

# MVLU COLLEGE R PRACTICAL 5

**Aim:** Sorting data using arrange() in R.

```
1 # Load the dplyr package
2 library(dplyr)
3
4 # --- ASSUMPTION: Data Loading ---
5 # We assume 'Fastest_Fifties' is loaded correctly.
6 # We'll use read.csv to match the previous fix's column names (like X6s, X8s).
7 fastest_fifties <- read.csv("Fastest Fifties - 2018.csv")
8
9
10 # Example 1: Sorting by a single variable (Ascending)
11 #
12 # Original Logic: Sort by 'volume' (lowest to highest)
13 # Adapted for you: Sort by 'BF' (balls faced) to find the absolute fastest Fifties.
14
15 # Sort by BF (balls faced) in ascending order (lowest to highest)
16 fifties_by_speed <- fastest_fifties %>%
17   arrange(BF)
18
19
20 cat("\n--- Top 5 Fastest Fifties (Lowest BF) ---\n")
21 # Check the first 5 rows-the lowest BF values should be at the top.
22 print(head(fifties_by_speed, 5))
23
24
25 # Example 2: Sorting by a single variable (Descending)
26 #
27 # Original Logic: Sort by 'high' price (highest to lowest)
28 # Adapted for you: Find the rows with the highest scores ('runs').
29
30 # Sort by runs in descending order (highest to lowest)
31 fifties_by_score <- fastest_fifties %>%
32   arrange(desc(Runs))
33
34
35 cat("\n--- Top 5 Highest Individual Scores ---\n")
36 # Check the first 5 rows-the highest run values should be at the top.
37 print(head(fifties_by_score, 5))
38
39
40 # Example 3: Sorting by Multiple columns
41 #
42 # Original Logic: Sort by 'high' (ascending), then by 'volume' (descending).
43 # Adapted for you: Sort first by 'BF' (Ascending, fastest scores first), and for ties, sort by 'Runs' (Descending, high)
44
45 # 1. Primary Sort: BF (Ascending, lowest balls faced first)
46 # 2. Secondary Sort (tie breaker): Runs (Descending, highest score first)
47 fifties_multi_sort <- fastest_fifties %>%
48   arrange(BF, desc(Runs))
49
50
51 cat("\n--- Top 5 Sorted by Speed (BF) then Score (Runs) ---\n")
52 # The results show lower BF first, but if BF is equal, the higher run score comes first.
53 print(head(fifties_multi_sort, 5))
54
55
56 # Example 4: Combined Filter and Sort (Chaining operations)
57 #
58 # Original filter: filter(high > 60, 50, volume > 300)
59 # Adapted filter: Filter for a high score (> 80 runs) AND high power hitting (> 5 sixes).
60 # Sorting: Then, sort those results by 'X6s' (Sixes) in descending order to see the most powerful knocks first.
61
62 # Filter for high runs and high 6s, then arrange by 6s
63 power_knocks_sorted <- fastest_fifties %>%
64   filter(Runs > 80, X6s > 5) %>%
65   arrange(desc(X6s))
66
67
68 cat("\n--- Top Power knocks (Runs > 80 & 6s > 5, Sorted by 6s) ---\n")
69 # Check the columns for confirmation
70 print(power_knocks_sorted %>% head(5))
71
72
```

**Output:-**

Hariprasad Vishwakarm  
S126  
SYCS

MVLU COLLEGE  
R PRACTICAL 5

The image displays a RStudio IDE window with a script editor on the left and an environment pane on the right. The script editor contains R code for data manipulation and visualization. The code starts with loading the 'dplyr' and 'ggplot2' libraries. It then reads a CSV file named 'fastest\_fifties.csv' into a data frame. The data is sorted by 'runs' in descending order. The code uses 'group\_by()' to group the data by 'player' and 'venue'. It then uses 'summarize()' to calculate the total runs for each player and venue. The results are printed to the console. The environment pane on the right shows the loaded objects: 'fastest\_fifties', 'fastest\_fifties\_tbl', 'fastest\_fifties...2018', 'fifties\_by\_score', 'fifties\_by\_speed', 'fifties\_multi\_sort', 'high\_runs\_high\_fours\_filt', 'high\_score\_subset', 'ipl\_2015\_full\_runs', 'low\_bat\_filter', 'low\_sixes\_high\_score\_subs', 'player\_filter', and 'power\_knocks\_sorted'. The environment pane also shows the size and modified date of each object.

Hariprasad Vishwakarma  
S126  
SYCS

```

Source
Console
Terminal
Background Jobs

R #452 - v1
POS Player runs BF x6s x6s Against Venue Match.date
13 Ishabh Pant 128 36 11 7 SRM wankhede stadium 30 May 2018
27 Shane watson 117 33 11 8 SRM wankhede stadium 27 May 2018
39 Shane watson 106 28 9 6 RR maharashtra cricket association's international stadium 20 April 2018
4 93 Chris Gayle 104 39 3 11 SRM IS Bindra stadium 19 April 2018
44 ambati rayudu 100 31 7 7 SRM maharashtra cricket association's international stadium 13 May 2018
# 1. Sorting: lower balls faced first
# 2. Secondary sort (tie breaker): runs (Descending, highest score first)
# 5. multi-sort > fastest_fifties >
+ arrange(BF, desc(x6s))
> cat("\n--- Top 5 Sorted by speed (BF) then Score (Runs) ---\n")

--- Top 5 Sorted by speed (BF) then Score (Runs) ---
# The results show low BF first, but if BF is equal, the higher run score comes first.
# print(head(fifties_multi_sort, 5))
POS Player runs BF x6s x6s Against Venue Match.date
1 K. Rahul 31 14 6 4 CC IS Bindra stadium 08 April 2018
2 Ishan kishan 62 17 5 6 KKR Eden gardens 09 May 2018
3 Sunil Narine 50 17 4 5 RCB Eden gardens 08 April 2018
4 4 Jos Buttler 67 18 4 7 CC Arun Jaitley stadium 02 May 2018
5 5 Sam Billings 56 21 2 5 KKR chidambaram 10 April 2018
# 1. Primary sort: BF (Descending, lowest balls faced first)
# 2. Secondary sort (tie breaker): runs (Descending, highest score first)
# 5. multi-sort > fastest_fifties >
+ arrange(BF, desc(x6s))
> cat("\n--- Top 5 Sorted by speed (BF) then Score (Runs) ---\n")

--- Top 5 Sorted by speed (BF) then Score (Runs) ---
# The results show low BF first, but if BF is equal, the higher run score comes first.
# print(head(fifties_multi_sort, 5))
POS Player runs BF x6s x6s Against Venue Match.date
1 K. Rahul 31 14 6 4 CC IS Bindra stadium 08 April 2018
2 Ishan kishan 62 17 5 6 KKR Eden gardens 09 May 2018
3 Sunil Narine 50 17 4 5 RCB Eden gardens 08 April 2018
4 4 Jos Buttler 67 18 4 7 CC Arun Jaitley stadium 02 May 2018
5 5 Sam Billings 56 21 2 5 KKR chidambaram 10 April 2018
# Filter for high runs and High 6s, then arrange by 6s
# power_knocks_sorted <- fastest_fifties >
+ filter(x6s > 80, x6s > 5) >
+ arrange(desc(x6s))
> cat("\n--- Top Power Knocks (Runs > 80 & 6s > 5, Sorted by 6s) ---\n")

--- Top Power Knocks (Runs > 80 & 6s > 5, Sorted by 6s) ---
# Check the columns for confirmation
# print(power_knocks_sorted > head(3))
POS Player runs BF x6s x6s Against Venue Match.date
1 21 andrew russell 88 26 1 11 CSK chidambaram 10 April 2018
2 93 Chris Gayle 104 39 3 11 SRM IS Bindra stadium 19 April 2018
3 35 shreyasyer 93 29 10 KKR Arun Jaitley stadium 27 April 2018
4 59 sanju samson 92 34 2 10 RCB M. Chinnasamy stadium 15 April 2018
5 37 Shane watson 127 33 11 8 SRM wankhede stadium 27 May 2018

```

Hariprasad Vishwakarm  
S126  
SYCS