WEEK2-3.R

shadankhan

2024-08-13

```
# Define the string
info_string <- "Name: Shadan Khan, Unit: Statistical Data Analysis, Task: Probability and Distributions

# Print the string
print(info_string)
```

```
## [1] "Name: Shadan Khan, Unit: Statistical Data Analysis, Task: Probability and Distributions Week 2-
```

```
#Question 2
data <- read.csv("weather (2).csv")

#Question 3
head(data)
```

```
##   origin year month day hour  temp  dewp humid wind_dir wind_speed wind_gust
## 1    EWR 2013     1   1    1 39.02 26.06 59.37      270   10.35702        NA
## 2    EWR 2013     1   1    2 39.02 26.96 61.63      250    8.05546        NA
## 3    EWR 2013     1   1    3 39.02 28.04 64.43      240   11.50780        NA
## 4    EWR 2013     1   1    4 39.92 28.04 62.21      250   12.65858        NA
## 5    EWR 2013     1   1    5 39.02 28.04 64.43      260   12.65858        NA
## 6    EWR 2013     1   1    6 37.94 28.04 67.21      240   11.50780        NA
##   precip pressure visib            time_hour
## 1      0   1012.0    10 2013-01-01T06:00:00Z
## 2      0   1012.3    10 2013-01-01T07:00:00Z
## 3      0   1012.5    10 2013-01-01T08:00:00Z
## 4      0   1012.2    10 2013-01-01T09:00:00Z
## 5      0   1011.9    10 2013-01-01T10:00:00Z
## 6      0   1012.4    10 2013-01-01T11:00:00Z
```

```
#Question 4: What is the number of observations and the number of variables?

# Get the dimensions of the dataset
dimensions <- dim(data)

# Number of observations (rows)
num_observations <- dimensions[1]

# Number of variables (columns)
num_variables <- dimensions[2]

# Print the results
cat("Number of Observations:", num_observations, "\n")
```

```
## Number of Observations: 26115
```

```r
cat("Number of Variables:", num_variables, "\n")
```

```
## Number of Variables: 15
```

```r
#Ques 5: Use piping and the appropriate commands to change the variable "origin" to have the
#factor data type. Show that the data type was successfully changed using the class()
#function.

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
```

```r
# Change the variable 'origin' to a factor and show the data type
data <- data %>%
  mutate(origin = as.factor(origin))

# Verify the data type of 'origin'
class(data$origin)
```

```
## [1] "factor"
```

```r
#question 6: Use piping and summarise() (or reframe) to display the mean and median for each of
#the levels in the origin variable.


# Summarize mean and median for each level of 'origin'
summary_stats <- data %>%
  group_by(origin) %>%
  summarise(
    mean_value = mean(temp, na.rm = TRUE),
    median_value = median(temp, na.rm = TRUE)
     )

# Print the summary statistics
print(summary_stats)
```

```
## # A tibble: 3 x 3
##   origin mean_value median_value
##   <fct>       <dbl>        <dbl>
## 1 EWR          55.5         55.9
## 2 JFK          54.5         54.0
## 3 LGA          55.8         55.9
```

```r
#Question 7: Read in the airports data from the nycflights13 library and merge the latitude and
#longitude variables with your dataset according to the origin airports as well as the
#destination airports. Name these variables "o_lat", "o_lon", "d_lat" and "d_lon".

library(nycflights13)
flights_data <- nycflights13::flights
airports_data <- nycflights13::airports

# Merge the latitude and longitude for the origin airports
flights_data <- flights_data %>%
  left_join(airports_data, by = c("origin" = "faa")) %>%
  rename(o_lat = lat, o_lon = lon)

# Merge the latitude and longitude for the destination airports
flights_data <- flights_data %>%
  left_join(airports_data, by = c("dest" = "faa")) %>%
  rename(d_lat = lat, d_lon = lon)
# Select relevant columns to display
flights_data <- flights_data %>% select(year, month, day, dep_time, arr_time, origin, dest, o_lat, o_lon
# Display the first few rows of the updated dataset
print(head(flights_data))
```

```
## # A tibble: 6 x 33
##    year month   day dep_time arr_time origin dest  o_lat o_lon d_lat d_lon
##   <int> <int> <int>    <int>    <int> <chr>  <chr> <dbl> <dbl> <dbl> <dbl>
## 1  2013     1     1      517      830 EWR    IAH    40.7 -74.2  30.0 -95.3
## 2  2013     1     1      533      850 LGA    IAH    40.8 -73.9  30.0 -95.3
## 3  2013     1     1      542      923 JFK    MIA    40.6 -73.8  25.8 -80.3
## 4  2013     1     1      544     1004 JFK    BQN    40.6 -73.8  NA    NA
## 5  2013     1     1      554      812 LGA    ATL    40.8 -73.9  33.6 -84.4
## 6  2013     1     1      554      740 EWR    ORD    40.7 -74.2  42.0 -87.9
## # i 22 more variables: sched_dep_time <int>, dep_delay <dbl>,
## #   sched_arr_time <int>, arr_delay <dbl>, carrier <chr>, flight <int>,
## #   tailnum <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>,
## #   time_hour <dttm>, name.x <chr>, alt.x <dbl>, tz.x <dbl>, dst.x <chr>,
## #   tzone.x <chr>, name.y <chr>, alt.y <dbl>, tz.y <dbl>, dst.y <chr>,
## #   tzone.y <chr>
```