Exercises: Hotel bookings - data wrangling

```
library(tidyverse)
library(skimr) # install.packages("skimr")
library(dplyr)
```

```
# load dataset
hotels <- read_csv("hotels.csv")
```

Exercise 1. Warm up! Take a look at an overview of the data with the `skim()` function.

Note: I already gave you the answer to this exercise. You just need to knit the document and view the output. A definition of all variables is given in the Data dictionary section at the end, though you don't need to familiarize yourself with all variables in order to work through these exercises.

```
skim(hotels)
```

Table 1: Data summary

Name	hotels
Number of rows	119390
Number of columns	32
Column type frequency:	
character	13
Date	1
numeric	18
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
hotel	0	1	10	12	0	2	0
arrival_date_month	0	1	3	9	0	12	0
meal	0	1	2	9	0	5	0
country	0	1	2	4	0	178	0
market_segment	0	1	6	13	0	8	0
distribution_channel	0	1	3	9	0	5	0
reserved_room_type	0	1	1	1	0	10	0
assigned_room_type	0	1	1	1	0	12	0
deposit_type	0	1	10	10	0	3	0
agent	0	1	1	4	0	334	0
company	0	1	1	4	0	353	0
customer_type	0	1	5	15	0	4	0
reservation_status	0	1	7	9	0	3	0

Variable type: Date

skim_variable	n_missing	complete_rate	min	max	median	n_unique
reservation_status_date	0	1	2014-10-17	2017-09-14	2016-08-07	926

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
is_canceled	0	1	0.37	0.48	0.00	0.00	0.00	1	1	
lead_time	0	1	104.01	106.86	0.00	18.00	69.00	160	737	
arrival_date_year	0	1	2016.16	0.71	2015.00	2016.00	2016.00	2017	2017	
arrival_date_week_number	0	1	27.17	13.61	1.00	16.00	28.00	38	53	
arrival_date_day_of_month	0	1	15.80	8.78	1.00	8.00	16.00	23	31	
stays_in_weekend_nights	0	1	0.93	1.00	0.00	0.00	1.00	2	19	
stays_in_week_nights	0	1	2.50	1.91	0.00	1.00	2.00	3	50	
adults	0	1	1.86	0.58	0.00	2.00	2.00	2	55	
children	4	1	0.10	0.40	0.00	0.00	0.00	0	10	
babies	0	1	0.01	0.10	0.00	0.00	0.00	0	10	
is_repeated_guest	0	1	0.03	0.18	0.00	0.00	0.00	0	1	
previous_cancellations	0	1	0.09	0.84	0.00	0.00	0.00	0	26	
previous_bookings_not_canceled	0	1	0.14	1.50	0.00	0.00	0.00	0	72	
booking_changes	0	1	0.22	0.65	0.00	0.00	0.00	0	21	
days_in_waiting_list	0	1	2.32	17.59	0.00	0.00	0.00	0	391	
adr	0	1	101.83	50.54	-6.38	69.29	94.58	126	5400	
required_car_parking_spaces	0	1	0.06	0.25	0.00	0.00	0.00	0	8	
total_of_special_requests	0	1	0.57	0.79	0.00	0.00	0.00	1	5	

Exercise 2. Are people traveling on a whim? Let's see...

Fill in the blanks for filtering for hotel bookings where the guest is **not** from the US (`country` code `"USA"`) and the `lead_time` is less than 1 day.

Note: You will need to set `eval=TRUE` when you have an answer you want to try out.

```
hotels %>%
  filter(
    country != "USA",
    lead_time < 1
    )
```

```
## # A tibble: 6,174 x 32
##    hotel  is_ca~1 lead_~2 arriv~3 arriv~4 arriv~5 arriv~6 stays~7 stays~8 adults
##    <chr>    <dbl>   <dbl>   <dbl> <chr>     <dbl>   <dbl>   <dbl>   <dbl>  <dbl>
##  1 Resor~       0       0    2015 July         27       1       0       2      2
##  2 Resor~       0       0    2015 July         27       1       0       1      2
##  3 Resor~       0       0    2015 July         27       2       0       1      2
##  4 Resor~       0       0    2015 July         27       2       0       1      2
##  5 Resor~       0       0    2015 July         27       2       0       1      2
##  6 Resor~       0       0    2015 July         28       5       1       0      2
##  7 Resor~       0       0    2015 July         28       6       0       0      1
##  8 Resor~       0       0    2015 July         28       7       0       1      1
##  9 Resor~       0       0    2015 July         28       7       0       1      3
## 10 Resor~       0       0    2015 July         28       7       0       1      1
## # ... with 6,164 more rows, 22 more variables: children <dbl>, babies <dbl>,
## #   meal <chr>, country <chr>, market_segment <chr>,
## #   distribution_channel <chr>, is_repeated_guest <dbl>,
## #   previous_cancellations <dbl>, previous_bookings_not_canceled <dbl>,
## #   reserved_room_type <chr>, assigned_room_type <chr>, booking_changes <dbl>,
## #   deposit_type <chr>, agent <chr>, company <chr>, days_in_waiting_list <dbl>,
## #   customer_type <chr>, adr <dbl>, required_car_parking_spaces <dbl>, ...
```

Exercise 3. How many bookings involve at least 1 child **or** baby?

In the following chunk, replace

- `[AT LEAST]` with the logical operator for "at least" (in two places)
- `[OR]` with the logical operator for "or"

Note: You will need to set `eval=TRUE` when you have an answer you want to try out.

```
hotels %>%
  filter(
    children >= 1 | babies >= 1
    )
```

```
## # A tibble: 9,332 x 32
##    hotel  is_ca~1 lead_~2 arriv~3 arriv~4 arriv~5 arriv~6 stays~7 stays~8 adults
##    <chr>    <dbl>   <dbl>   <dbl> <chr>     <dbl>   <dbl>   <dbl>   <dbl>  <dbl>
##  1 Resor~       0      18    2015 July         27       1       0       4      2
##  2 Resor~       1      47    2015 July         27       2       2       5      2
##  3 Resor~       0       1    2015 July         27       2       0       1      2
##  4 Resor~       0      10    2015 July         27       3       0       2      2
##  5 Resor~       1      79    2015 July         27       3       6      15      2
##  6 Resor~       0     101    2015 July         27       3       2       5      2
##  7 Resor~       0      92    2015 July         27       4       2       4      1
##  8 Resor~       1      26    2015 July         27       4       2       5      2
##  9 Resor~       0     102    2015 July         27       4       2       5      2
## 10 Resor~       0      78    2015 July         27       4       2       5      2
## # ... with 9,322 more rows, 22 more variables: children <dbl>, babies <dbl>,
## #   meal <chr>, country <chr>, market_segment <chr>,
## #   distribution_channel <chr>, is_repeated_guest <dbl>,
## #   previous_cancellations <dbl>, previous_bookings_not_canceled <dbl>,
## #   reserved_room_type <chr>, assigned_room_type <chr>, booking_changes <dbl>,
## #   deposit_type <chr>, agent <chr>, company <chr>, days_in_waiting_list <dbl>,
## #   customer_type <chr>, adr <dbl>, required_car_parking_spaces <dbl>, ...
```

Exercise 4. Do you think it's more likely to find bookings with children or babies in city hotels or resort hotels? Test your intuition. Using `filter()` determine the number of bookings in resort hotels that have more than 1 child **or** baby in the room? Then, do the same for city hotels, and compare the numbers of rows in the resulting filtered data frames.

```
hotels %>%
  filter(hotel == "Resort Hotel",
         children > 1 | babies > 1
         )
```

```
## # A tibble: 1,655 x 32
##    hotel  is_ca~1 lead_~2 arriv~3 arriv~4 arriv~5 arriv~6 stays~7 stays~8 adults
##    <chr>    <dbl>   <dbl>   <dbl> <chr>     <dbl>   <dbl>   <dbl>   <dbl>  <dbl>
##  1 Resor~       1      47    2015 July         27       2       2       5      2
##  2 Resor~       0       1    2015 July         27       2       0       1      2
##  3 Resor~       0      10    2015 July         27       3       0       2      2
##  4 Resor~       0      92    2015 July         27       4       2       4      1
##  5 Resor~       1      26    2015 July         27       4       2       5      2
##  6 Resor~       0       2    2015 July         27       4       0       1      2
##  7 Resor~       1      34    2015 July         28       5       2       4      2
##  8 Resor~       0      97    2015 July         28       5       2       5      2
##  9 Resor~       0       8    2015 July         28       5       4       5      2
```

```
## 10 Resor~       0     115    2015 July         28       6       1       4      2
## # ... with 1,645 more rows, 22 more variables: children <dbl>, babies <dbl>,
## #   meal <chr>, country <chr>, market_segment <chr>,
## #   distribution_channel <chr>, is_repeated_guest <dbl>,
## #   previous_cancellations <dbl>, previous_bookings_not_canceled <dbl>,
## #   reserved_room_type <chr>, assigned_room_type <chr>, booking_changes <dbl>,
## #   deposit_type <chr>, agent <chr>, company <chr>, days_in_waiting_list <dbl>,
## #   customer_type <chr>, adr <dbl>, required_car_parking_spaces <dbl>, ...
```

```
# pay attention to correctness and code style
```

```
hotels %>%
  filter(hotel == "City Hotel",
         children > 1 | babies > 1
         )
```

```
## # A tibble: 2,091 x 32
##    hotel  is_ca~1 lead_~2 arriv~3 arriv~4 arriv~5 arriv~6 stays~7 stays~8 adults
##    <chr>    <dbl>   <dbl>   <dbl> <chr>     <dbl>   <dbl>   <dbl>   <dbl>  <dbl>
##  1 City ~       0      67    2015 July         28      10       2       3      2
##  2 City ~       0       6    2015 August       32       3       1       0      1
##  3 City ~       0       0    2015 August       32       7       0       2      2
##  4 City ~       0       0    2015 August       32       8       0       1      1
##  5 City ~       0       0    2015 August       32       8       2       1      2
##  6 City ~       0       1    2015 August       33       9       2       3      2
##  7 City ~       0       1    2015 August       33       9       1       0      2
##  8 City ~       0       1    2015 August       33       9       2       0      2
##  9 City ~       0       0    2015 August       33       9       2       0      2
## 10 City ~       0       1    2015 August       33      10       1       1      0
## # ... with 2,081 more rows, 22 more variables: children <dbl>, babies <dbl>,
## #   meal <chr>, country <chr>, market_segment <chr>,
## #   distribution_channel <chr>, is_repeated_guest <dbl>,
## #   previous_cancellations <dbl>, previous_bookings_not_canceled <dbl>,
## #   reserved_room_type <chr>, assigned_room_type <chr>, booking_changes <dbl>,
## #   deposit_type <chr>, agent <chr>, company <chr>, days_in_waiting_list <dbl>,
## #   customer_type <chr>, adr <dbl>, required_car_parking_spaces <dbl>, ...
```

```
# pay attention to correctness and code style
# It is more likely to find bookings with children or babies in
# city hotels.
```

Exercise 5. Create a frequency table of the number of `adults` in a booking. Display the results in descending order so the most common observation is on top. What is the most common number of adults in bookings in this dataset? Are there any surprising results?

Note: Don't forget to label your R chunk as well (where it says `label-me-1`). Your label should be short, informative, and shouldn't include spaces. It also shouldn't repeat a previous label, otherwise R Markdown will give you an error about repeated R chunk labels.

```
hotels %>%
    count(adults) %>%
    arrange(desc(n))
```

```
## # A tibble: 14 x 2
##    adults     n
##     <dbl> <int>
```

```
##  1      2 89680
##  2      1 23027
##  3      3  6202
##  4      0   403
##  5      4    62
##  6     26     5
##  7      5     2
##  8     20     2
##  9     27     2
## 10      6     1
## 11     10     1
## 12     40     1
## 13     50     1
## 14     55     1
```

```
# pay attention to correctness and code style
# The most common number of adults is 2.
```

Exercise 6. Calculate minimum, mean, median, and maximum average daily rate (`adr`) grouped by `hotel` type so that you can get these statistics separately for resort and city hotels. Which type of hotel is higher, on average?

```
hotels %>%
  group_by(hotel) %>%
    summarize(min(adr), mean(adr), median(adr), max(adr))
```

```
## # A tibble: 2 x 5
##   hotel        `min(adr)` `mean(adr)` `median(adr)` `max(adr)`
##   <chr>             <dbl>       <dbl>         <dbl>      <dbl>
## 1 City Hotel         0          105.           99.9       5400
## 2 Resort Hotel      -6.38        95.0          75          508
```

```
# pay attention to correctness and code style
# City hotels are higher on average
```

Exercise 7. We observe two unusual values in the summary statistics above – a negative minimum, and a very high maximum). What types of hotels are these? Locate these observations in the dataset and find out the arrival date (year and month) as well as how many people (adults, children, and babies) stayed in the room. You can investigate the data in the viewer to locate these values, but preferably you should identify them in a reproducible way with some code.

Hint: For example, you can `filter` for the given `adr` amounts and `select` the relevant columns.

```
hotels %>%
  filter(adr == -6.38 |
         adr == 5400
         ) %>%
    select(hotel, arrival_date_month, arrival_date_year, adults, children, babies)
```

```
## # A tibble: 2 x 6
##   hotel        arrival_date_month arrival_date_year adults children babies
##   <chr>        <chr>                          <dbl>  <dbl>    <dbl>  <dbl>
## 1 Resort Hotel March                           2017      2        0      0
## 2 City Hotel   March                           2016      2        0      0
```

```
# pay attention to correctness and code style
```

Data dictionary Below is the full data dictionary. Note that it is long (there are lots of variables in the data), but we will be using a limited set of the variables for our analysis.

variable	class	description
hotel	character	Hotel (H1 = Resort Hotel or H2 = City Hotel)
is_canceled	double	Value indicating if the booking was canceled (1) or not (0)
lead_time	double	Number of days that elapsed between the entering date of the booking into the PMS and the arrival date
arrival_date_year	double	Year of arrival date
arrival_date_month	character	Month of arrival date
arrival_date_week_number	double	Week number of year for arrival date
arrival_date_day_of_month	double	Day of arrival date
stays_in_weekend_nights	double	Number of weekend nights (Saturday or Sunday) the guest stayed or booked to stay at the hotel
stays_in_week_nights	double	Number of week nights (Monday to Friday) the guest stayed or booked to stay at the hotel
adults	double	Number of adults
children	double	Number of children
babies	double	Number of babies
meal	character	Type of meal booked. Categories are presented in standard hospitality meal packages: Undefined/SC – no meal package;BB – Bed & Breakfast; HB – Half board (breakfast and one other meal – usually dinner); FB – Full board (breakfast, lunch and dinner)
country	character	Country of origin. Categories are represented in the ISO 3155–3:2013 format
market_segment	character	Market segment designation. In categories, the term “TA” means “Travel Agents” and “TO” means “Tour Operators”
distribution_channel	character	Booking distribution channel. The term “TA” means “Travel Agents” and “TO” means “Tour Operators”
is_repeated_guest	double	Value indicating if the booking name was from a repeated guest (1) or not (0)
previous_cancellations	double	Number of previous bookings that were cancelled by the customer prior to the current booking
previous_bookings_not_canceled	double	Number of previous bookings not cancelled by the customer prior to the current booking
reserved_room_type	character	Code of room type reserved. Code is presented instead of designation for anonymity reasons
assigned_room_type	character	Code for the type of room assigned to the booking. Sometimes the assigned room type differs from the reserved room type due to hotel operation reasons (e.g. overbooking) or by customer request. Code is presented instead of designation for anonymity reasons
booking_changes	double	Number of changes/amendments made to the booking from the moment the booking was entered on the PMS until the moment of check-in or cancellation
deposit_type	character	Indication on if the customer made a deposit to guarantee the booking. This variable can assume three categories:No Deposit – no deposit was made;Non Refund – a deposit was made in the value of the total stay cost;Refundable – a deposit was made with a value under the total cost of stay.
agent	character	ID of the travel agency that made the booking
company	character	ID of the company/entity that made the booking or responsible for paying the booking. ID is presented instead of designation for anonymity reasons
days_in_waiting_list	double	Number of days the booking was in the waiting list before it was confirmed to the customer

variable	class	description
customer_type	character	Type of booking, assuming one of four categories:Contract - when the booking has an allotment or other type of contract associated to it;Group – when the booking is associated to a group;Transient – when the booking is not part of a group or contract, and is not associated to other transient booking;Transient-party – when the booking is transient, but is associated to at least other transient booking
adr	double	Average Daily Rate as defined by dividing the sum of all lodging transactions by the total number of staying nights
required_car_parking_spaces	double	Number of car parking spaces required by the customer
total_of_special_requests	double	Number of special requests made by the customer (e.g. twin bed or high floor)
reservation_status	character	Reservation last status, assuming one of three categories:Canceled – booking was canceled by the customer;Check-Out – customer has checked in but already departed;No-Show – customer did not check-in and did inform the hotel of the reason why
reservation_status_date	double	Date at which the last status was set. This variable can be used in conjunction with the ReservationStatus to understand when was the booking canceled or when did the customer checked-out of the hotel