download_file.Rmd

Bo

2022-10-03

Download InsideAirBnB data

This R script is used to download datasets from InsideAirBnB.com, which will later be used, after further cleaning and preparing, for analysis of AirBnB pricing in relation to weekends and weekdays.

Step 1) Installing and running the required packages:

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.8     v dplyr   1.0.9
## v tidyr   1.2.0     v stringr 1.4.1
## v readr   2.1.2     v forcats 0.5.2
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

Step 2) Creating lists of the urls that need to be downloaded: (2 lists need to be created, one for the listings and one for the calender data)

```
urls_calender = c("http://data.insideairbnb.com/united-states/co/denver/2022-09-26/data/calendar.csv.gz
urls_listing = c("http://data.insideairbnb.com/united-states/co/denver/2022-09-26/data/listings.csv.gz"
```

Step 3) Create a for loop that iterates over the urls_calender list and, 1) downloads the files located in the urls and 2) renames these files such that the files ar easiliy recognizable:

```
for (url in urls_calender) {
  filename = paste(gsub('[^a-zA-Z]', '', url), '.csv')
  filename = gsub('httpdatainsideairbnbcom', '', filename)
  download.file(url, destfile = filename) # download file
}
```

Step 4) Create a for loop that iterates over the urls_listing list and, 1) downloads the files located in the urls and 2) renames these files such that the files are easiliy recognizable:

```
for (url in urls_listing) {
  filename = paste(gsub('[^a-zA-Z]', '', url), '.csv')
  filename = gsub('httpdatainsideairbnbcom', '', filename)
```

```
  download.file(url, destfile = filename) # download file
}
```

Step 5) Create a complete list for urls_calender data, which includes all of the downloaded files. This complete list is called calender_data:

```
calender_data <- lapply(urls_calender, function(url) {
  ds = read_csv(url, n_max = 5000)
  city_name = strsplit(url, '/')[[1]][6]
  ds = ds %>% mutate(city = city_name)
  ds
})
```

Step 6) Create a complete list for urls_listing data, which includes all of the downloaded files, but only reads the "id" and "room_type" columns, which are the columns that are required for the analysis. This complete list is called listing_data.

```
listing_data <- lapply(urls_listing, function(url) {
  ds = read_csv(url, col_select = c("id","room_type"), n_max = 5000)
  ds
})
```

Step 7) Bind all of the rows for both complete lists, which is possible because all of the lists contain the same columns:

```
calender_data <- calender_data %>% bind_rows()
listing_data <- listing_data %>% bind_rows()
```

The final datasets should now be downloaded and read into R, ready for further preparation!