```
# DPLYR Package: Data wrangling or manipulation package
# 6 functions in dplyr()
# 1. filter() - to extract rows based on certain conditions
# 2. select() - to select particular column
# 3. arrange() - it will sort the data
# 4. groupby() - to group the data (need to use summmarise or mutate)
# 5. summarise() - to aggregate the data
# 6. mutate() - to create a new derived cloumn

# pipeline statement - %>%
# dataname %>% function1 %>% function2

# installing dplyr package

install.packages("dplyr")
library(dplyr)

# set working directory
setwd("D:/RISE - WPU/R Studio")
getwd()

# read the dataset

hrdata = read.csv("HRDataset_v13.csv")
View(hrdata)

# select () - To select the column empname, ID, position, state, dept &
payrate
colnames(hrdata)
hr_select = hrdata %>%
select(Employee_Name,EmpID,Position,State,Department,PayRate)
View(hr_select)

# : - to select the range of columns
hr_select2 = hrdata %>% select(MarriedID : FromDiversityJobFairID)
View(hr_select2)

# to exclude any columns

hr_select3 = hrdata %>% select(-c(MarriedID,GenderID,PerfScoreID,PositionID))
View(hr_select3)

# 2. filter()
# employment status is active

hr_filter = hrdata %>% filter(EmploymentStatus == "Active")
View(hr_filter)

# to filter where perfomance score are not fully meets and exceed

hrdata$PerformanceScore

hr_filter2 = hrdata %>% filter(!PerformanceScore == "Fully Meets"
                               & !PerformanceScore == "Exceeds")
View(hr_filter2)
```

```
# to filter PS is PIP or status is active

hr_filter3 = hrdata %>% filter(PerformanceScore == "PIP"
                               | EmploymentStatus == "Active")
View(hr_filter3)

# 3. arrange()
# to arrange pay rate in asc. order

hr_arrange = hrdata %>% arrange(PayRate)
View(hr_arrange)

# pay rate > 50 and arrange in dsc order

hr_arrange2 = hrdata %>% filter(PayRate>50) %>% arrange(desc(PayRate))
View(hr_arrange2)

# Select the columns Employee name, emp id, pay rate, performance score,
employee satisfaction
## and special projects counts
## Filter the data where pay rate is lower than 50 and employee satisfaction
is less than 3
## find out the who has the lowest pay rate

hr_data1 = hrdata %>% select(Employee_Name,EmpID,PayRate,PerformanceScore,
                             EmpSatisfaction,SpecialProjectsCount) %>%
  filter(PayRate<50 & EmpSatisfaction<3) %>% arrange(PayRate) %>%
  nrow()
hr_data1
View(hr_data1)

# New Dataset: Superstore
library(readxl)
store = read_excel("D:/RISE - WPU/R Studio/Sample-Superstore-Subset-
Excel.xlsx")
View(store)

## PS: Give me the row ID where order priority is not
#cricital, customer segment is business,
## product container is wrap bag and the sales is the highest

store_data = store %>% select(`Row ID`, `Order Priority`, `Customer Segment`,
`Product Container`,Sales) %>%
  filter(!`Order Priority` == "Critical" & `Customer Segment` == "Small
Business" & `Product Container` == "Wrap Bag") %>%
  arrange(desc(Sales))
View(store_data)
```