PHW251 Problem Set 7

AUTHOR
Val Stacey

PUBLISHED
November 17, 2025

Part 1

For part 1 of this problem set we will work with motor vehicle crash data from New York City. You can read more about this publicly available data set on their website.

The data file is called “Motor_Vehicle_Collisions_Crashes.csv”. We want you to perform the following:

1. Rename the column names to lower-case and replace spaces with an underscore.
2. Select only:
    - crash_date
    - number_of_persons_injured
    - contributing_factor_vehicle_1
    - vehicle_type_code_1
3. Drop all rows that contain an NA value.
4. Make the values in the vehicle_type_code_1 variable all lowercase and replace the spaces with a dash.
5. Filter the data for vehicles that have a count of at least 500 (appear in the data set 500 times or more)
    - Hints: group_by(), mutate(), n(), filter()
6. Calculate the percentage of accidents by vehicle type
7. Which vehicle group accounted for 1.55% (0.0155) of the accidents?

We have grouped the questions below to push you to perform commands with less code. As you’re building your code we recommend going line by line to test, then combining to perform multiple steps in one command.

Questions 1-3

```
df <- df_motor %>%
clean_names() %>%
select(
    crash_date, number_of_persons_injured,
    contributing_factor_vehicle_1, vehicle_type_code_1
) %>%
drop_na()
```

Questions 4-5

```
df_a <- df %>%
mutate(
    vehicle_type_code_1 =
      str_to_lower(vehicle_type_code_1) %>%
      str_replace_all("[ /]", "-")
) %>%
group_by(vehicle_type_code_1) %>%
  mutate(n_crashes_v = n()) %>%
ungroup() %>%
filter(n_crashes_v >= 500)
```

Question 6

```
df_v_sum <- df_a %>%
  group_by(vehicle_type_code_1) %>%
  summarise(
    n_crashes_v = first(n_crashes_v),
    pct_crashes = n_crashes_v / nrow(df_a),
    .groups = "drop"
  )
```

Question 7

BUS

Part 2

For this part we will work with four tables that are relational to each other. The following keys link the tables together:

- patient_id: patients, schedule
- visit_id: schedule, visits
- doctor_id: visits, doctors

Question 8

You've been asked to collect information on patients who are actually on the schedule. To start this task, you need to join the patient data to the schedule data, since we only want to keep the observations that are present in both the patient data AND the schedule data.

Which kind of join do you use?

inner join

How many observations do you see in your joined data set? Notice that some patients have multiple visits.

```
patients_scheduled <- schedule %>%
  inner_join(patients, by = "patient_id")

n_distinct(patients_scheduled$patient_id)
```

```
[1] 44
```

There are 124 *observations* in the joined data set, but **only 44 unique patients**, (so 7 of them were not on the schedule)

Question 9

In the visits data, we have a variable called “follow_up” where Y means a follow-up is needed and N means a follow-up is not needed. How many patients require a follow-up? You will want to first make a join and then subset. Start with the data frame created in the previous question.

```
patients_visits <- patients_scheduled %>%
  left_join(visits, by = "visit_id", relationship = "one-to-many") %>%
  drop_na() %>%
  filter(follow_up == "Y")
```

Which join did you use?

left join (one-to-many)

How many patients need a follow-up?

26

Question 10

Which doctors do these patients need follow-up with? Print out each doctor's name.

```
dr_fups <- patients_visits %>%
  inner_join(doctors, by = "doctor_id") %>%
  select(doctor) %>%
  distinct() %>%
  arrange(doctor)

print(dr_fups)
```

```
# A tibble: 17 × 1
   doctor
   <chr>
 1 Ammar Phelps
 2 Amritpal Goodman
 3 Ariadne Anthony
 4 Bea Frame
 5 Cade Gale
 6 Daanyaal Griffin
 7 Ellesha Castaneda
 8 Estelle Landry
 9 Huzaifa Chung
10 Jamie-Lee Wilder
11 Jeremy Camacho
12 Merlin Jacobs
13 Millie Albert
14 Rabia Browning
15 Tudor Moran
16 Vera Irwin
17 Wiktoria Travis
```

Which join did you use?

inner_join