pandas (R)osetta: R

Intro to R for Pandas users and vice versa.

Lisa Malins

Contents

This is a demonstration of basic data wrangling operations in base R.

Sister notebooks demonstrate the exact same operations in Tidyverse R and the Pandas library in Python, just like the Rosetta Stone.

I/O

Create dataframe from scratch

```
df <- data.frame(
  letter = c("a", "b", "c", "d", "e"),
  number = c(1:5),
  fruit = c("apple", "banana", "coconut", "date", "elderberry"),
  vegetable = c("arugula", "beet", "carrot", "daikon", "eggplant"),
  name = c("Alice", "Bob", "Carol", "Dan", "Eve")
)
df
```

```
##   letter number      fruit vegetable  name
## 1      a      1      apple   arugula Alice
## 2      b      2     banana      beet   Bob
## 3      c      3    coconut    carrot Carol
## 4      d      4       date    daikon   Dan
## 5      e      5 elderberry  eggplant   Eve
```

Write

```
write.csv(df, "data/R_letters.csv", row.names=FALSE)
write.table(df, "data/R_letters.tsv", sep="\t", row.names=FALSE)
```

Read

```
read.csv("data/R_letters.csv")
```

```
##   letter number      fruit vegetable  name
## 1      a      1      apple   arugula Alice
## 2      b      2     banana      beet   Bob
## 3      c      3    coconut    carrot Carol
## 4      d      4       date    daikon   Dan
## 5      e      5 elderberry  eggplant   Eve
```

```
read.delim("data/R_letters.tsv")
```

```
##   letter number      fruit vegetable  name
## 1      a      1      apple   arugula Alice
## 2      b      2     banana      beet   Bob
## 3      c      3    coconut    carrot Carol
## 4      d      4       date    daikon   Dan
## 5      e      5 elderberry  eggplant   Eve
```

Accessing data

Note for Pandas users:

- Both R and Pandas have the same convention of `[row, column]` for retrieving a cell from a dataframe.
- However, if only one number is specified with no comma, in pandas a row is returned, but in R a column is returned.
- Also, remember that **R has 1-based indexing** while Python has 0-based indexing.
- Thus,
  - In pandas: `df.iloc[1]` returns the second row
  - In R: `df[1]` returns the first column

```
# Returns first column ("letter")
df[1]
```

```
##   letter
## 1      a
## 2      b
## 3      c
## 4      d
## 5      e
```

Select cell

To **select a single cell**, **use double brackets** with the row number, followed by the column number or name.

```
# Get "banana" cell value
df[[2, 3]]
df[[2, "fruit"]]
```

```
## [1] "banana"
## [1] "banana"
```

Cells can also be selected with single brackets. The two notations *are* different and are used for different things (more on that later), but when selecting a single value, the distinction is moot in base R.

```
# Usually, single brackets return a dataframe,
# but when it's a 1x1 selection, just the value is returned
df[2, 3]
df[2, "fruit"]
# The 4 expressions above are all identical
```

```
## [1] "banana"
## [1] "banana"
```