

```
##### FUNCAO SAMPLE.CHECK #####

#CHAMADA DA FUNÇÃO

sample.check <- function(file, RAZ1MIN = 1.8, RAZ1MAX = 2, RAZ2 = 1.5, CONC
= 15, VOL = 55){ #comando para executar a função e chamar os argumentos
RAZ1MIN, RAZ1MAX, RAZ2, CONC e VOL

#VERIFICANDO ARQUIVO DE ENTRADA
  if (is.data.frame(file) == FALSE){ # condicional para checagem da classe
de FILE
    stop("ERRO: arquivo de input fora do padrao, nao segue formato de
tabela, consulte o help da funcao")} # warning com parada quando o arquivo
de entrada não é classe data.frame
    if (ncol(file)!= 4){ # condicional para checagem de colunas
      stop("ERRO: arquivo de input fora do padrao, informacao faltante de
razoes ou concentracao, consulte o help da funcao")} # warning com parada
quando o arquivo de entrada não apresenta número de colunas adequado
      if (any(is.na(file)) == TRUE){ # condicional para checagem de ocorrência
de NA
        stop("ERRO: arquivo de input fora do padrao, informação ausente em
alguma amostra, consulte o help da funcao")} # warning com parada quando o
arquivo de entrada apresenta "NA"
        if (is.character(file[,2:4]) == TRUE) { # condicional para checagem de
ocorrência de characters indevidos
          stop("ERRO: arquivo de input fora do padrao, há letras onde deveria
haver apenas números nas colunas de razão e concentração")} # warning com
parada quando o arquivo de entrada apresenta letras entre a coluna 2 até a 4
do input.
#CRIANDO OUTPUT "FRAME"
      frame <- as.data.frame(matrix(NA, ncol = (ncol(file)+2), nrow =
(nrow(file)))) #criação do data.frame FRAME para receber os resultados da
função
      colnames(frame) <- c(" AMOSTRA", " 0D260/0D280", " 0D260/0D230", "
CONCENTRACAO", " VOL. AMOSTRA", " VOL. BUFFER") # atribuição de nomes as
colunas

#LENDO LINHAS DE "FILE" E CRIANDO OBJETOS
      for (j in 1:nrow(file)){ # loop para execução de tarefas a partir de todas
as linhas do arquivo de entrada FILE.
        amostra <- file[j,] # atribuição da primeira linha do arquivo FILE ao
objeto AMOSTRA
        frame[j,1] <- amostra[1] # atribuição da primeira posição do objeto
AMOSTRA a posição primeira posição na respectiva linha, copiando o nome da
amostra de FILE para FRAME.
        RAZ1MINi <- amostra[2] # atribuição da posição 2 de AMOSTRA
correspondente a j, ao objeto RAZ1MINi. Esse é o valor da razao 260/280
informado em FILE.
        PUR1V <- (RAZ1MINi >= RAZ1MIN) + (RAZ1MINi <= RAZ1MAX) # atribuição do
resultado booleano do produto da comparação do valor da razão 260/280 com os
```

mínimo (RAZ1MIN) e máximo (RAZ1MAX) ao objeto PUR1V.

CONCi <- amostra[4] # atribuição da posição 4 da AMOSTRA correspondente a j, ao objeto CONCi. Esse é o valor de concentração informado pelo usuário.

CONCV <- CONCi >= CONC # atribuição do resultado booleano da comparação entre o valor da concentração CONCi e o o valor referência de concentração CONC, ao objeto CONCV.

RAZ2i <- amostra[3] # atribuição da posição 3 de AMOSTRA correspondente a j, ao objeto RAZ2i. Esse é o valor da razão 260/230 informado em FILE pelo usuário.

RAZ2V <- RAZ2i >= RAZ2 # atribuição do resultado booleano da comparação entre o valor da razão 260/230 com o parâmetro RAZ2 determinado pelo usuário.

APLICANDO CONDICIONAIS PARA MONTAR "FRAME"

if (PUR1V == "2"){ # checagem condicional do valor booleano do objeto PUR1V, se o valor da razão 260/280 (RAZ1MINi) estiver acima do mínimo (RAZ1MIN) e abaixo do máximo (RAZ1MAX) o resultado será 2 (TRUE + TRUE).

frame[j,2] <- paste("adequada") } # Caso a condição do if seja cumprida, atribua a coluna 2 da linha j de FRAME "adequada", atestando que a razão da amostra foi checada e passou no parâmetro.

else {frame[j,2] <- paste("inadequada")} # Caso a condição do if não seja cumprida, atribua a coluna 2 da linha j de FRAME "inadequada", atestando que a razão da amostra foi checada e não passou no parâmetro.

if (RAZ2V == T){ # checagem condicional, se o valor booleano do objeto RAZ2V for True, execute abaixo.

frame[j,3] <- paste("adequada") } # Caso a condição do if seja cumprida, atribua a coluna 3 da linha j de FRAME "adequada", atestando que a razão da amostra foi checada e passou no parâmetro.

else {frame[j,3] <- paste("inadequada")} # Caso a condição do if não seja cumprida, atribua a coluna 3 da linha j de FRAME "adequada", atestando que a razão da amostra foi checada e passou no parâmetro.

if (CONCi > CONC){ #checagem condicional, se o valor booleano da concentração informada, CONCi, for superior ao valor de referência CONC, execute a baixo.

frame[j,4] <- paste("adequada")} # Caso a condição do if seja cumprida, atribua a coluna 4 da linha j de FRAME "adequada", atestando que a razão da amostra foi checada e passou no parâmetro.

else if (CONCi == CONC){ # checagem condicional multipla, caso a concentração informada seja igual a concentração de referência, execute.

frame[j,4] <- paste("adequada(limite))} # Caso o if seja cumprido, atribua "adequada(limite)" a coluna 4 da linha j.

else if (CONCi == (CONC-1)){ # checagem condicional multipla, caso a concentração informada seja igual a concentração de referência -1, permitindo uma margem de aproveitamento da amostra, execute.

frame[j,4] <- paste("adequada(limite))} # Caso o if seja cumprido, atribua "adequada(limite)" a coluna 4 da linha j.

else frame[j,4] <- paste("inadequada(baixa)") # Caso nenhuma condição acima seja cumprida, atribua "inadequada(baixa)" para a coluna 4 da linha j.

if (PUR1V + RAZ2V + CONCV == "4"){ # checagem condicional, se todos os valores booleanos forem TRUE para a razão 260/280 (PUR1V), para a razão

260/230 (RAZ2V) e para a concentração (CONCV), execute.

```
frame[j,5] <- round(((CONC*VOL)/amostra[4]), 1)} # atribua a coluna 5 da linha j o resultado com 1 casa decimal da formula C1*V1=C2*V2, de forma a determinar qual o volume da amostra a ser separado.
```

```
else {frame[j,5] <- paste("----")} # caso a soma de todos os parâmetros seja diferente de 4 significa que alguma razão ou a concentração está fora do parâmetro default ou do determinado pelo usuário, o que será informado no FRAME (resultado da função), então não será possível calcular o volume de amostra e do buffer.
```

```
if (frame[j,5]!="----"){ # checagem condicional, caso na coluna 5 da linha j não tenha "----", simbolizando que todos os parâmetro estão adequados e o Vol. da Amostra pode ser calculado, execute.
```

```
frame[j,6] <- VOL-round(((CONC*VOL)/amostra[4]), 1)} # atribua a coluna 6 da linha j o valor do volume da amostra calculado acima na coluna 5 - o volume final (VOL) determinado pelo usuário. Esse é o volume de buffer para ser completado.
```

```
else{frame[j,6] <- paste("----")} # caso a condição do if não seja cumprido há "----" na coluna 5 da linha j, e por tanto algum parâmetro não está adequado, o Vol. da Amostra não pode ser calculado e por tanto o Vol do Buffer também não.
```

```
}
```

```
return(frame) # retorne o data.frame FRAME como saída da execução da função.
```

```
}
```

From:

<http://ecor.ib.usp.br/> - **ecoR**

Permanent link:

http://ecor.ib.usp.br/doku.php?id=05_curso_antigo:r2018:alunos:trabalho_final:vinyborges:sample.check 

Last update: **2020/08/12 06:04**