CG2CG2-ACTIVITY01

CG2CG2

PART 1: LOAD DATA

```
setwd("C:/Users/sethd/Documents/CS3203N/CG2CG2_ACT1")

library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
edata <- read.csv("employee_data.csv")

edata
```

```
##   EmployeeID Name Department Salary YearsWorked Performance
## 1        101  Ana      Sales  55000           6           4
## 2        102  Ben         IT  62000           4           5
## 3        103 Cory         HR  48000           5          NA
## 4        104 Dina         IT  59000           3           3
## 5        105  Eli      Sales  50000           7           5
## 6        106 Faye         HR  47000           2          NA
## 7        107 Gary         IT  63000           8           4
## 8        108 Hana      Sales  51000           1           2
```

```
head(edata)
```

```
##   EmployeeID Name Department Salary YearsWorked Performance
## 1        101  Ana      Sales  55000           6           4
## 2        102  Ben         IT  62000           4           5
## 3        103 Cory         HR  48000           5          NA
## 4        104 Dina         IT  59000           3           3
## 5        105  Eli      Sales  50000           7           5
## 6        106 Faye         HR  47000           2          NA
```

```r
colSums(is.na(edata))
```

```
##  EmployeeID        Name  Department      Salary YearsWorked Performance
##           0           0           0           0           0           2
```

```r
edata %>% filter(is.na(Performance))
```

```
##   EmployeeID Name Department Salary YearsWorked Performance
## 1        103 Cory         HR  48000           5          NA
## 2        106 Faye         HR  47000           2          NA
```

Based on the first six rows of the dataset, the *Performance* variable contains two missing values (NA).

PART 2: SELECT AND FILTER

```r
edata_sel <- edata %>% select(Name, Department, Salary)
edata_sel
```

Part 2(a): Select Name, Department, Salary

```
##   Name Department Salary
## 1  Ana      Sales  55000
## 2  Ben         IT  62000
## 3 Cory         HR  48000
## 4 Dina         IT  59000
## 5  Eli      Sales  50000
## 6 Faye         HR  47000
## 7 Gary         IT  63000
## 8 Hana      Sales  51000
```

```r
edata_sel_base <- data.frame(
  Name = edata$Name,
  Department = edata$Department,
  Salary = edata$Salary
)
edata_sel_base
```

```
##   Name Department Salary
## 1  Ana      Sales  55000
## 2  Ben         IT  62000
## 3 Cory         HR  48000
## 4 Dina         IT  59000
## 5  Eli      Sales  50000
## 6 Faye         HR  47000
## 7 Gary         IT  63000
## 8 Hana      Sales  51000
```

```r
edata_high_salary <- edata %>% filter(Salary > 55000)
edata_high_salary
```

Part 2(b): Filter Salary > 55,000

```
##   EmployeeID Name Department Salary YearsWorked Performance
## 1        102  Ben         IT  62000           4           5
## 2        104 Dina         IT  59000           3           3
## 3        107 Gary         IT  63000           8           4
```

PART 3: MUTATE AND ARRANGE

```r
edata_senior <- edata %>% mutate(Seniority = YearsWorked > 5)
edata_senior
```

Part 3(a): Create Seniority column

```
##   EmployeeID Name Department Salary YearsWorked Performance Seniority
## 1        101  Ana      Sales  55000           6           4      TRUE
## 2        102  Ben         IT  62000           4           5     FALSE
## 3        103 Cory         HR  48000           5          NA     FALSE
## 4        104 Dina         IT  59000           3           3     FALSE
## 5        105  Eli      Sales  50000           7           5      TRUE
## 6        106 Faye         HR  47000           2          NA     FALSE
## 7        107 Gary         IT  63000           8           4      TRUE
## 8        108 Hana      Sales  51000           1           2     FALSE
```

```r
edata_sorted_perf <- edata %>% arrange(desc(Performance))
edata_sorted_perf
```

Part 3(b): Arrange by Performance (descending)

```
##   EmployeeID Name Department Salary YearsWorked Performance
## 1        102  Ben         IT  62000           4           5
## 2        105  Eli      Sales  50000           7           5
## 3        101  Ana      Sales  55000           6           4
## 4        107 Gary         IT  63000           8           4
## 5        104 Dina         IT  59000           3           3
## 6        108 Hana      Sales  51000           1           2
## 7        103 Cory         HR  48000           5          NA
## 8        106 Faye         HR  47000           2          NA
```

Part 3(c): Describe the last output Employees are sorted from highest to lowest performance rating. Rows with missing performance values appear at the bottom.

PART 4: GROUPED OPERATIONS

```
dept_avg_salary <- edata %>%
  group_by(Department) %>%
  summarise(MeanSalary = mean(Salary), .groups = "drop")
dept_avg_salary
```

Part 4(a): Average salary by Department

```
## # A tibble: 3 x 2
##   Department MeanSalary
##   <chr>           <dbl>
## 1 HR              47500
## 2 IT              61333.
## 3 Sales           52000
```

```
aggregate(Salary ~ Department, data = edata, mean)
```

```
##   Department   Salary
## 1         HR 47500.00
## 2         IT 61333.33
## 3      Sales 52000.00
```

```
dept_counts <- edata %>%
  group_by(Department) %>%
  summarise(EmployeeCount = n(), .groups = "drop")
dept_counts
```

Part 4(b): Count employees per Department

```
## # A tibble: 3 x 2
##   Department EmployeeCount
##   <chr>              <int>
## 1 HR                     2
## 2 IT                     3
## 3 Sales                  3
```

Part 4(c): Interpretation The IT department has the highest average salary. The IT and Sales departments have the most employees, with three employees each.