Question 1

a. I used the read.table() command and set header to False since there are no column headings in the data.

```
rain_df <- read.table("data/rnf6080.dat", header = FALSE)
```

b. There are 5070 rows and 27 columns in rain_df.

```
nrow(rain_df)
```

```
## [1] 5070
```

```
ncol(rain_df)
```

```
## [1] 27
```

c.

```
colnames(rain_df)
```

```
##  [1] "V1"  "V2"  "V3"  "V4"  "V5"  "V6"  "V7"  "V8"  "V9"  "V10" "V11" "V12"
## [13] "V13" "V14" "V15" "V16" "V17" "V18" "V19" "V20" "V21" "V22" "V23" "V24"
## [25] "V25" "V26" "V27"
```

d. The value is 0.

```
rain_df[2, 4]
```

```
## [1] 0
```

e.

```
rain_df[2, ]
```

```
##   V1 V2 V3 V4 V5 V6 V7 V8 V9 V10 V11 V12 V13 V14 V15 V16 V17 V18 V19 V20 V21
## 2 60  4  2  0  0  0  0  0  0   0   0   0   0   0   0   0   0   0   0   0   0
##   V22 V23 V24 V25 V26 V27
## 2   0   0   0   0   0   0
```

f. The following command renames the column headings of rain_df. The first, second, and third columns are named year, month, and day respectively. The remaining 24 columns are renamed with sequential numbers from 0 to 23.

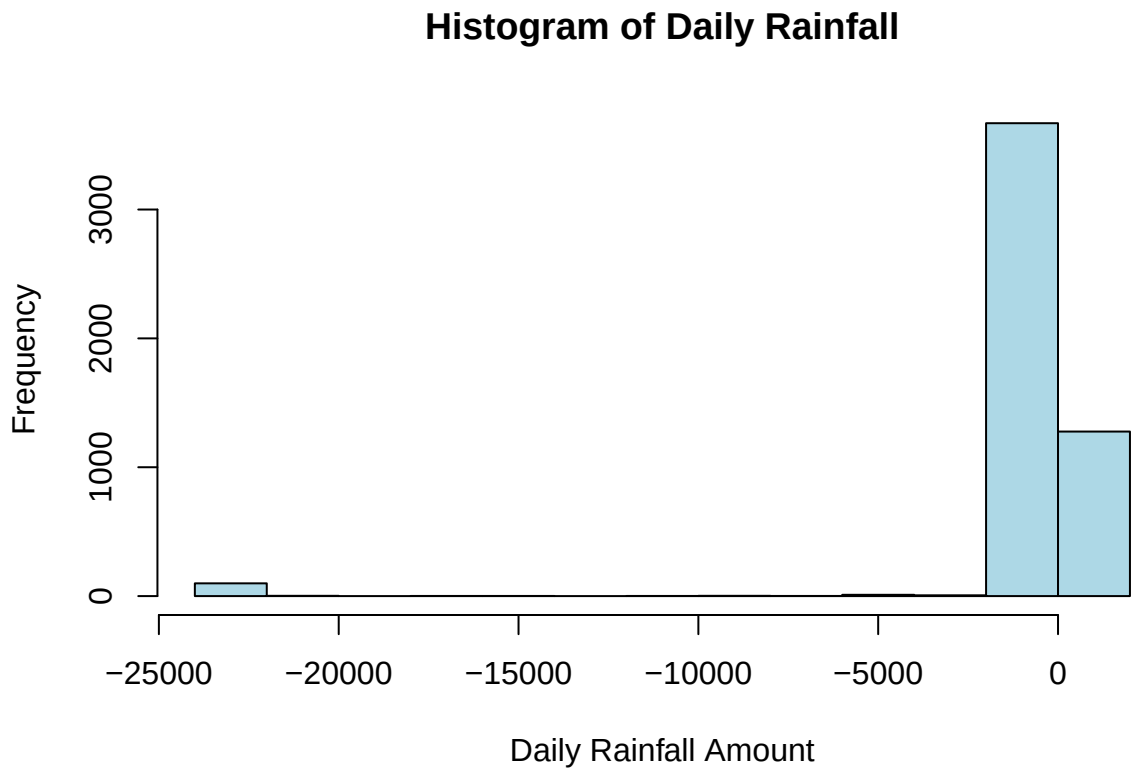
```
names(rain_df) <- c("year","month","day",seq(0,23))
```

g.

```r
rain_df$daily_rain_fall <- rowSums(rain_df[, 4:27])
```

h.

```r
hist(rain_df$daily_rain_fall, main = "Histogram of Daily Rainfall",
     xlab = "Daily Rainfall Amount", ylab = "Frequency", col = "lightblue", border = "black")
```



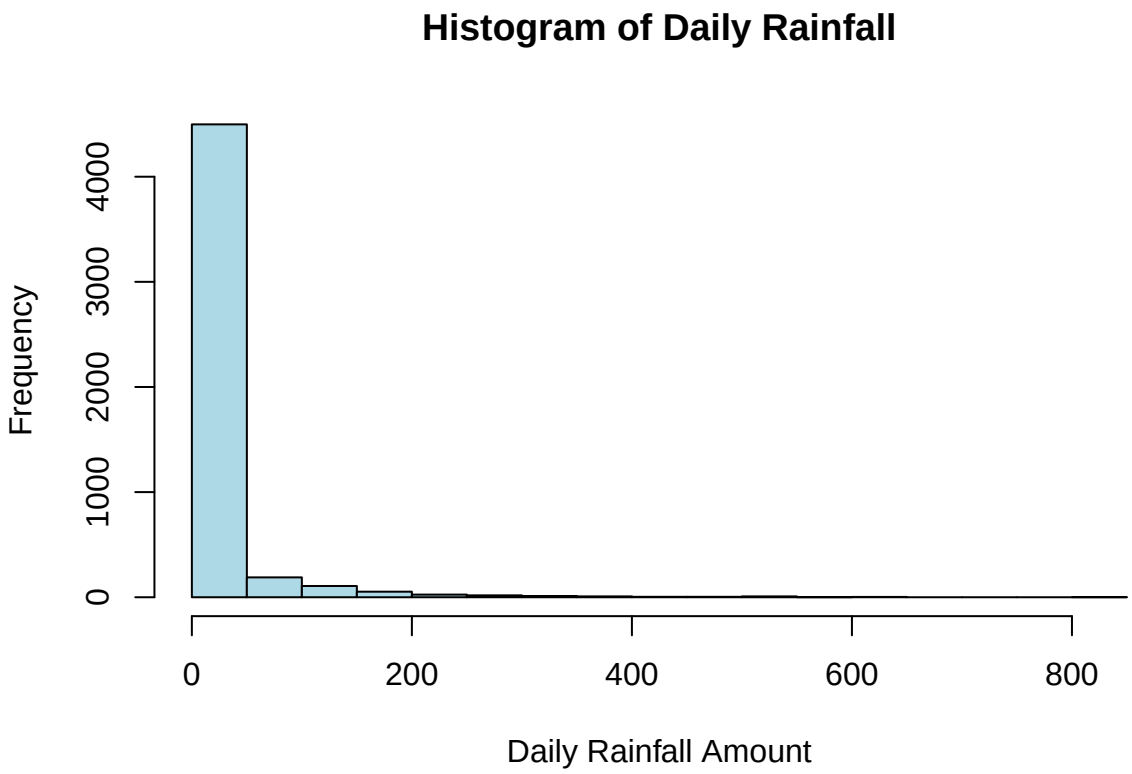
i. Rainfall amounts cannot be negative, so the presence of negative values in the histogram indicates that it cannot be right.

j.

```r
rain_df$corrected_daily_rain_fall <- rain_df$daily_rain_fall
rain_df$corrected_daily_rain_fall[rain_df$daily_rain_fall < 0] <- NA
```

k. This histogram is more reasonable since the daily rainfall values are within an expected positive range. The irregularities are filtered out of the graph.

```r
hist(rain_df$corrected_daily_rain_fall, main = "Histogram of Daily Rainfall",
     xlab = "Daily Rainfall Amount", ylab = "Frequency", col = "lightblue", border = "black")
```



Question 2

a. This does not produce an error because the max() function treats the values as characters rather than numbers and performs a lexicographic comparison between them. This is not the expected value.

```
x <- c("5", "12", "7")
max(x)
```

```
## [1] "7"
```

The sort() function can handle character vectors. When applied to a character vector, it sorts the values lexicographically. This is not the expected value.

```
sort(x)
```

```
## [1] "12" "5"  "7"
```

ERROR: The max() function expects numerical input, but x is a character vector. Hence, there is an error.

```
#sum(x)
```

b. In the first command, R attempts to combine different data types into a single vector. So, it coerces all elements to the most flexible data type which is a character vector. In this process, 7 and 12 become "7" and "12".

ERROR: In the second step, there is an error in attempting to perform an arithmetic operation on character data.

```
y <- c("5",7,12)
#y[2] + y[3]
```

c. In the first command, R attempts to create a data frame where each column has a single data type. So, z1 containing "5" becomes a character column while z2 and z3 are numeric.

In step 2, values from the first row, second column (here, 7) and first row, third column (here, 12) are extracted and a successful arithmetic addition is performed on them to return 19.

```
z <- data.frame(z1="5",z2=7,z3=12)
z[1,2] + z[1,3]
```

```
## [1] 19
```

Question 3

a. Reproducible code ensures that results can be consistently replicated. If others run the same code with the same inputs, they should obtain the same results, ensuring accuracy and reliability in findings.Reproducible code also makes it easier to verify results, and identify and fix bugs. In research, reproducibility enhances transparency by allowing others to follow the same methods, building trust in the findings. Furthermore, the code can be reused and built upon for future work.

b. During the class, I may encounter unexpected results or errors. If my code is not reproducible, it can be difficult to trace and fix these issues. Reproducible code will help me explain my work to TAs for any questions as well as refer back to it and understand methodology when preparing for exams.

c. 3