Name (Last, First):

Student ID:

Assignment 2

METCS544A3A4_F2024

Instructions:

1. For answering programming questions, please use Adobe Acrobat to edit the pdf file in two steps **[See Appendix: Example Question and Answer]**:
    a. Copy and paste your R code as text in the box provided (so that your teaching team can run your code);
    b. Screenshot your R console outputs, save them as a .PNG image file, and paste/insert them in the box provided.
    c. Show all work - credit will not be given for code without showing the code in action by including the screenshot of R console outputs.
2. To answer non-programming questions, please type or handwrite your final answers clearly in the boxes. Show all work - credit will not be given for numerical solutions that appear without explanation in the space above the boxes.
3. **[Total 87 pts = 21 + 9 + 42 + 15 Extra Credit pts]**

Grading Rubric

Each question is worth 3 points and will be graded as follows:

3 points: Correct answer with work shown

2 points: Incorrect answer but attempt shows some understanding (work shown)

1 point: Incorrect answer but an attempt was made (work shown), or **correct answer without explanation (work not shown)**

0 points: Left blank or made little to no effort/work not shown

Reflective Journal [3 pts]

(Copy and paste the link to your live Google doc in the box below)

Part I: Descriptive Statistics

1. **Interpreting ratios [21 pts]:** A total of 1824 adults were asked the following question: "Do you favor or oppose the death penalty for persons convicted of murder?" The following table compares the adults' responses with their ethnicities.

	African American	Caucasian	Other	Total
Favor	128	953	108	1189
Oppose	140	414	81	635
Total	268	1367	189	1824

Source: *General Social Survey*

a. Find the number of Caucasians in the survey who oppose the death penalty.

Answer:

b. Find the number of African Americans in the survey who oppose the death penalty.

Answer:

c. For the Caucasians in the study, find the unit ratio of the number who oppose the death penalty to the number who are in favor.

Answer:

Caucasians in favor

Caucasians against

d. For the African Americans in the study, find the unit ratio of the number who oppose the death penalty to the number who are in favor.

Answer:

African American in favor

African American against

e. A student says that Caucasians in the study are more likely to oppose the death penalty than African Americans in the study because more Caucasians oppose the death penalty than African Americans. What would you tell the student?

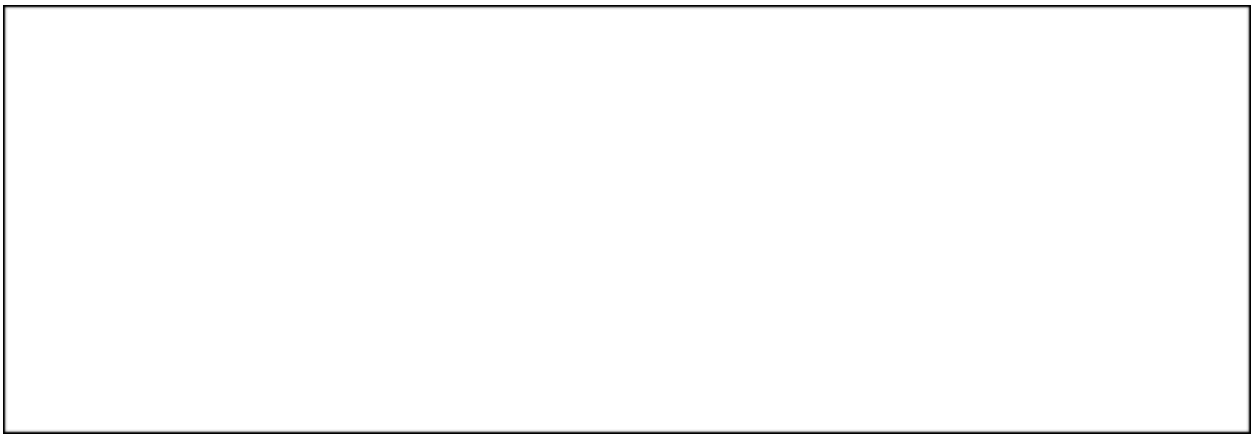
Answer:

2. **Graphing categorical data (9 pts):** Below is a list of unorganized data from the 2020 election. This gives the state, their geographical location, and the voting majority. [The data is accessible via a .csv file saved under the course shared folder "Course Contents".]

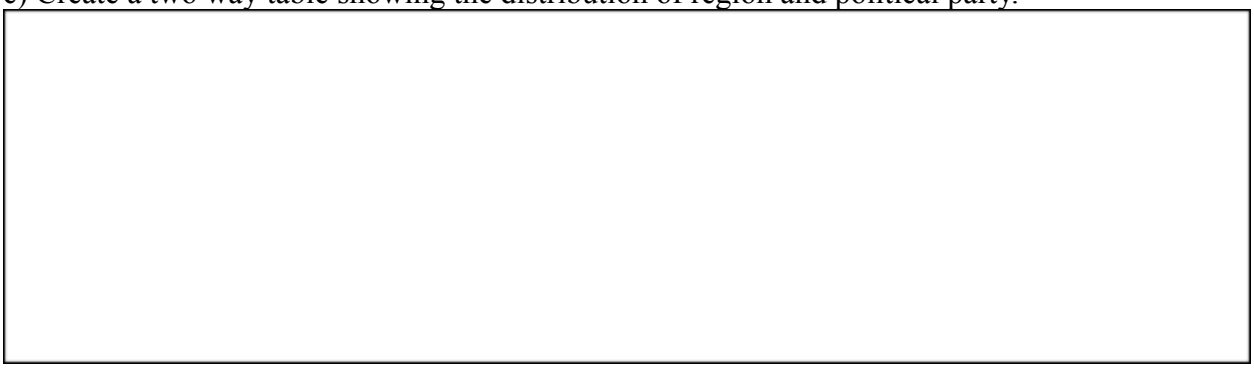
State	*Location*	*Electoral College Vote in 2020*
AL	South	Republican
AK	West	Republican
AZ	West	Democrat
AR	South	Republican
CA	West	Democrat
CO	West	Democrat
CT	Northeast	Democrat
DE	South	Democrat
FL	South	Republican
GA	South	Democrat
HI	West	Democrat
ID	West	Republican
IL	Midwest	Democrat
IN	Midwest	Republican
IA	Midwest	Republican
KS	Midwest	Republican
KY	South	Republican
LA	South	Republican
ME	Northeast	Democrat
MD	South	Democrat
MA	Northeast	Democrat
MI	Midwest	Democrat
MN	Midwest	Democrat
MS	South	Republican
MO	Midwest	Republican
MT	West	Republican
NE	Midwest	Republican
NV	West	Democrat
NH	Northeast	Democrat
NJ	Northeast	Democrat
NM	West	Democrat
NY	Northeast	Democrat
NC	South	Republican
ND	Midwest	Republican
OH	Midwest	Republican
OK	South	Republican
OR	West	Democrat
PA	Northeast	Democrat
RI	Northeast	Democrat
SC	South	Republican
SD	Midwest	Republican
TN	South	Republican
TX	South	Republican
UT	West	Republican
VT	Northeast	Democrat
VA	South	Democrat
WA	West	Democrat
WV	South	Republican
WI	Midwest	Democrat
WY	West	Republican

a) Construct a bar graph of the amount of states in each region of the US.

b) Construct a bar graph of the amount of states who voted either republican or democrat in the 2020 election.



c) Create a two way table showing the distribution of region and political party.



Part II Statistical Programming (54 pts, including 12 Extra Credits)

The following sample data shows the scores of the students in an exam:

68, 25, 87, 89, 91, 79, 99, 80, 62, 74

Do the following using R code **with only a single expression (one-liner)** for each question. The solutions should be generic and work for any data size **(meaning no hard coding)**. You can assume there will be an even number of values in the given data.

1. **Assigning and Accessing Data (15 pts):**
    a. Assign the above data as a *vector* in the same order to the variable ***scores***. Use this variable for the remaining problems.
    b. Using the *length* function, compute how many students took the exam? Store the expression in the variable ***n*.**
    c. Using indexing, write the expression for accessing the first two items. Store the expression in the variable ***first_and_second*.**
    d. Using indexing, write the expression for accessing the first and last items. Store the expression in the variable ***first_and_last.***
    e. Using indexing, write the expression for accessing the middle two items. Store the expression in the variable ***middle_two.***

Sample output:

```
[1] 10
[1] 68 25
[1] 68 74
[1] 91 79
```

2. **Median (15 pts)**
    a. Use *median(scores)* to compute the median of the data. Store the expression in the variable **median_score**.
    b. Using comparison operators, write the R expression for scores less than or equal to the median of the data. Store the expression in the variable **below_median.**
    c. Using comparison operators, write the R expression for scores greater than the median of the data. Store the expression in the variable **above_median.**
    d. Using the **sum** function, write the R expression for the number of scores less than or equal to the median of the data. Store the expression in the variable **count_below_median.**
    e. Using the **sum** function, write the R expression for the number of scores greater than the median of the data. Store the expression in the variable **count_above_median.**

Sample output:

```
[1] 79.5
[1]  TRUE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE
[1] FALSE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE FALSE
[1] 5
[1] 5
```

3. **Indexing and subsetting (12 pts)**

   a. Using logical indexing and the results from Q2), write the R expression for all the scores that are less than or equal to the median value of the data. Store the expression in the variable **scores_below_median.**
   b. Similarly, write the R expression for all the scores that are greater than the median. Store the expression in the variable **scores_above_median.**

Sample output:
[1] 68 25 79 62 74
[1] 87 89 91 99 80

Use the **seq** function to generate the numeric indices for (c) and (d) below.

c. Using numeric indexing, write the R expression for the odd indexed values from the scores. Store the expression in the variable **odd_index_values**.
d. Similarly, write the R expression for the even indexed values from the scores. Store the expression in the variable **even_index_values**.

Sample output:
[1] 68 87 91 99 62
[1] 25 89 79 80 74

a. Using the **paste** function with LETTERS, write the expression for the following output. Store the expression in the variable **format_scores_version1**. You can assume there are no more than 26 values.

Sample Output:
[1] "A=68" "B=25" "C=87" "D=89" "E=91" "F=79" "G=99" "H=80" "I=62" "J=74"

b. Similarly, using the **paste** function with LETTERS, write the expression for the following output. Store the expression in the variable **format_scores_version2**.

Sample output:
[1] "J=68" "I=25" "H=87" "G=89" "F=91" "E=79" "D=99" "C=80" "B=62" "A=74"

4. **Extra Credit: Matrix and named matrix (12 pts)**

   a. Create a matrix with two rows using the ***scores*** data. The first half of the values belong to the first row of the matrix. Store the expression in the variable **scores_matrix.**
   The code should work for any size input data.
   You can assume that there are an even number of values in scores.

Sample output:

```
     [,1] [,2] [,3] [,4] [,5]
[1,]   68   25   87   89   91
[2,]   79   99   80   62   74
```

   b. Write the expression for displaying the first and last columns of the above matrix. The code should work for any size matrix. Store the expression in the variable **first_and_last_version1.**

Sample output:

```
     [,1] [,2]
[1,]   68   91
[2,]   79   74
```

c. Copy **scores_matrix** to the variable **named_matrix**.
Assign column names for the *named_matrix* as Student_1, Student_2,… and row names as Quiz_1, Quiz_2, ... The code should work for any size matrix, i.e., for any number of columns in the matrix and any number of rows.

Sample output:

```
       Student_1 Student_2 Student_3 Student_4 Student_5
Quiz_1        68        25        87        89        91
Quiz_2        79        99        80        62        74
```

d. Show the result for displaying the first and last columns of the *named_matrix*. The code should work for any size matrix. Store the expression in the variable **first_and_last_version2.**

Sample output:

```
       Student_1 Student_5
Quiz_1        68        91
Quiz_2        79        74
```

Answer: Copy and paste your R code in the box below (not an image but the text).

Screenshot of your R console outputs and paste the image in the box below

Appendix: Example Question and Answer for R programming questions:

Calculate the sum $\sum_{j=0}^{n} r^j$, where r has been assigned the value 1.08, and compare with $(1-r^{n+1})/(1-r)$, for $n = 10, 20, 30, 40$.

Answer: Copy and paste your R code in the box below (not an image but the text).

```
r <- 1.08
n <- c(10, 20, 30, 40)
sum1 <- c()
for(i in n){
  x <- 0:i
  sum1 <- c(sum1, sum(r^x))
}
sum1   # This gives the calculated sums for n = 10, 20, 30, 40.

sum2 <- (1 - r^(n + 1)) / (1 - r)
sum2

sum2 - sum1   # The formula works.
```

Screenshot of your R console outputs and paste the image in the box below

```
> r <- 1.08
> n <- c(10, 20, 30, 40)
> sum1 <- c()
> for(i in n){
+   x <- 0:i
+   sum1 <- c(sum1, sum(r^x))
+ }
> sum1   # This gives the calculated sums for n = 10, 20, 30, 40.
[1]  16.64549  50.42292 123.34587 280.78104
> sum2 <- (1 - r^(n + 1)) / (1 - r)
> sum2
[1]  16.64549  50.42292 123.34587 280.78104
> sum2 - sum1   # The formula works.
[1] 0 0 0 0
```

THE END