For_Exam_ver2

2014311048 백우현

2018년 12월 20일

```
## stringr ##

strings <- c(
  "apple",
  "219 733 8965",
  "329-293-8753",
  "Work: 579-499-7527; Home: 543.355.3679"
)
phone_num <- character(20)
phone <- "([2-9][0-9]{2})[- .]([0-9]{3})[- .]([0-9]{4})"
str_detect(strings, phone) # T/F
```

```
## [1] FALSE  TRUE  TRUE  TRUE
```

```
str_detect(iris$Species, "Seto") %>% sum()
```

```
## [1] 0
```

```
str_subset(strings, phone)
```

```
## [1] "219 733 8965"
## [2] "329-293-8753"
## [3] "Work: 579-499-7527; Home: 543.355.3679"
```

```
str_count(strings,phone)
```

```
## [1] 0 1 1 2
```

```
#phone_num[str_detect(phone_num, "010[. ]\\d+[. ]")]
#phone_num[str_detect(phone_num, "010[^-]\\d{4}[^-]")]
rownames(mtcars)[str_detect(rownames(mtcars), "^D\\w+\\s\\d{3}")]
```

```
## [1] "Datsun 710" "Duster 360"
```

```
#replace vs case_when
str_replace(colnames(iris) ,'Length','LLL')
```

```
## [1] "Sepal.LLL"   "Sepal.Width" "Petal.LLL"   "Petal.Width" "Species"
```

```
case_when( str_detect(colnames(iris) ,'Length')~ 'aaa',
           TRUE ~ as.character(colnames(iris)))
```

```
## [1] "aaa"           "Sepal.Width" "aaa"          "Petal.Width" "Species"
```

```
input <- rownames(mtcars)[1:15];
str_locate_all(strings, phone)
```

```
## [[1]]
##      start end
##
## [[2]]
##      start end
## [1,]     1  12
##
## [[3]]
##      start end
## [1,]     1  12
##
## [[4]]
##      start end
## [1,]     7  18
## [2,]    27  38
```

```
str_locate(strings, phone)
```

```
##      start end
## [1,]    NA  NA
## [2,]     1  12
## [3,]     1  12
## [4,]     7  18
```

```
# + str_sub 와 세트
```

```
str_extract(strings, phone)
```

```
## [1] NA              "219 733 8965" "329-293-8753" "579-499-7527"
```

```
str_extract_all(strings, phone)
```

```
## [[1]]
## character(0)
##
## [[2]]
## [1] "219 733 8965"
##
## [[3]]
## [1] "329-293-8753"
##
## [[4]]
## [1] "579-499-7527" "543.355.3679"
```

```r
input <- rownames(mtcars)[1:15];
str_extract(input, "\\d") # digit
```

```
## [1] "4" "4" "7" "4" NA  NA  "3" "2" "2" "2" "2" "4" "4" "4" NA
```

```r
str_extract(input, "\\D")
```

```
## [1] "M" "M" "D" "H" "H" "V" "D" "M" "M" "M" "M" "M" "M" "M" "C"
```

```r
str_extract(input, "\\w") #word
```

```
## [1] "M" "M" "D" "H" "H" "V" "D" "M" "M" "M" "M" "M" "M" "M" "C"
```

```r
str_extract(input, "\\W")
```

```
## [1] " " " " " " " " " " NA  " " " " " " " " " " " " " " " " " "
```

```r
str_extract(input, "\\s") # space
```

```
## [1] " " " " " " " " " " NA  " " " " " " " " " " " " " " " " " "
```

```r
# 시작
input[str_detect(input, "^D")]
```

```
## [1] "Datsun 710" "Duster 360"
```

```r
# 끝
input[str_detect(input, "0$")]
```

```
## [1] "Datsun 710" "Duster 360" "Merc 230"   "Merc 280"
```

```r
# \\b can match every beginning/end of string of a character element
rownames(mtcars)[str_detect(rownames(mtcars), "\\bC")]
```

```
## [1] "Cadillac Fleetwood"  "Lincoln Continental" "Chrysler Imperial"
## [4] "Honda Civic"         "Toyota Corolla"      "Toyota Corona"
## [7] "Dodge Challenger"    "Camaro Z28"
```

```r
rownames(mtcars)[str_detect(rownames(mtcars), "e\\b")]
```

```
## [1] "Hornet 4 Drive"   "Dodge Challenger" "Porsche 914-2"
```

```r
rownames(mtcars)[str_detect(row.names(mtcars),"^C")]
```

```
## [1] "Cadillac Fleetwood" "Chrysler Imperial"  "Camaro Z28"
```

```
#extract all numb
gsub("\\W((.+)\\W)","-\\1",strings)
```

```
## [1] "apple"
## [2] "219 733 8965"
## [3] "329-293-8753"
## [4] "Work: 579-499-7527; Home: 543.355.3679"
```

```
x <- c("abcdef", "ghifjk")
str_dup(x, c(2, 3))
```

```
## [1] "abcdefabcdef"        "ghifjkghifjkghifjk"
```

```
x <- c(" a ", "b ", " c")
str_trim(x)
```

```
## [1] "a" "b" "c"
```

```
str_trim(x, "left")
```

```
## [1] "a " "b " "c"
```

```
# [ ] 사이에 or 가 숨어있고 유일한 예외 : -

set.seed(2007)
year <- sample(c(2006:2017), 20, replace = TRUE)
number <- sample(c(310000:320000), 20, replace = TRUE)
student_ID <- paste0(year, number); student_ID
```

```
##  [1] "2014318186" "2009315671" "2007312545" "2013315769" "2010317115"
##  [6] "2007314875" "2017310136" "2009312841" "2011314088" "2014314133"
## [11] "2017313751" "2006313927" "2016312615" "2016312988" "2007313538"
## [16] "2014316015" "2013315134" "2013316312" "2017317757" "2011318137"
```

```
student_ID[str_detect(student_ID, "20[01][0-389]")]
```

```
## [1] "2009315671" "2013315769" "2010317115" "2009312841" "2011314088"
## [6] "2013315134" "2013316312" "2011318137"
```

```
student_ID[str_detect(student_ID, "20[01][0-389]")]
```

```
## [1] "2009315671" "2013315769" "2010317115" "2009312841" "2011314088"
## [6] "2013315134" "2013316312" "2011318137"
```

```
LETTERS[str_detect(LETTERS, "[AE-O]")]
```

```
## [1] "A" "E" "F" "G" "H" "I" "J" "K" "L" "M" "N" "O"
```

```
student_ID[str_detect(student_ID, "20(08|09|10|11|12|13)")]
```

```
## [1] "2009315671" "2013315769" "2010317115" "2009312841" "2011314088"
## [6] "2013315134" "2013316312" "2011318137"
```

```
set.seed(2007)
separator <- sample(c("-", ".", " "), 20, replace = TRUE, prob = c(0.8, 0.1, 0.1))
phone_num <- character(20)
for (i in 1:20) phone_num[i] <- paste("010", sample(1000:9999), sample(1000:9999), sep = separator[i])
head(phone_num, 10)
```

```
## [1] "010-8366-9980" "010-3754-8053" "010-3733-8189" "010-6499-3758"
## [5] "010-6668-7499" "010-5280-5627" "010.6307.3763" "010-5045-9981"
## [9] "010-1769-7613" "010-1541-1929"
```

```
phone_num[str_detect(phone_num, "010[. ]\dd\d\d[. ]")]
```

```
## [1] "010.6307.3763" "010.9023.8758" "010 4815 9050" "010 5172 5343"
## [5] "010.7581.8023"
```

```
phone_num[str_detect(phone_num, "010[^-]\d\d\d\d[^-]")]
```

```
## [1] "010.6307.3763" "010.9023.8758" "010 4815 9050" "010 5172 5343"
## [5] "010.7581.8023"
```

```
input[str_detect(input, "^([a-zA-Z]+)\s([a-zA-Z]+)$")]
```

```
## [1] "Hornet Sportabout"  "Cadillac Fleetwood"
```

```
strings <- c(" 219 733 8965", "329-293-8753 ", "banana", "595 794 7569",
             "387 287 6718", "apple", "233.398.9187  ", "482 952 3315",
             "239 923 8115 and 842 566 4692", "Work: 579-499-7527", "$1000",
             "Home: 543.355.3679")
phone <- "([2-9][0-9]{2})[- .]([0-9]{3})[- .]([0-9]{4})"

str_extract(strings, phone)
```

```
## [1] "219 733 8965" "329-293-8753" NA              "595 794 7569"
## [5] "387 287 6718" NA              "233.398.9187" "482 952 3315"
## [9] "239 923 8115" "579-499-7527" NA              "543.355.3679"
```

```
str_match(strings, phone)
```

```
##        [,1]              [,2]  [,3]  [,4]
## [1,] "219 733 8965"   "219" "733" "8965"
## [2,] "329-293-8753"   "329" "293" "8753"
## [3,] NA               NA    NA    NA
## [4,] "595 794 7569"   "595" "794" "7569"
## [5,] "387 287 6718"   "387" "287" "6718"
## [6,] NA               NA    NA    NA
## [7,] "233.398.9187"   "233" "398" "9187"
## [8,] "482 952 3315"   "482" "952" "3315"
## [9,] "239 923 8115"   "239" "923" "8115"
## [10,] "579-499-7527"  "579" "499" "7527"
## [11,] NA              NA    NA    NA
## [12,] "543.355.3679"  "543" "355" "3679"
```

```
str_replace(strings, phone, "XXX-XXX-XXXX")
```

```
## [1] " XXX-XXX-XXXX"                  "XXX-XXX-XXXX "
## [3] "banana"                         "XXX-XXX-XXXX"
## [5] "XXX-XXX-XXXX"                   "apple"
## [7] "XXX-XXX-XXXX  "                 "XXX-XXX-XXXX"
## [9] "XXX-XXX-XXXX and 842 566 4692" "Work: XXX-XXX-XXXX"
## [11] "$1000"                         "Home: XXX-XXX-XXXX"
```

```
str_split("a-b-c", "-")
```

```
## [[1]]
## [1] "a" "b" "c"
```

```
str_split_fixed("a-b-c", "-", n = 2)
```

```
##      [,1] [,2]
## [1,] "a"  "b-c"
```

```
result <- str_match(input, "^([a-zA-Z]+)₩₩s([a-zA-Z]+)$")
result[complete.cases(result), ]
```

```
##      [,1]                 [,2]       [,3]
## [1,] "Hornet Sportabout"  "Hornet"   "Sportabout"
## [2,] "Cadillac Fleetwood" "Cadillac" "Fleetwood"
```

```
# 0-3 / 7 을 제외
student_ID[str_detect(student_ID, "20[^0][^0-37]")]
```

```
## [1] "2014318186" "2014314133" "2016312615" "2016312988" "2014316015"
```

```
###### dplyr ++ ######

x <- c(1, -1, 0, 10)
100 / x
```

```
## [1]  100 -100  Inf   10
```

```
100 / na_if(x, 0)
```

```
## [1]  100 -100   NA   10
```

```
scramble <- function(x) x[sample(nrow(x)), sample(ncol(x))]
# By default, ordering of rows and columns ignored
all_equal(mtcars, scramble(mtcars))
```

```
## [1] TRUE
```

```
# But those can be overriden if desired
all_equal(mtcars, scramble(mtcars), ignore_col_order = FALSE)
```

```
## [1] "Same column names, but different order"
```

```
all_equal(mtcars, scramble(mtcars), ignore_row_order = FALSE)
```

```
## [1] "Same row values, but different order"
```

```
mtcars %>% add_count(cyl)
```

```
## # A tibble: 32 x 12
##      mpg   cyl  disp    hp  drat    wt  qsec    vs    am  gear  carb     n
##    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <int>
## 1  21       6  160    110  3.9   2.62  16.5     0     1     4     4     7
## 2  21       6  160    110  3.9   2.88  17.0     0     1     4     4     7
## 3  22.8     4  108     93  3.85  2.32  18.6     1     1     4     1    11
## 4  21.4     6  258    110  3.08  3.22  19.4     1     0     3     1     7
## 5  18.7     8  360    175  3.15  3.44  17.0     0     0     3     2    14
## 6  18.1     6  225    105  2.76  3.46  20.2     1     0     3     1     7
## 7  14.3     8  360    245  3.21  3.57  15.8     0     0     3     4    14
## 8  24.4     4  147.    62  3.69  3.19  20       1     0     4     2    11
## 9  22.8     4  141.    95  3.92  3.15  22.9     1     0     4     2    11
## 10 19.2     6  168.   123  3.92  3.44  18.3     1     0     4     4     7
## # ... with 22 more rows
```

```
# ungroup 필요 없음
starwars %>% count(species, homeworld, sort = TRUE)
```

```
## # A tibble: 58 x 3
##    species  homeworld      n
##    <chr>    <chr>      <int>
##  1 Human    Tatooine       8
##  2 Human    Naboo          5
##  3 Human    <NA>           5
##  4 Gungan   Naboo          3
##  5 Human    Alderaan       3
##  6 Droid    Tatooine       2
##  7 Droid    <NA>           2
##  8 Human    Corellia       2
##  9 Human    Coruscant      2
## 10 Kaminoan Kamino         2
## # ... with 48 more rows
```

```
# 변수값별로 다른 list 로 구분
split(mtcars, mtcars$cyl)
```

```
## $`4`
##                  mpg cyl  disp  hp drat    wt  qsec vs am gear carb
## Datsun 710      22.8   4 108.0  93 3.85 2.320 18.61  1  1    4    1
## Merc 240D       24.4   4 146.7  62 3.69 3.190 20.00  1  0    4    2
## Merc 230        22.8   4 140.8  95 3.92 3.150 22.90  1  0    4    2
## Fiat 128        32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
## Honda Civic     30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
## Toyota Corolla 33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
## Toyota Corona  21.5   4 120.1  97 3.70 2.465 20.01  1  0    3    1
## Fiat X1-9       27.3   4  79.0  66 4.08 1.935 18.90  1  1    4    1
## Porsche 914-2   26.0   4 120.3  91 4.43 2.140 16.70  0  1    5    2
## Lotus Europa    30.4   4  95.1 113 3.77 1.513 16.90  1  1    5    2
## Volvo 142E      21.4   4 121.0 109 4.11 2.780 18.60  1  1    4    2
##
## $`6`
##                  mpg cyl  disp  hp drat    wt  qsec vs am gear carb
## Mazda RX4       21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
## Mazda RX4 Wag   21.0   6 160.0 110 3.90 2.875 17.02  0  1    4    4
## Hornet 4 Drive 21.4   6 258.0 110 3.08 3.215 19.44  1  0    3    1
## Valiant         18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
## Merc 280        19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
## Merc 280C       17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
## Ferrari Dino    19.7   6 145.0 175 3.62 2.770 15.50  0  1    5    6
##
## $`8`
##                     mpg cyl  disp  hp drat    wt  qsec vs am gear carb
## Hornet Sportabout  18.7   8 360.0 175 3.15 3.440 17.02  0  0    3    2
## Duster 360         14.3   8 360.0 245 3.21 3.570 15.84  0  0    3    4
## Merc 450SE         16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
## Merc 450SL         17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
## Merc 450SLC        15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
## Cadillac Fleetwood 10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
## Lincoln Continental 10.4  8 460.0 215 3.00 5.424 17.82  0  0    3    4
## Chrysler Imperial  14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
## Dodge Challenger   15.5   8 318.0 150 2.76 3.520 16.87  0  0    3    2
## AMC Javelin        15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
## Camaro Z28         13.3   8 350.0 245 3.73 3.840 15.41  0  0    3    4
## Pontiac Firebird   19.2   8 400.0 175 3.08 3.845 17.05  0  0    3    2
## Ford Pantera L     15.8   8 351.0 264 4.22 3.170 14.50  0  1    5    4
## Maserati Bora      15.0   8 301.0 335 3.54 3.570 14.60  0  1    5    8
```

```
# 교집합, 합집합
first <- mtcars[1:20, ]
second <- mtcars[10:32, ]
intersect(first, second)
```

```
##    mpg cyl  disp  hp drat    wt  qsec vs am gear carb
## 1  19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
## 2  17.8   6 167.6 123 3.92 3.440 18.90  1  0    4    4
## 3  16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
## 4  17.3   8 275.8 180 3.07 3.730 17.60  0  0    3    3
## 5  15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
## 6  10.4   8 472.0 205 2.93 5.250 17.98  0  0    3    4
## 7  10.4   8 460.0 215 3.00 5.424 17.82  0  0    3    4
## 8  14.7   8 440.0 230 3.23 5.345 17.42  0  0    3    4
## 9  32.4   4  78.7  66 4.08 2.200 19.47  1  1    4    1
## 10 30.4   4  75.7  52 4.93 1.615 18.52  1  1    4    2
## 11 33.9   4  71.1  65 4.22 1.835 19.90  1  1    4    1
```

```
union(first, second) %>% head()
```

```
##    mpg cyl  disp  hp drat    wt  qsec vs am gear carb
## 1 15.2   8 304.0 150 3.15 3.435 17.30  0  0    3    2
## 2 21.0   6 160.0 110 3.90 2.620 16.46  0  1    4    4
## 3 18.1   6 225.0 105 2.76 3.460 20.22  1  0    3    1
## 4 15.2   8 275.8 180 3.07 3.780 18.00  0  0    3    3
## 5 19.2   6 167.6 123 3.92 3.440 18.30  1  0    4    4
## 6 16.4   8 275.8 180 3.07 4.070 17.40  0  0    3    3
```

```
union_all(first, second) %>% head()
```

```
##    mpg cyl disp  hp drat    wt  qsec vs am gear carb
## 1 21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## 2 21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## 3 22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## 4 21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## 5 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## 6 18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
setdiff(first, second) %>% head()
```

```
##    mpg cyl disp  hp drat    wt  qsec vs am gear carb
## 1 21.0   6  160 110 3.90 2.620 16.46  0  1    4    4
## 2 21.0   6  160 110 3.90 2.875 17.02  0  1    4    4
## 3 22.8   4  108  93 3.85 2.320 18.61  1  1    4    1
## 4 21.4   6  258 110 3.08 3.215 19.44  1  0    3    1
## 5 18.7   8  360 175 3.15 3.440 17.02  0  0    3    2
## 6 18.1   6  225 105 2.76 3.460 20.22  1  0    3    1
```

```
setequal(first,second) %>% head()
```

```
## [1] FALSE
```

```
## case_when!!!!!! ######

x <- 1:50
case_when(
  x %% 35 == 0 ~ "fizz buzz",
  x %% 5 == 0 ~ "fizz",
  x %% 7 == 0 ~ "buzz",
  TRUE ~ as.character(x)
)
```

```
##  [1] "1"        "2"        "3"        "4"        "fizz"
##  [6] "6"        "buzz"     "8"        "9"        "fizz"
## [11] "11"       "12"       "13"       "buzz"     "fizz"
## [16] "16"       "17"       "18"       "19"       "fizz"
## [21] "buzz"     "22"       "23"       "24"       "fizz"
## [26] "26"       "27"       "buzz"     "29"       "fizz"
## [31] "31"       "32"       "33"       "34"       "fizz buzz"
## [36] "36"       "37"       "38"       "39"       "fizz"
## [41] "41"       "buzz"     "43"       "44"       "fizz"
## [46] "46"       "47"       "48"       "buzz"     "fizz"
```

```
a<-iris$Sepal.Length
case_when(a >= 7 ~ 100,
          TRUE ~ as.double(a))
```

```
##   [1]   5.1   4.9   4.7   4.6   5.0   5.4   4.6   5.0   4.4   4.9   5.4
##  [12]   4.8   4.8   4.3   5.8   5.7   5.4   5.1   5.7   5.1   5.4   5.1
##  [23]   4.6   5.1   4.8   5.0   5.0   5.2   5.2   4.7   4.8   5.4   5.2
##  [34]   5.5   4.9   5.0   5.5   4.9   4.4   5.1   5.0   4.5   4.4   5.0
##  [45]   5.1   4.8   5.1   4.6   5.3   5.0 100.0   6.4   6.9   5.5   6.5
##  [56]   5.7   6.3   4.9   6.6   5.2   5.0   5.9   6.0   6.1   5.6   6.7
##  [67]   5.6   5.8   6.2   5.6   5.9   6.1   6.3   6.1   6.4   6.6   6.8
##  [78]   6.7   6.0   5.7   5.5   5.5   5.8   6.0   5.4   6.0   6.7   6.3
##  [89]   5.6   5.5   5.5   6.1   5.8   5.0   5.6   5.7   5.7   6.2   5.1
## [100]   5.7   6.3   5.8 100.0   6.3   6.5 100.0   4.9 100.0   6.7 100.0
## [111]   6.5   6.4   6.8   5.7   5.8   6.4   6.5 100.0 100.0   6.0   6.9
## [122]   5.6 100.0   6.3   6.7 100.0   6.2   6.1   6.4 100.0 100.0 100.0
## [133]   6.4   6.3   6.1 100.0   6.3   6.4   6.0   6.9   6.7   6.9   5.8
## [144]   6.8   6.7   6.7   6.3   6.5   6.2   5.9
```

```
iris %>% mutate(SS = case_when(Sepal.Length >=7 ~ 100,
                               TRUE ~ as.double(Sepal.Length))  ) %>% select(SS) %>% head()
```

```
##    SS
## 1 5.1
## 2 4.9
## 3 4.7
## 4 4.6
## 5 5.0
## 6 5.4
```

```
iris %>% mutate(SS = case_when(Species =='setosa' ~ 'sex',
                               TRUE ~ as.character(Species))  ) %>% select(SS) %>% head()
```

```
##    SS
## 1 sex
## 2 sex
## 3 sex
## 4 sex
## 5 sex
## 6 sex
```

```
vecs <- list(
  c(1, 2, NA, NA, 5),
  c(NA, NA, 3, 4, 5)
)
## NA 인 놈들 바꾸기
x <- sample(c(1:5, NA, NA, NA))
coalesce(x, 100L)
```

```
## [1]   4   5 100 100   1   2   3 100
```

```
## NA 로 missing 인 부분 합치기
y <- c(1, 2, NA, NA, 5)
z <- c(NA, NA, 3, 4, 5)
coalesce(y, z)
```

```
## [1] 1 2 3 4 5
```

```
coalesce(!!!vecs)
```

```
## [1] 1 2 3 4 5
```

```
# near
sqrt(2) ^ 2 == 2
```

```
## [1] FALSE
```

```
near(sqrt(2) ^ 2, 2)
```

```
## [1] TRUE
```

```
# nth
x <- 1:10
y <- 13:22
z<-2:11

dplyr::first(x)
```

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

```
dplyr::last(y)
```

```
## [1] 22
```

```
nth(x, 5)
```

```
## [1] 5
```

```
dplyr::last(x,y,z)
```

```
## [1] 10
```

```
# 유니크 밸류
n_distinct(iris$Species)
```

```
## [1] 3
```

```
## Function ###


rescale01 <- function(x) {
  rng <- range(x, na.rm = TRUE, finite = TRUE)
  (x - rng[1]) / (rng[2] - rng[1])
}
rescale01(x)
```

```
##  [1] 0.0000000 0.1111111 0.2222222 0.3333333 0.4444444 0.5555556 0.6666667
##  [8] 0.7777778 0.8888889 1.0000000
```