**OUTPUT**

A graph of different colored squares

Description automatically generated

library(readxl)

> library(dplyr)

> library(tidyr) # For drop\_na()

> library(ggplot2) # For plotting

> # 1. Data cleaning: Load data and remove invalid entries

> national\_salaries <- read\_excel("NationalSalaries.xlsx")

> national\_salaries <- national\_salaries %>% select(-c(GROUP, EMP\_PRSE, MEAN\_PRSE, H\_PCT10, H\_PCT25, H\_MEDIAN, H\_PCT75, H\_PCT90, A\_PCT10, A\_PCT25, A\_MEDIAN, A\_PCT75,A\_PCT90, ANNUAL, HOURLY))

> clean\_data<-na.omit(national\_salaries)

> print("Original NationalSalaries data:")

[1] "Original NationalSalaries data:"

> head(clean\_data)

# A tibble: 6 × 8

AREA ST STATE OCC\_CODE OCC\_TITLE TOT\_EMP H\_MEAN A\_MEAN

*<chr>* *<chr>* *<chr>* *<chr>* *<chr>* *<chr>* *<chr>* *<chr>*

1 02 AK Alaska 29-1061 Anesthesiologists \*\* # #

2 05 AR Arkansas 29-1023 Orthodontists \*\* # #

3 12 FL Florida 29-1023 Orthodontists \*\* # #

4 16 ID Idaho 29-1066 Psychiatrists \*\* # #

5 24 MD Maryland 29-1022 Oral and maxillofacial s… \*\* # #

6 27 MN Minnesota 29-1023 Orthodontists \*\* # #

> # 2. Select columns from Salaries.xlsx and keep only those columns in the cleaned data

> salaries <- read\_excel("Salaries.xlsx")

There were 50 or more warnings (use warnings() to see the first 50)

> salaries <- salaries %>% select(-c(Group))

> head(salaries)

# A tibble: 6 × 8

ID State StateName JobCode JobName TotalEmployment AverageHourlySalary

*<dbl>* *<chr>* *<chr>* *<chr>* *<chr>* *<dbl>* *<dbl>*

1 7807 GU Guam 35-3041 Food s… 30 6

2 7800 GU Guam 51-3022 Meat, … 30 7

3 28287 PR Puerto Rico 21-2099 Religi… 30 7

4 7781 GU Guam 49-2011 Comput… 30 8

5 33307 VI Virgin Islands 51-9022 Grindi… 30 8

6 26164 OK Oklahoma 39-6032 Transp… 30 9

# ℹ 1 more variable: AverageYearlySalary <dbl>

> clean\_data <- clean\_data %>%

+ rename(ID = "AREA",

+ State = "ST",

+ StateName = "STATE",

+ JobCode = "OCC\_CODE",

+ JobName = "OCC\_TITLE",

+ TotalEmployment = "TOT\_EMP",

+ AverageHourlySalary = "H\_MEAN",

+ AverageYearlySalary = "A\_MEAN")

> common\_columns <- intersect(colnames(clean\_data), colnames(salaries))

> selected\_columns <- clean\_data %>% select(all\_of(common\_columns))

> print("Selected columns from clean\_data that match salaries:")

[1] "Selected columns from clean\_data that match salaries:"

> print(head(selected\_columns))

# A tibble: 6 × 8

ID State StateName JobCode JobName TotalEmployment AverageHourlySalary

*<chr>* *<chr>* *<chr>* *<chr>* *<chr>* *<chr>* *<chr>*

1 02 AK Alaska 29-1061 Anesthesiol… \*\* #

2 05 AR Arkansas 29-1023 Orthodontis… \*\* #

3 12 FL Florida 29-1023 Orthodontis… \*\* #

4 16 ID Idaho 29-1066 Psychiatris… \*\* #

5 24 MD Maryland 29-1022 Oral and ma… \*\* #

6 27 MN Minnesota 29-1023 Orthodontis… \*\* #

# ℹ 1 more variable: AverageYearlySalary <chr>

> #selected\_columns <- subset(selected\_columns, TotalEmployment !="\*\*", AverageHourlySalary !="\*",AverageHourlySalary !="#", AverageYearlySalary !="\*", AverageYearlySalary !="#")

> selected\_columns <- selected\_columns %>%

+ filter(TotalEmployment != "\*\*",

+ AverageHourlySalary != "\*",

+ AverageHourlySalary != "#",

+ AverageYearlySalary != "\*",

+ AverageYearlySalary != "#")

> print(selected\_columns)

# A tibble: 32,486 × 8

ID State StateName JobCode JobName TotalEmployment AverageHourlySalary

*<chr>* *<chr>* *<chr>* *<chr>* *<chr>* *<chr>* *<chr>*

1 20 KS Kansas 41-2012 Gaming… 100 10.13

2 35 NM New Mexico 37-3012 Pestic… 100 10.21

3 55 WI Wisconsin 39-3019 Gaming… 100 10.29

4 78 VI Virgin Islan… 43-2011 Switch… 100 10.31

5 56 WY Wyoming 47-3016 Helper… 100 10.39

6 40 OK Oklahoma 51-9193 Coolin… 100 10.48

7 78 VI Virgin Islan… 21-1093 Social… 100 10.66

8 56 WY Wyoming 13-2082 Tax pr… 100 10.73

9 20 KS Kansas 51-4052 Pourer… 100 10.78

10 16 ID Idaho 51-4035 Millin… 100 11.02

# ℹ 32,476 more rows

# ℹ 1 more variable: AverageYearlySalary <chr>

# ℹ Use `print(n = ...)` to see more rows

> selected\_columns <- selected\_columns %>%

+ mutate(

+ TotalEmployment = as.integer(TotalEmployment),

+ AverageHourlySalary = as.numeric(AverageHourlySalary),

+ AverageYearlySalary = as.numeric(AverageYearlySalary)

+ )

> print(selected\_columns)

# A tibble: 32,486 × 8

ID State StateName JobCode JobName TotalEmployment AverageHourlySalary

*<chr>* *<chr>* *<chr>* *<chr>* *<chr>* *<int>* *<dbl>*

1 20 KS Kansas 41-2012 Gaming… 100 10.1

2 35 NM New Mexico 37-3012 Pestic… 100 10.2

3 55 WI Wisconsin 39-3019 Gaming… 100 10.3

4 78 VI Virgin Islan… 43-2011 Switch… 100 10.3

5 56 WY Wyoming 47-3016 Helper… 100 10.4

6 40 OK Oklahoma 51-9193 Coolin… 100 10.5

7 78 VI Virgin Islan… 21-1093 Social… 100 10.7

8 56 WY Wyoming 13-2082 Tax pr… 100 10.7

9 20 KS Kansas 51-4052 Pourer… 100 10.8

10 16 ID Idaho 51-4035 Millin… 100 11.0

# ℹ 32,476 more rows

# ℹ 1 more variable: AverageYearlySalary <dbl>

# ℹ Use `print(n = ...)` to see more rows

> # Save the result to a new file for later use

> write.csv(selected\_columns, "selected\_columns.csv")

> # 3. Randomly select 1500 rows

> print(paste("Number of rows in selected\_columns:", nrow(selected\_columns)))

[1] "Number of rows in selected\_columns: 32486"

> random\_sample <- selected\_columns %>% sample\_n(1500)

> print("Randomly selected 1500 rows:")

[1] "Randomly selected 1500 rows:"

> print(head(random\_sample))

# A tibble: 6 × 8

ID State StateName JobCode JobName TotalEmployment AverageHourlySalary

*<chr>* *<chr>* *<chr>* *<chr>* *<chr>* *<int>* *<dbl>*

1 23 ME Maine 51-9071 Jewelers an… 220 16.3

2 10 DE Delaware 21-1029 Social work… 620 20.4

3 31 NE Nebraska 51-2031 Engine and … 490 12.6

4 17 IL Illinois 49-9091 Coin, vendi… 1590 14.3

5 50 VT Vermont 51-5011 Bindery wor… 130 13.8

6 18 IN Indiana 47-4011 Constructio… 1140 19.7

# ℹ 1 more variable: AverageYearlySalary <dbl>

> # 4. Create a data frame with jobs having average hourly salary < 15

> low\_salary\_jobs <- selected\_columns %>% filter(AverageHourlySalary < 15)

> print("Jobs with average hourly salary less than 15:")

[1] "Jobs with average hourly salary less than 15:"

> print((low\_salary\_jobs))

# A tibble: 11,206 × 8

ID State StateName JobCode JobName TotalEmployment AverageHourlySalary

*<chr>* *<chr>* *<chr>* *<chr>* *<chr>* *<int>* *<dbl>*

1 20 KS Kansas 41-2012 Gaming… 100 10.1

2 35 NM New Mexico 37-3012 Pestic… 100 10.2

3 55 WI Wisconsin 39-3019 Gaming… 100 10.3

4 78 VI Virgin Islan… 43-2011 Switch… 100 10.3

5 56 WY Wyoming 47-3016 Helper… 100 10.4

6 40 OK Oklahoma 51-9193 Coolin… 100 10.5

7 78 VI Virgin Islan… 21-1093 Social… 100 10.7

8 56 WY Wyoming 13-2082 Tax pr… 100 10.7

9 20 KS Kansas 51-4052 Pourer… 100 10.8

10 16 ID Idaho 51-4035 Millin… 100 11.0

# ℹ 11,196 more rows

# ℹ 1 more variable: AverageYearlySalary <dbl>

# ℹ Use `print(n = ...)` to see more rows

> # 5. Create a data frame with Indiana jobs and bin yearly salaries into 10 intervals

> indiana\_jobs <- random\_sample %>% filter(State == "IN")

> indiana\_jobs <- indiana\_jobs %>%

+ mutate(Salary\_Bin = cut(AverageYearlySalary, breaks = 10))

> # Print the first few rows of the indiana\_jobs data frame

> print("Indiana jobs with yearly salary bins:")

[1] "Indiana jobs with yearly salary bins:"

> print(head(indiana\_jobs))

# A tibble: 6 × 9

ID State StateName JobCode JobName TotalEmployment AverageHourlySalary

*<chr>* *<chr>* *<chr>* *<chr>* *<chr>* *<int>* *<dbl>*

1 18 IN Indiana 47-4011 Constructio… 1140 19.7

2 18 IN Indiana 51-4072 Molding, co… 8580 14.4

3 18 IN Indiana 17-2081 Environment… 520 31.9

4 18 IN Indiana 29-1199 Health diag… 870 41.4

5 18 IN Indiana 47-2121 Glaziers 1010 17.3

6 18 IN Indiana 27-2041 Music direc… 130 18.9

# ℹ 2 more variables: AverageYearlySalary <dbl>, Salary\_Bin <fct>

> # 6. Find total employment for each state

> total\_employment <- selected\_columns %>%

+ group\_by(State) %>%

+ summarise(TotalEmployment = sum(TotalEmployment, na.rm = TRUE))

> print("Total employment by state:")

[1] "Total employment by state:"

> print((total\_employment))

# A tibble: 54 × 2

State TotalEmployment

*<chr>* *<int>*

1 AK 868540

2 AL 5681050

3 AR 3370120

4 AZ 7777640

5 CA 44630120

6 CO 6627910

7 CT 4909250

8 DC 1801500

9 DE 1230770

10 FL 23479790

# ℹ 44 more rows

# ℹ Use `print(n = ...)` to see more rows

> # 7. Find the average yearly salary of jobs in Indiana

> avg\_indiana\_salary <- indiana\_jobs %>% summarise(Average\_Salary = mean(AverageYearlySalary))

> print("Average yearly salary in Indiana:")

[1] "Average yearly salary in Indiana:"

> print(avg\_indiana\_salary)

# A tibble: 1 × 1

Average\_Salary

*<dbl>*

1 42317.

> # 8. Compare average yearly salaries of 'Computer and mathematical occupations' across three states

> computer\_jobs <- selected\_columns %>% filter(grepl("^15", JobCode))

> selected\_states <- computer\_jobs %>% filter(State %in% c("IN", "CA", "NY"))

> print("Computer and mathematical occupations")

[1] "Computer and mathematical occupations"

> print(selected\_states)

# A tibble: 46 × 8

ID State StateName JobCode JobName TotalEmployment AverageHourlySalary

*<chr>* *<chr>* *<chr>* *<chr>* *<chr>* *<int>* *<dbl>*

1 36 NY New York 15-2021 Mathemati… 100 41.9

2 18 IN Indiana 15-1011 Computer … 100 43.2

3 06 CA California 15-1061 Database … 11480 36.2

4 18 IN Indiana 15-1099 Computer … 1350 31.0

5 36 NY New York 15-1081 Network s… 17210 37.5

6 18 IN Indiana 15-1061 Database … 1760 29.2

7 36 NY New York 15-2011 Actuaries 1760 47.3

8 36 NY New York 15-1032 Computer … 17960 46.2

9 36 NY New York 15-2091 Mathemati… 180 24.1

10 36 NY New York 15-1071 Network a… 18840 37.3

# ℹ 36 more rows

# ℹ 1 more variable: AverageYearlySalary <dbl>

# ℹ Use `print(n = ...)` to see more rows

> # Create a bar chart comparing the average salaries

> ggplot(selected\_states, aes(x=State, y=AverageYearlySalary, fill=State)) +

+ geom\_bar(stat="identity") +

+ labs(title="Comparison of Computer and Mathematical Occupations") +

+ theme\_minimal()

A graph of different colored squares

Description automatically generated