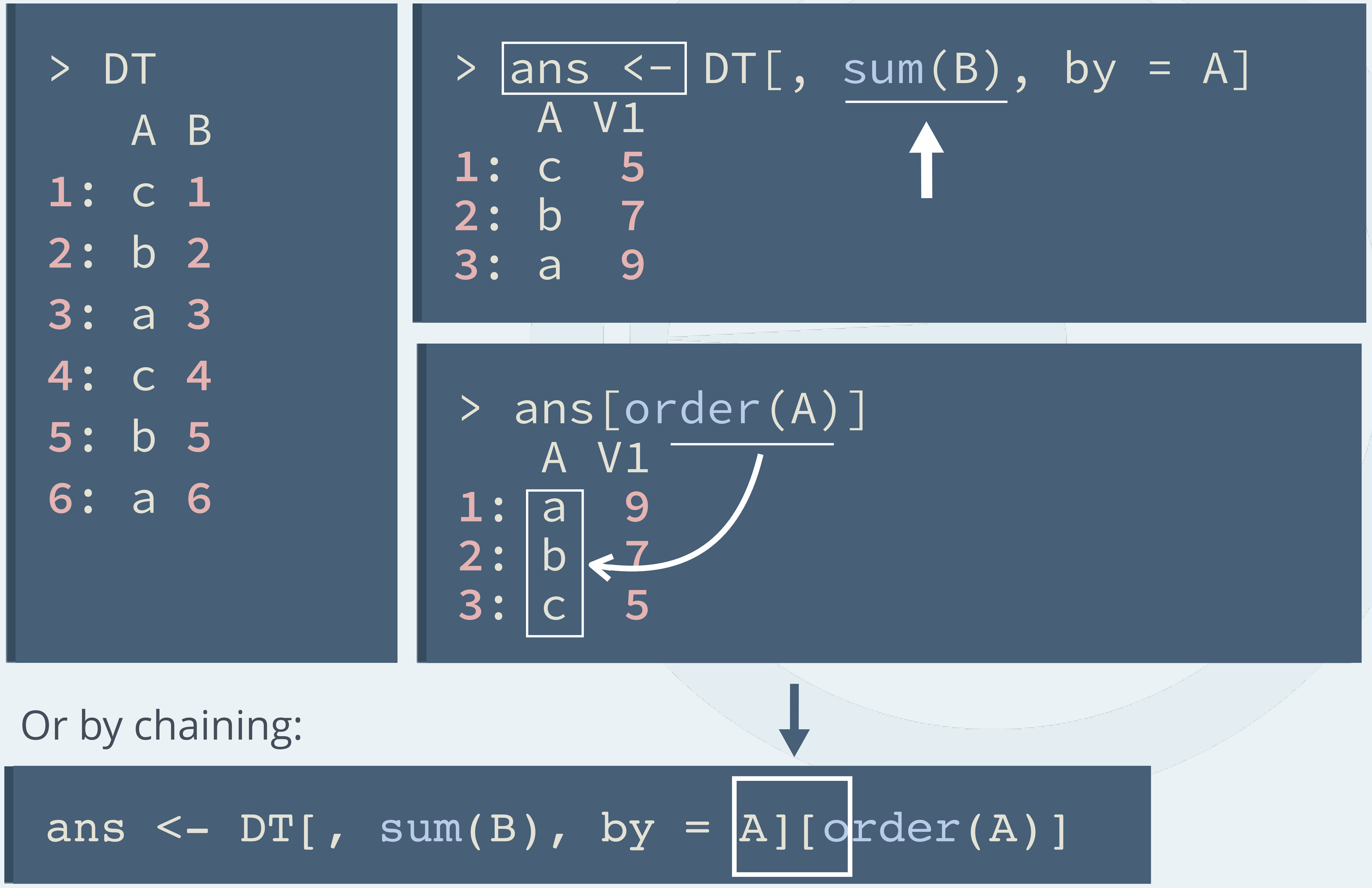
DATA ANALYSIS - THE DATA TABLE WAY

CHAINING

Chaining operations together



```
> DT
   A B
1: c 1
2: b 2
3: a 3
4: c 4
5: b 5
6: a 6
```

```
> ans <- DT[, sum(B), by = A]
   A V1
1: c  5
2: b  7
3: a  9
```

```
> ans[order(A)]
   A V1
1: a  9
2: b  7
3: c  5
```

Or by chaining:

```
ans <- DT[, sum(B), by = A][order(A)]
```

DATA.TABLE

Let’s practice

DATA ANALYSIS - THE DATA TABLE WAY

SUBSET OF DATA

Recapping previous exercise

```
> DT <- as.data.table(iris)
> DT[, .(Sepal.Length = median(Sepal.Length),
         Sepal.Width  = median(Sepal.Width),
         Petal.Length = median(Petal.Length),
         Petal.Width  = median(Petal.Width)),
  by = Species]
```

What if you had more than 100 columns?

[S]ubset of [D]ata - .SD

```
> DT[, lapply(.SD, median), by = Species]

      Species Sepal.Length Sepal.Width Petal.Length Petal.Width
1:     setosa          5.0         3.4         1.50         0.2
2: versicolor          5.9         2.8         4.35         1.3
3:  virginica          6.5         3.0         5.55         2.0
```

NB: .() is an alias to list() and lapply() returns a list.

DATA.TABLE

Let’s practice

DATA ANALYSIS - THE DATA TABLE WAY

USING := IN J

Add/update columns in j using :=

DATA.TABLE

```
> DT

   x  y
1: 1  6
2: 1  7
3: 1  8
4: 2  9
5: 2 10
```

LHS

RHS

```
> DT[, c(“x”, “z”) := .(rev(x), 10:6)]

   x  y  z
1: 2  6       10
2: 2  7       9
3: 1  8       8
4: 1  9       7
5: 1  10      6
```

Shortcut if just one :

```
DT[, x := rev(x)]
```

Remove columns using :=

```
> DT

   x  y  z  a
1: 2  6  10 10
2: 2  7  9  9
3: 1  8  8  8
4: 1  9  7  7
5: 1  10 6  6
```

```
> DT[, c(“y”, “z”) := NULL]
```

Shortcut if just one :

```
DT[, y := NULL]
```

Remove columns using :=

```
> DT

   x  y  z  a
1: 2  6  10 10
2: 2  7  9  9
3: 1  8  8  8
4: 1  9  7  7
5: 1  10 6  6
```

```
MyCols = c(“y”, “z”)

DT[, (MyCols) := NULL]
```

```
DT[, paste0(“colNamePrefix", 1:4) := NULL ]
```

Functional :=

```
> DT

   x  a  y  z
1: 2  10 6  1
2: 2  9  7  1
3: 1  8  8  1
4: 1  7  9  1
5: 1  6  10 1
```

```
> DT[, `:=`(y = 6:10,    # y (kg)
            z = 1)]
```

:= combined with i and by

```
> DT

   x  y  z
1: 2  6 NA
2: 2  7 NA
3: 1  8 17
4: 1  9 17
5: 1 10 NA
```

```
> DT[2:4, z := sum(y), by = x]
```

DATA.TABLE

Let’s practice

DATA ANALYSIS - THE DATA TABLE WAY

USING SET()

set()

```
> DT


     x   y   z
1:   1   1   2
2:   2   8   5
3:   3   1   4
4:   4   1   2
5:   5   1   3
```

```
> for (i in 1:5) DT[i, z := i+1]

> for (i in 1:5) set(DT, i, 3L, i+1)


     x   y   z
1:   1   1   2
2:   2   8   3
3:   3   1   4
4:   4   1   5
5:   5   1   6
```

setnames()

```
> DT

   x  y
1: 1  a
2: 2  b
3: 3  c
4: 4  d
5: 5  e
```

```
> setnames(DT, “y”, “z”))

   x  z
1: 1  a
2: 2  b
3: 3  c
4: 4  d
5: 5  e
```

setnames(DT, “old”, “new”)
convenient and robust

The new names are assigned by reference.

setcolorder()

```
> DT

   x  y
1: 1  a
2: 2  b
3: 3  c
4: 4  d
5: 5  e
```

```
> setcolorder(DT, c(“y”, “x”))

   y  x
1: a  1
2: b  2
3: c  3
4: d  4
5: e  5
```

DT’s column orders are changed by reference.

DATA.TABLE

Let’s practice