2.2e-16

alternative hypothesis: true mean is greater than 38500

95 percent confidence interval:

39415.68 Inf

sample estimates:

mean of x

39473.51

At 95% confidence level, there is significant difference (p-value < 0,05) of the two means. Accept the hyphothesis that new means is greater than the old means. The new mean its greater than 39415.68.

Question 8

Which of the following sentences are correct about R?

1. R cannot show more than 1,000 rows in a View
2. R is case-sensitive
3. R cannot import data from Excel
4. R cannot use external packages and has low connectivity with other software.
5. Only I
6. Only II
7. I and II
8. II and IV
9. I, II and IV

Business Case

For this part you will need the CSV files that were shared with you as part of the test.

Part 1

You are an analyst hired by the Risk department of AdventureWorks Corp. This department has detected suspicious behaviour in the transactions of some Sales Orders. That is why they need you to create a unique R function with the following characteristics:

The only input of the function will be an **OrderID** that would be provided by them.

The function needs to be able to find a ‘Population of Interest’. This population corresponds to all the Sales carried out until 300 days after the order date of the given Sales Order ID.

Within the detected population, you need to be able to provide the following:

* Ranking of the most frequent Clients
* Dual axis chart with Number of Sales and Sale Amount (use bar for number of Sales and line for Sale amount)
* Ranking of Main Products sold

The function needs to be tested out by you and be ready to be used by the Risk folks.

Part 2

Marketing team wants to launch a campaign for most loyal customer so they need to identify top 5 clients whom spent more in a territory in a particular month. Marketing is not R proficient so they recruit you to create an R code to execute this task.

setwd("C:/Users/bkuasney/Desktop/ML/EVALUE")

salesOrder = read.csv2("SalesOrder.csv", sep=',')

str(salesOrder)

names(salesOrder)

territory = read.csv2("Territory.csv", sep=',')

str(territory)

names(territory)

require(sqldf)

salesOrder = sqldf('select \* from salesOrder left join territory using(TerritoryID)', drv = 'SQLite')

names(salesOrder)

# TRAT 1

salesOrder\_trat = salesOrder[,c(2,3,9,10,11,12)]

head(salesOrder\_trat)

distinct(salesOrder\_trat, Name)

names(salesOrder\_trat)

salesOrder\_trat = salesOrder\_trat[,1:4]

names(salesOrder\_trat)

# TRAT 2

salesOrder\_trat2 = sqldf('select OrderDate, CustomerID, sum(TotalDue) as total, Name from salesOrder\_trat group by OrderDate, CustomerID, Name', drv='SQLite')

str(salesOrder\_trat2)

library(lubridate)

salesOrder\_trat2$OrderDate = mdy(salesOrder\_trat2$OrderDate, locale="en\_US.UTF-8")

str(salesOrder\_trat2$OrderDate)

customer\_spend = function(date,territory) {

subset(salesOrder\_trat2[order(salesOrder\_trat2$total),], OrderDate == date & Name == territory)

}

customer\_spend("2006-02-02","Germany")

Part 2.1

It’s expected that the final user must have the ability to write only one line of code specifying one **Sales Territory Name** and Month-Year (**mm-yyyy**). The report for the Territory must include basic customer information (full name, contact information, etc.), the store where they purchase and total amount spent. The final output must be an xlsx file. (**Note**: Use ‘openxlsx’ package to export final output)

This question its basically the same question as above. Basically, we need to transform date into ‘mm-yyyy’ and use same technique to left join other dataset in csv, using de sqldf package, to bring the extra informations (full name, contact information, etc). Using, aggregate function and group by to calculate total amount spend.

Part 2.2

Create a new function that calls the previous function (no needs modification) and generate a full report for one year (including again top 5 clients for each month). The output should be a spreadsheet that contains one month per tab. (**Note**: Use ‘openxlsx’ package to export final output)

We call the saved function, apply a subset range with dates for one year. Each month contains 5 top clients. Use “for x in 1:12” for execute the code each month on year, and save each output in a distinct data frame. Next, we create another “for” with a lists of data frame to put it one data frame per tab in spreadsheet.