HW 3

February 26, 2021

1 IST 387 HW 3

Copyright 2021, Jeffrey Stanton, Jeffrey Saltz, and Jasmina Tacheva

[1]:
```
# Enter your name here: Connor Hanan
```

1.0.1 Attribution statement: (choose only one and delete the rest)

[1]:
```
# 1. I did this homework by myself, with help from the book and the professor.
```

1.0.2 Reminders of things to practice from last week:

Make a data frame data.frame( ) Row index of max/min which.max( ) which.min( ) Sort value or order rows sort( ) order( ) Descriptive statistics mean( ) sum( ) max( ) Conditional statement if (condition) “true stuff” else “false stuff”

1.0.3 This Week:

Often, when you get a dataset, it is not in the format you want. You can (and should) use code to refine the dataset to become more useful. As Chapter 6 of Introduction to Data Science mentions, this is called “data munging.” In this homework, you will read in a dataset from the web and work on it (in a data frame) to improve its usefulness.

1.1 Part 1: Use read_csv( ) to read a CSV file from the web into a data frame:

A. Use R code to read directly from a URL on the web. Store the dataset into a new dataframe, called dfComps. The URL is: “https://ist387.s3.us-east-2.amazonaws.com/data/Companies.csv” **Hint:** use read_csv( ), not read.csv( ). This is from the **tidyverse package**. Check the help to compare them.

[2]:
```
library(tidyverse) #initialize library
dfComps <- read_csv("https://ist387.s3.us-east-2.amazonaws.com/data/Companies.
↪csv") #read csv into variable
```

```
  Attaching packages                               tidyverse
1.3.0

  ggplot2 3.3.2       purrr   0.3.4
  tibble  3.0.4       dplyr   1.0.2
```

```
 tidyr    1.1.2      stringr 1.4.0
 readr    1.4.0      forcats 0.5.0

  Conflicts
tidyverse_conflicts()
 dplyr::filter() masks stats::filter()
 dplyr::lag()    masks stats::lag()


  Column specification

cols(
  permalink = col_character(),
  name = col_character(),
  homepage_url = col_character(),
  category_list = col_character(),
  market = col_character(),
  funding_total_usd = col_character(),
  status = col_character(),
  country_code = col_character(),
  state_code = col_character(),
  region = col_character(),
  city = col_character(),
  funding_rounds = col_double(),
  founded_at = col_character(),
  founded_month = col_character(),
  founded_quarter = col_character(),
  founded_year = col_double(),
  first_funding_at = col_character(),
  last_funding_at = col_character()
)
```

1.2 Part 2: Create a new data frame that only contains companies with a homepage URL:

B. Use head( ) and tail( ) to examine the **dfComps** dataframe. **Add a block comment that briefly describes what you see.**

[3]:

```
head(dfComps) #check the first 6 rows
tail(dfComps) #check the last 6 rows

#I see various information about companies and history, including URLs, region,␣
 ↪industry, founding year, and so on
```

A tibble: 6 × 18

permalink <chr>	name <chr>	homepage_url <chr>
/organization/waywire	#waywire	http://www.waywire.com
/organization/tv-communications	&TV Communications	http://enjoyandtv.com
/organization/rock-your-paper	'Rock' Your Paper	http://www.rockyourpaper.org
/organization/in-touch-network	(In)Touch Network	http://www.InTouchNetwork.com
/organization/n-plusn	+n (PlusN)	http://plusn.com
/organization/r-ranch-and-mine	-R- Ranch and Mine	NA

A tibble: 6 × 18

permalink <chr>	name <chr>	homepa <chr>
/organization/zytoprotec	Zytoprotec	http://
/organization/zzish	Zzish	http://
/organization/zznode-science-and-technology-co-ltd	ZZNode Science and Technology	http://
/organization/zzzzapp-com	Zzzzapp Wireless ltd.	http://
/organization/a-list-games	[a]list games	http://
/organization/x	[x+1]	http://

C. Create a variable (called **noURL**) that has a value of **TRUE** if a company is missing a homepage URL.

[4]:
```r
str(dfComps) #check structure to see how many rows are total so I can see if data makes sense

noURL <- is.na(dfComps$homepage_url) #save TRUE/FALSE if they are missing a URL
noURL #look at stored data
```

```
tibble [47,758 × 18] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
 $ permalink        : chr [1:47758] "/organization/waywire" "/organization/tv-
communications" "/organization/rock-your-paper" "/organization/in-touch-network"
…
 $ name             : chr [1:47758] "#waywire" "&TV Communications" "'Rock' Your
Paper" "(In)Touch Network" …
 $ homepage_url     : chr [1:47758] "http://www.waywire.com"
"http://enjoyandtv.com" "http://www.rockyourpaper.org"
"http://www.InTouchNetwork.com" …
 $ category_list    : chr [1:47758] "|Entertainment|Politics|Social Media|News|"
"|Games|" "|Publishing|Education|" "|Electronics|Guides|Coffee|Restaurants|Music
|iPhone|Apps|Mobile|iOS|E-Commerce|" …
 $ market           : chr [1:47758] "News" "Games" "Publishing" "Electronics"
…
 $ funding_total_usd: chr [1:47758] "1 750 000" "4 000 000" "40 000" "1 500 000"
…
 $ status           : chr [1:47758] "acquired" "operating" "operating"
"operating" …
 $ country_code     : chr [1:47758] "USA" "USA" "EST" "GBR" …
 $ state_code       : chr [1:47758] "NY" "CA" NA NA …
 $ region           : chr [1:47758] "New York City" "Los Angeles" "Tallinn"
"London" …
```

```
 $ city             : chr [1:47758] "New York" "Los Angeles" "Tallinn" "London"
…
 $ funding_rounds   : num [1:47758] 1 2 1 1 2 1 1 1 1 1 …
 $ founded_at       : chr [1:47758] "01/06/2012" NA "26/10/2012" "01/04/2011"
…
 $ founded_month    : chr [1:47758] "2012-06" NA "2012-10" "2011-04" …
 $ founded_quarter  : chr [1:47758] "2012-Q2" NA "2012-Q4" "2011-Q2" …
 $ founded_year     : num [1:47758] 2012 NA 2012 2011 2012 …
 $ first_funding_at : chr [1:47758] "30/06/2012" "04/06/2010" "09/08/2012"
"01/04/2011" …
 $ last_funding_at  : chr [1:47758] "30/06/2012" "23/09/2010" "09/08/2012"
"01/04/2011" …
 - attr(*, "spec")=
  .. cols(
  ..   permalink = col_character(),
  ..   name = col_character(),
  ..   homepage_url = col_character(),
  ..   category_list = col_character(),
  ..   market = col_character(),
  ..   funding_total_usd = col_character(),
  ..   status = col_character(),
  ..   country_code = col_character(),
  ..   state_code = col_character(),
  ..   region = col_character(),
  ..   city = col_character(),
  ..   funding_rounds = col_double(),
  ..   founded_at = col_character(),
  ..   founded_month = col_character(),
  ..   founded_quarter = col_character(),
  ..   founded_year = col_double(),
  ..   first_funding_at = col_character(),
  ..   last_funding_at = col_character()
  .. )
```

1. FALSE 2. FALSE 3. FALSE 4. FALSE 5. FALSE 6. TRUE 7. FALSE 8. FALSE 9. FALSE
10. FALSE 11. FALSE 12. FALSE 13. FALSE 14. TRUE 15. FALSE 16. FALSE 17. TRUE
18. FALSE 19. FALSE 20. FALSE 21. FALSE 22. FALSE 23. FALSE 24. FALSE 25. FALSE
26. FALSE 27. FALSE 28. FALSE 29. FALSE 30. FALSE 31. FALSE 32. FALSE 33. FALSE
34. FALSE 35. FALSE 36. FALSE 37. FALSE 38. FALSE 39. FALSE 40. FALSE 41. TRUE
42. FALSE 43. FALSE 44. FALSE 45. FALSE 46. FALSE 47. FALSE 48. FALSE 49. FALSE
50. FALSE 51. FALSE 52. FALSE 53. FALSE 54. FALSE 55. FALSE 56. FALSE 57. FALSE
58. FALSE 59. TRUE 60. FALSE 61. FALSE 62. FALSE 63. FALSE 64. FALSE 65. FALSE
66. FALSE 67. FALSE 68. FALSE 69. FALSE 70. FALSE 71. FALSE 72. FALSE 73. FALSE
74. FALSE 75. FALSE 76. FALSE 77. FALSE 78. FALSE 79. FALSE 80. FALSE 81. FALSE
82. FALSE 83. FALSE 84. TRUE 85. FALSE 86. FALSE 87. TRUE 88. FALSE 89. FALSE
90. FALSE 91. FALSE 92. FALSE 93. FALSE 94. FALSE 95. FALSE 96. TRUE 97. FALSE
98. FALSE 99. FALSE 100. FALSE 101. TRUE 102. FALSE 103. FALSE 104. FALSE 105. FALSE
106. FALSE 107. FALSE 108. FALSE 109. FALSE 110. FALSE 111. FALSE 112. TRUE 113. FALSE

```
114. FALSE 115. TRUE 116. FALSE 117. FALSE 118. FALSE 119. FALSE 120. FALSE 121. FALSE
122. FALSE 123. FALSE 124. FALSE 125. FALSE 126. FALSE 127. FALSE 128. FALSE 129. FALSE
130. FALSE 131. FALSE 132. FALSE 133. FALSE 134. TRUE 135. FALSE 136. FALSE 137. FALSE
138. FALSE 139. FALSE 140. FALSE 141. FALSE 142. FALSE 143. FALSE 144. FALSE 145. FALSE
146. FALSE 147. FALSE 148. FALSE 149. FALSE 150. FALSE 151. FALSE 152. FALSE 153. FALSE
154. FALSE 155. FALSE 156. TRUE 157. FALSE 158. FALSE 159. FALSE 160. FALSE 161. FALSE
162. FALSE 163. FALSE 164. TRUE 165. FALSE 166. FALSE 167. FALSE 168. FALSE 169. FALSE
170. FALSE 171. FALSE 172. FALSE 173. FALSE 174. FALSE 175. FALSE 176. FALSE 177. FALSE
178. FALSE 179. FALSE 180. FALSE 181. FALSE 182. FALSE 183. FALSE 184. FALSE 185. FALSE
186. FALSE 187. FALSE 188. FALSE 189. FALSE 190. FALSE 191. FALSE 192. FALSE 193. FALSE
194. FALSE 195. FALSE 196. FALSE 197. FALSE 198. TRUE 199. FALSE 200. FALSE 201.
202. FALSE 203. FALSE 204. FALSE 205. FALSE 206. FALSE 207. FALSE 208. FALSE 209. FALSE
210. FALSE 211. FALSE 212. FALSE 213. FALSE 214. FALSE 215. FALSE 216. FALSE 217. FALSE
218. FALSE 219. FALSE 220. FALSE 221. FALSE 222. FALSE 223. FALSE 224. FALSE 225. FALSE
226. FALSE 227. FALSE 228. FALSE 229. FALSE 230. FALSE 231. FALSE 232. FALSE 233. FALSE
234. FALSE 235. FALSE 236. FALSE 237. FALSE 238. FALSE 239. FALSE 240. FALSE 241. FALSE
242. FALSE 243. FALSE 244. FALSE 245. FALSE 246. FALSE 247. FALSE 248. FALSE 249. FALSE
250. FALSE 251. FALSE 252. FALSE 253. FALSE 254. FALSE 255. FALSE 256. FALSE 257. FALSE
258. FALSE 259. FALSE 260. FALSE 261. FALSE 262. FALSE 263. FALSE 264. FALSE 265. FALSE
266. FALSE 267. FALSE 268. FALSE 269. FALSE 270. FALSE 271. FALSE 272. FALSE 273. FALSE
274. FALSE 275. FALSE 276. FALSE 277. FALSE 278. FALSE 279. FALSE 280. FALSE 281. FALSE
282. FALSE 283. FALSE 284. FALSE 285. FALSE 286. FALSE 287. FALSE 288. FALSE 289. FALSE
290. FALSE 291. FALSE 292. FALSE 293. FALSE 294. FALSE 295. FALSE 296. FALSE 297. FALSE
298. FALSE 299. FALSE 300. FALSE 301. FALSE 302. FALSE 303. FALSE 304. FALSE 305. FALSE
306. FALSE 307. FALSE 308. FALSE 309. FALSE 310. FALSE 311. TRUE 312. FALSE 313. FALSE
314. FALSE 315. FALSE 316. FALSE 317. FALSE 318. FALSE 319. FALSE 320. FALSE 321. FALSE
322. FALSE 323. FALSE 324. FALSE 325. FALSE 326. TRUE 327. FALSE 328. FALSE 329. FALSE
330. FALSE 331. FALSE 332. FALSE 333. FALSE 334. FALSE 335. FALSE 336. FALSE 337. FALSE
338. FALSE 339. FALSE 340. FALSE 341. FALSE 342. FALSE 343. FALSE 344. FALSE 345. FALSE
346. FALSE 347. FALSE 348. FALSE 349. FALSE 350. FALSE 351. FALSE 352. FALSE 353. FALSE
354. FALSE 355. FALSE 356. FALSE 357. TRUE 358. FALSE 359. FALSE 360. FALSE 361. FALSE
362. FALSE 363. FALSE 364. FALSE 365. FALSE 366. FALSE 367. FALSE 368. TRUE 369. FALSE
370. FALSE 371. FALSE 372. FALSE 373. FALSE 374. FALSE 375. FALSE 376. FALSE 377. FALSE
378. FALSE 379. FALSE 380. FALSE 381. FALSE 382. FALSE 383. FALSE 384. FALSE 385. FALSE
386. FALSE 387. FALSE 388. FALSE 389. FALSE 390. FALSE 391. FALSE 392. FALSE 393. FALSE
394. FALSE 395. FALSE 396. FALSE 397. FALSE 398. FALSE 399. FALSE 400. FALSE 401. FALSE
```

D. Use the table( ) command to summarize the contents of **noURL**. **Write a comment interpreting what you see – how many companies are missing a homepage URL?**

[5]:
```
table(noURL)

#3323 companies are missing a homepage URL, while 44435 companies have one
```

```
noURL
FALSE  TRUE
44435  3323
```

E. Use **subsetting** to create a new dataframe that contains only the companies with homepage

URLs (store that dataframe in **urlComps**).

[6]:

```r
urlComps <- dfComps[is.na(dfComps$homepage_url)==FALSE,] #store all rows where homepage_url != NA
str(urlComps) #check structure to see that number of rows matches our table() check
```

```
tibble [44,435 × 18] (S3: tbl_df/tbl/data.frame)
 $ permalink        : chr [1:44435] "/organization/waywire" "/organization/tv-
communications" "/organization/rock-your-paper" "/organization/in-touch-network"
…
 $ name             : chr [1:44435] "#waywire" "&TV Communications" "'Rock' Your
Paper" "(In)Touch Network" …
 $ homepage_url     : chr [1:44435] "http://www.waywire.com"
"http://enjoyandtv.com" "http://www.rockyourpaper.org"
"http://www.InTouchNetwork.com" …
 $ category_list    : chr [1:44435] "|Entertainment|Politics|Social Media|News|"
"|Games|" "|Publishing|Education|" "|Electronics|Guides|Coffee|Restaurants|Music
|iPhone|Apps|Mobile|iOS|E-Commerce|" …
 $ market           : chr [1:44435] "News" "Games" "Publishing" "Electronics"
…
 $ funding_total_usd: chr [1:44435] "1 750 000" "4 000 000" "40 000" "1 500 000"
…
 $ status           : chr [1:44435] "acquired" "operating" "operating"
"operating" …
 $ country_code     : chr [1:44435] "USA" "USA" "EST" "GBR" …
 $ state_code       : chr [1:44435] "NY" "CA" NA NA …
 $ region           : chr [1:44435] "New York City" "Los Angeles" "Tallinn"
"London" …
 $ city             : chr [1:44435] "New York" "Los Angeles" "Tallinn" "London"
…
 $ funding_rounds   : num [1:44435] 1 2 1 1 2 1 1 1 1 1 …
 $ founded_at       : chr [1:44435] "01/06/2012" NA "26/10/2012" "01/04/2011"
…
 $ founded_month    : chr [1:44435] "2012-06" NA "2012-10" "2011-04" …
 $ founded_quarter  : chr [1:44435] "2012-Q2" NA "2012-Q4" "2011-Q2" …
 $ founded_year     : num [1:44435] 2012 NA 2012 2011 2012 …
 $ first_funding_at : chr [1:44435] "30/06/2012" "04/06/2010" "09/08/2012"
"01/04/2011" …
 $ last_funding_at  : chr [1:44435] "30/06/2012" "23/09/2010" "09/08/2012"
"01/04/2011" …
```

F. Use the dim() command on **urlComps** to confirm that the data frame contains **44,435** observations and **19** columns/variables.

[7]:

```r
dim(urlComps) #all my tables have 18 columns, assuming that is correct
```

1. 44435 2. 18

1.3 Part 3: Analyze the numeric variables in the dataframe.

G. How many **numeric variables** does the dataframe have? You can figure that out by looking at the output of **str(urlComps)**.

[8]:

```
str(urlComps) #check structure of the df

#it has two (2) numeric variables (funding_rounds and founded_year)
```

```
tibble [44,435 × 18] (S3: tbl_df/tbl/data.frame)
 $ permalink        : chr [1:44435] "/organization/waywire" "/organization/tv-
communications" "/organization/rock-your-paper" "/organization/in-touch-network"
…
 $ name             : chr [1:44435] "#waywire" "&TV Communications" "'Rock' Your
Paper" "(In)Touch Network" …
 $ homepage_url     : chr [1:44435] "http://www.waywire.com"
"http://enjoyandtv.com" "http://www.rockyourpaper.org"
"http://www.InTouchNetwork.com" …
 $ category_list    : chr [1:44435] "|Entertainment|Politics|Social Media|News|"
"|Games|" "|Publishing|Education|" "|Electronics|Guides|Coffee|Restaurants|Music
|iPhone|Apps|Mobile|iOS|E-Commerce|" …
 $ market           : chr [1:44435] "News" "Games" "Publishing" "Electronics"
…
 $ funding_total_usd: chr [1:44435] "1 750 000" "4 000 000" "40 000" "1 500 000"
…
 $ status           : chr [1:44435] "acquired" "operating" "operating"
"operating" …
 $ country_code     : chr [1:44435] "USA" "USA" "EST" "GBR" …
 $ state_code       : chr [1:44435] "NY" "CA" NA NA …
 $ region           : chr [1:44435] "New York City" "Los Angeles" "Tallinn"
"London" …
 $ city             : chr [1:44435] "New York" "Los Angeles" "Tallinn" "London"
…
 $ funding_rounds   : num [1:44435] 1 2 1 1 2 1 1 1 1 1 …
 $ founded_at       : chr [1:44435] "01/06/2012" NA "26/10/2012" "01/04/2011"
…
 $ founded_month    : chr [1:44435] "2012-06" NA "2012-10" "2011-04" …
 $ founded_quarter  : chr [1:44435] "2012-Q2" NA "2012-Q4" "2011-Q2" …
 $ founded_year     : num [1:44435] 2012 NA 2012 2011 2012 …
 $ first_funding_at : chr [1:44435] "30/06/2012" "04/06/2010" "09/08/2012"
"01/04/2011" …
 $ last_funding_at  : chr [1:44435] "30/06/2012" "23/09/2010" "09/08/2012"
"01/04/2011" …
```

H. What is the average number of funding rounds for the companies in **urlComps**?

[9]:

```
urlComps.mfr <- mean(urlComps$funding_rounds) #total funding rounds div by␣
 ↪number of rows
```

I. What year was the oldest company in the dataframe founded? **Hint:** If you get a value of “NA,” most likely there are missing values in this variable which preclude R from properly calculating the min & max values. Instead of running, for example, mean(urlComps$founded_year), something like this will work for determining the average:

[10]:

```
urlComps$founded_year[which.min(urlComps$founded_year)] #said na.rm was not needed, no error spit out
```

1900

J. Create another dataframe containing the companies that do not have homepage URLs. Find out the mean number of funding rounds for those companies. Compare that to the answer you recorded for problem H.

[11]:

```
missing.url <- dfComps[is.na(dfComps$homepage_url)==TRUE,] #store df that is missing urls
missing.url.mfr <- mean(missing.url$funding_rounds, na.rm=TRUE) #store mean as data for comparison

urlComps.mfr
missing.url.mfr

#companies missing homepage url had, on average, fewer funding rounds than those with a homepage url
```

1.72519410374705

1.19891664158893

1.4 Part 4: Use gsub() to clean the data.

K. The **permalink variable** in **urlComps** contains the name of each company but the names are currently preceded by the prefix “/organization/”. We can use gsub() to clean the values of this variable:

[12]:

```
urlComps$company <- gsub("/organization/", "", urlComps$permalink)
urlComps$company
#This line of code replaces every matching string of "/organization/" with "" (namely nothing) and saved in a new column
```

1. ’waywire’ 2. ’tv-communications’ 3. ’rock-your-paper’ 4. ’in-touch-network’ 5. ’n-plusn’ 6. ’club-domains’ 7. ’fox-networks’ 8. ’0-6-com’ 9. ’004-technologies’ 10. ’01games-technology’ 11. ’0xdata’ 12. ’1-2-3-listo’ 13. ’1-800-dentist’ 14. ’1-800-doctors’ 15. ’10-20-media’ 16. ’1000-corks’ 17. ’1000-markets’ 18. ’1000jobboersen-de’ 19. ’1000memories’ 20. ’1000museums-com’ 21. ’1001-menus’ 22. ’1006-tv’ 23. ’100du-tv’ 24. ’100e-com’ 25. ’100plus’ 26. ’1010data’ 27. ’10bestthings’ 28. ’10sec’ 29. ’10seconds-software’ 30. ’10six’ 31. ’10x-technologies’ 32. ’10x10-room’ 33. ’115-network-disks’ 34. ’117go’ 35. ’11i-solutions’ 36. ’12-star-survival’ 37. ’120-sports’ 38. ’121cast’ 39. ’121nexus’ 40. ’1234enter’ 41. ’123contactform’ 42. ’123people’ 43. ’1248’ 44. ’12bis’ 45. ’12return’ 46. ’12society’ 47. ’1366-technologies’ 48. ’139shop’

49. '13th-lab' 50. '140-proof' 51. '140fire' 52. 'phoneuser-network' 53. '15five' 54. '15minutesnow' 55. '169-st' 56. '170-systems' 57. '17u-cn' 58. '1871' 59. '19pay' 60. '1bib' 61. '1c--company' 62. '1calendar' 63. '1cast' 64. '1click' 65. '1cloudstar-asia' 66. '1d4-pty' 67. '1daylater' 68. '1daymakeover' 69. '1docway' 70. '1energy-systems' 71. '1eq' 72. '1jiajie' 73. '1lay' 74. '1life-healthcare' 75. '1mind' 76. '1o1media' 77. '1rebel' 78. '1ring' 79. '1sdk' 80. '1spire' 81. '1st-merchant-funding' 82. '1stdibs' 83. '1stgig-com' 84. '1world-online' 85. '2-minutes' 86. '2-pro-media-group' 87. '2-observe' 88. '20-20-gene-systems-inc' 89. '20-20-mobile' 90. '20lines' 91. '20x200' 92. '21cake-food-co' 93. '21grams' 94. '21st-century-oncology' 95. '21vianet' 96. '22nd--century-group' 97. '22seeds' 98. '2345-com' 99. '2359-media' 100. '23andme' 101. '23press' 102. '24-media-network' 103. '24-7-card' 104. '247-techies' 105. '24fundraiser-com' 106. '24h00' 107. '24m-technologies' 108. '24pagebooks' 109. '24symbols' 110. '24tidy' 111. '247-learning-private' 112. '250ok' 113. '25eight' 114. '265-network' 115. '27-bards' 116. '27-perry' 117. '280-north' 118. '28msec' 119. '29west' 120. '2adpro-media-solutions' 121. '2c2p' 122. '2can' 123. '2checkout-com' 124. '2code-online' 125. '2crisk' 126. '2d2c' 127. '2degreesmobile' 128. 'in--and-out-cash-management-software' 129. '2duche' 130. '2go-software-solutions' 131. '2heuresavant' 132. '2houses' 133. '2nd-story-software-inc' 134. '2nd-watch' 135. '2ndnature' 136. '2-n-gage-u' 137. '2nite2nite-net' 138. 'qlipso' 139. '2sms' 140. '2theloo' 141. '2threads' 142. '2u' 143. '2vancouver' 144. '2win-solutions' 145. '3-day-blinds' 146. '3-four-5-group' 147. '3-v-biosciences' 148. '30-second-showcase' 149. '303-luxury-car-service' 150. '31dover' 151. '33across' 152. '360cities' 153. '360fly-inc' 154. '360guanxi' 155. '360imaging' 156. '360incentives-com' 157. '360learning' 158. 'gazaro' 159. '360shop' 160. '360t' 161. '365-data-centers' 162. '365-docobites' 163. '365-good-teacher' 164. '365-retail-markets' 165. '365looks' 166. '365looks-coqueta-me' 167. 'house365-com' 168. '365scores' 169. '365webcall' 170. '36kr' 171. '37coins' 172. '37mhealth' 173. '382-communications' 174. '39-health' 175. '3baysover' 176. '3c-plus' 177. '3ci' 178. '3click-emr-corporation' 179. '3clogic' 180. '3d-biomatrix' 181. '3d-control-systems' 182. '3d-data' 183. '3d-eye-solutions' 184. '3d-future-vision-ii' 185. '3d-hubs' 186. '3d-industri-es' 187. '3d-operations-inc' 188. '3d-product-imaging' 189. '3d-robotics' 190. '3d-sports-technology' 191. '3d-systems' 192. '3dcart-shopping-cart-software' 193. '3derm-systems' 194. '3dim' 195. '3divi-company' 196. '3dlt-com' 197. '3dmgame' 198. '3dplusme' 199. '3dr-laboratories' 200. '3dsoc' 201. 202. 'zitra-com' 203. 'ziva-software' 204. 'zivame-com' 205. 'zivity' 206. 'zivix' 207. 'zixi' 208. 'zizerones' 209. 'zjdg-cn' 210. 'zkattter' 211. 'zkipster' 212. 'zlense' 213. 'zlien' 214. 'zlio' 215. 'zmags' 216. 'zmanda' 217. 'zmp' 218. 'zmqnw-com-cn' 219. 'znapshop' 220. 'znaptag' 221. 'znode' 222. 'zocdoc' 223. 'tyrosine-pharmaceuticals' 224. 'zocko' 225. 'zodio-com' 226. 'zoe-center-for-children' 227. 'zoe-majeste' 228. 'twt-digital' 229. 'zoeticx' 230. 'zogenix' 231. 'zogotennis' 232. 'zoidu' 233. 'zoji' 234. 'zokem' 235. 'zokos' 236. 'zola' 237. 'zola-books' 238. 'zolai-energy' 239. 'zolkc' 240. 'zollo' 241. 'zolo-technologies' 242. 'zolpy' 243. 'zolvers' 244. 'foodiebay' 245. 'zomazz' 246. 'zon-networks' 247. 'zonare-medical-systems' 248. 'zonbo-media' 249. 'zonder' 250. 'zondle' 251. 'zoned-nutrition' 252. 'zones' 253. 'zong' 254. 'zonit-structured-solutions' 255. 'zonoff' 256. 'zoobe' 257. 'zoobean' 258. 'zoodak' 259. 'zoodig' 260. 'zoodles' 261. 'zoojoo-be' 262. 'zookal' 263. 'zoom' 264. 'zoom-video-communications' 265. 'zoom-media-marketing' 266. 'zoom-technologies' 267. 'zoom-telephonics' 268. 'zoom-tv' 269. 'zoomaal' 270. 'zoomabet' 271. 'zoombu' 272. 'zoomcar-india' 273. 'zoomcare' 274. 'zoomdata' 275. 'zoomforth' 276. 'zoomin' 277. 'zoominfo' 278. 'zoomingo' 279. 'zoomio-holding' 280. 'zoomorama' 281. 'zoomph' 282. 'zoomsafer' 283. 'zoomsquare' 284. 'zoomsystems' 285. 'zoomtilt' 286. 'zoomy' 287. 'zoona' 288. 'zoondy' 289. 'zooomr' 290. 'zoop' 291. 'zoopla' 292. 'zooplus' 293. 'zooppa' 294. 'zoopshop' 295. 'zoosk' 296. 'zootcard' 297. 'zootrock' 298. 'zoove' 299. 'zooz' 300. 'zopa' 301. 'zopim' 302. 'zorap' 303. 'zorilla-research-llc' 304. 'zosano-pharma' 305. 'zostel' 306. 'zounds' 307. 'zounds--hearing-aids' 308. 'zoutons' 309. 'zouxiu' 310. 'zova' 311. 'zoweetv' 312. 'zowpow' 313. 'zoyi'

314. 'zozi' 315. 'zpower' 316. 'zqgame' 317. 'zs-genetics' 318. 'zs-pharma' 319. 'zscaler' 320. 'zsoup' 321. 'ztail' 322. 'zte9-corporation' 323. 'ztory' 324. 'zuberance' 325. 'zubican' 326. 'zubie' 327. 'zubka' 328. 'zuchem' 329. 'zuga-medical' 330. 'zuggi' 331. 'zuki' 332. 'zula' 333. 'zulahoo' 334. 'zulama' 335. 'zuldi' 336. 'zuli' 337. 'zulily' 338. 'zulu' 339. 'zumatek' 340. 'zumba--fitness' 341. 'zumbl' 342. 'zumbox' 343. 'zume-life' 344. 'zumeo-com' 345. 'zumi-networks' 346. 'zumigo' 347. 'zummzumm' 348. 'zumobi' 349. 'zumper' 350. 'zuora' 351. 'zupcat' 352. 'zuppler' 353. 'zura' 354. 'zurex-pharma' 355. 'zurff' 356. 'zurn-international-e-commerce-co-ltd' 357. 'zurrba-group' 358. 'zursh-2' 359. 'zuta-labs' 360. 'zutux' 361. 'zuu-onlnine' 362. 'zuujit' 363. 'zuuka' 364. 'zuvvu' 365. 'zuznow' 366. 'zuzuche' 367. 'zvents' 368. 'zvooq' 369. 'zweemie' 370. 'zweitgeist' 371. 'zwipe' 372. 'zwittle' 373. 'zwoor-com' 374. 'zyante' 375. 'zyb' 376. 'zyga-technology' 377. 'zygo' 378. 'zygo-corporation' 379. 'zyken-nightcove' 380. 'zykis' 381. 'zylie-the-bear' 382. 'zylun-staffing' 383. 'zyme-solutions' 384. 'zymergen' 385. 'zymetis' 386. 'zymeworks' 387. 'zympi' 388. 'zyncd' 389. 'zyncro' 390. 'zynga' 391. 'zyngenia' 392. 'zynstra' 393. 'zyomyx-inc' 394. 'zyraz-technology' 395. 'zyrra' 396. 'zytoprotec' 397. 'zzish' 398. 'zznode-science-and-technology-co-ltd' 399. 'zzzzapp-com' 400. 'a-list-games' 401. 'x'

L. Can you identify another variable which should be numeric but is currently coded as character? Use the as.numeric() function to add a new variable to **urlComps** which contains the values from the char variable as numbers. Do you notice anything about the number of NA values in this new column compared to the original "char" one?

[13]:
```r
urlComps$numeric.funding_total_usd <- as.numeric(urlComps$funding_total_usd)
urlComps$numeric.funding_total_usd #they are all NA, due to spaces in the character strings
```

```
Warning message in eval(expr, envir, enclos):
"NAs introduced by coercion"
```

1. <NA> 2. <NA> 3. <NA> 4. <NA> 5. <NA> 6. <NA> 7. <NA> 8. <NA> 9. <NA> 10. <NA> 11. <NA> 12. <NA> 13. <NA> 14. <NA> 15. <NA> 16. <NA> 17. <NA> 18. <NA> 19. <NA> 20. <NA> 21. <NA> 22. <NA> 23. <NA> 24. <NA> 25. <NA> 26. <NA> 27. <NA> 28. <NA> 29. <NA> 30. <NA> 31. <NA> 32. <NA> 33. <NA> 34. <NA> 35. <NA> 36. <NA> 37. <NA> 38. <NA> 39. <NA> 40. <NA> 41. <NA> 42. <NA> 43. <NA> 44. <NA> 45. <NA> 46. <NA> 47. <NA> 48. <NA> 49. <NA> 50. <NA> 51. <NA> 52. <NA> 53. <NA> 54. <NA> 55. <NA> 56. <NA> 57. <NA> 58. <NA> 59. <NA> 60. <NA> 61. <NA> 62. <NA> 63. <NA> 64. <NA> 65. <NA> 66. <NA> 67. <NA> 68. <NA> 69. <NA> 70. <NA> 71. <NA> 72. <NA> 73. <NA> 74. <NA> 75. <NA> 76. <NA> 77. <NA> 78. <NA> 79. <NA> 80. <NA> 81. <NA> 82. <NA> 83. <NA> 84. <NA> 85. <NA> 86. <NA> 87. <NA> 88. <NA> 89. <NA> 90. <NA> 91. <NA> 92. <NA> 93. <NA> 94. <NA> 95. <NA> 96. <NA> 97. <NA> 98. <NA> 99. <NA> 100. <NA> 101. <NA> 102. <NA> 103. <NA> 104. <NA> 105. <NA> 106. <NA> 107. <NA> 108. <NA> 109. <NA> 110. <NA> 111. <NA> 112. <NA> 113. <NA> 114. <NA> 115. <NA> 116. <NA> 117. <NA> 118. <NA> 119. <NA> 120. <NA> 121. <NA> 122. <NA> 123. <NA> 124. <NA> 125. <NA> 126. <NA> 127. <NA> 128. <NA> 129. <NA> 130. <NA> 131. <NA> 132. <NA> 133. <NA> 134. <NA> 135. <NA> 136. <NA> 137. <NA> 138. <NA> 139. <NA> 140. <NA> 141. <NA> 142. <NA> 143. <NA> 144. <NA> 145. <NA> 146. <NA> 147. <NA> 148. <NA> 149. <NA> 150. <NA> 151. <NA> 152. <NA> 153. <NA> 154. <NA> 155. <NA> 156. <NA> 157. <NA> 158. <NA> 159. <NA> 160. <NA> 161. <NA> 162. <NA> 163. <NA> 164. <NA> 165. <NA> 166. <NA> 167. <NA> 168. <NA> 169. <NA> 170. <NA> 171. <NA> 172. <NA> 173. <NA> 174. <NA> 175. <NA> 176. <NA> 177. <NA> 178. <NA> 179. <NA>

```
180. <NA> 181. <NA> 182. <NA> 183. <NA> 184. <NA> 185. <NA> 186. <NA> 187. <NA>
188. <NA> 189. <NA> 190. <NA> 191. <NA> 192. <NA> 193. <NA> 194. <NA> 195. <NA>
196. <NA> 197. <NA> 198. <NA> 199. <NA> 200. <NA> 201.  202. <NA> 203. <NA>
204. <NA> 205. <NA> 206. <NA> 207. <NA> 208. <NA> 209. <NA> 210. <NA> 211. <NA>
212. <NA> 213. <NA> 214. <NA> 215. <NA> 216. <NA> 217. <NA> 218. <NA> 219. <NA>
220. <NA> 221. <NA> 222. <NA> 223. <NA> 224. <NA> 225. <NA> 226. <NA> 227. <NA>
228. <NA> 229. <NA> 230. <NA> 231. <NA> 232. <NA> 233. <NA> 234. <NA> 235. <NA>
236. <NA> 237. <NA> 238. <NA> 239. <NA> 240. <NA> 241. <NA> 242. <NA> 243. <NA>
244. <NA> 245. <NA> 246. <NA> 247. <NA> 248. <NA> 249. <NA> 250. <NA> 251. <NA>
252. <NA> 253. <NA> 254. <NA> 255. <NA> 256. <NA> 257. <NA> 258. <NA> 259. <NA>
260. <NA> 261. <NA> 262. <NA> 263. <NA> 264. <NA> 265. <NA> 266. <NA> 267. <NA>
268. <NA> 269. <NA> 270. <NA> 271. <NA> 272. <NA> 273. <NA> 274. <NA> 275. <NA>
276. <NA> 277. <NA> 278. <NA> 279. <NA> 280. <NA> 281. <NA> 282. <NA> 283. <NA>
284. <NA> 285. <NA> 286. <NA> 287. <NA> 288. <NA> 289. <NA> 290. <NA> 291. <NA>
292. <NA> 293. <NA> 294. <NA> 295. <NA> 296. <NA> 297. <NA> 298. <NA> 299. <NA>
300. <NA> 301. <NA> 302. <NA> 303. <NA> 304. <NA> 305. <NA> 306. <NA> 307. <NA>
308. <NA> 309. <NA> 310. <NA> 311. <NA> 312. <NA> 313. <NA> 314. <NA> 315. <NA>
316. <NA> 317. <NA> 318. <NA> 319. <NA> 320. <NA> 321. <NA> 322. <NA> 323. <NA>
324. <NA> 325. <NA> 326. <NA> 327. <NA> 328. <NA> 329. <NA> 330. <NA> 331. <NA>
332. <NA> 333. <NA> 334. <NA> 335. <NA> 336. <NA> 337. <NA> 338. <NA> 339. <NA>
340. <NA> 341. <NA> 342. <NA> 343. <NA> 344. <NA> 345. <NA> 346. <NA> 347. <NA>
348. <NA> 349. <NA> 350. <NA> 351. <NA> 352. <NA> 353. <NA> 354. <NA> 355. <NA>
356. <NA> 357. <NA> 358. <NA> 359. <NA> 360. <NA> 361. <NA> 362. <NA> 363. <NA>
364. <NA> 365. <NA> 366. <NA> 367. <NA> 368. <NA> 369. <NA> 370. <NA> 371. <NA>
372. <NA> 373. <NA> 374. <NA> 375. <NA> 376. <NA> 377. <NA> 378. <NA> 379. <NA>
380. <NA> 381. <NA> 382. <NA> 383. <NA> 384. <NA> 385. <NA> 386. <NA> 387. <NA>
388. <NA> 389. <NA> 390. <NA> 391. <NA> 392. <NA> 393. <NA> 394. <NA> 395. <NA>
396. <NA> 397. <NA> 398. <NA> 399. <NA> 400. <NA> 401. <NA>
```

M. To ensure the char values are converted correctly, we first need to remove the spaces between the numbers in the variable. Use the gsub() command to do that. Check if this works:

```
[115]: install.packages("stringi") #DO NOT RUN, BREAKS COMMANDS LATER
```

```
Updating HTML index of packages in '.Library'

Making 'packages.html' …
 done
```

```
[14]: library(stringi)
```

```
[16]: urlComps$numeric.funding_total_usd <-␣
       ↪stri_replace_all_charclass(urlComps$funding_total_usd,"\\p{WHITE_SPACE}", "")

      urlComps$numeric.funding_total_usd
      #Does this variable look better than the one we created in L? Explain in a␣
       ↪comment:
```

```
#this looks much better, as each character string has been converted to a␣
↪numeral, with some left over as dashes
```

1. ’1750000’ 2. ’4000000’ 3. ’40000’ 4. ’1500000’ 5. ’1200000’ 6. ’7000000’ 7. ’4912393’ 8. ’2000000’
9. ’-’ 10. ’41250’ 11. ’10600000’ 12. ’40000’ 13. ’-’ 14. ’1750000’ 15. ’2050000’ 16. ’40000’ 17. ’500000’
18. ’-’ 19. ’2535000’ 20. ’4962651’ 21. ’4059079’ 22. ’10000000’ 23. ’3000000’ 24. ’3000000’
25. ’1250000’ 26. ’35000000’ 27. ’50000’ 28. ’1600000’ 29. ’100000’ 30. ’-’ 31. ’3000000’ 32. ’77500’
33. ’-’ 34. ’20000000’ 35. ’1800000’ 36. ’-’ 37. ’-’ 38. ’270000’ 39. ’794000’ 40. ’650267’ 41. ’-’
42. ’-’ 43. ’378812’ 44. ’130636’ 45. ’619494’ 46. ’-’ 47. ’66450000’ 48. ’-’ 49. ’700000’ 50. ’5500000’
51. ’500000’ 52. ’6204822’ 53. ’3400000’ 54. ’200000’ 55. ’50000’ 56. ’14000000’ 57. ’84440319’
58. ’2500000’ 59. ’9478511’ 60. ’-’ 61. ’200000000’ 62. ’40000’ 63. ’-’ 64. ’-’ 65. ’-’ 66. ’40000’ 67. ’43811’
68. ’50000’ 69. ’-’ 70. ’1450000’ 71. ’2100000’ 72. ’4163132’ 73. ’170000’ 74. ’30000000’ 75. ’500000’
76. ’-’ 77. ’2572969’ 78. ’-’ 79. ’156000’ 80. ’30000’ 81. ’10000000’ 82. ’117000000’ 83. ’750000’
84. ’1500000’ 85. ’1660000’ 86. ’41250’ 87. ’204189’ 88. ’150000’ 89. ’29750000’ 90. ’1236454’
91. ’2800000’ 92. ’1464128’ 93. ’2580000’ 94. ’325000000’ 95. ’100000000’ 96. ’14984750’ 97. ’250000’
98. ’1610541’ 99. ’615000’ 100. ’111949900’ 101. ’50000’ 102. ’1209701’ 103. ’3452941’ 104. ’600000’
105. ’400000’ 106. ’-’ 107. ’16000000’ 108. ’50000’ 109. ’-’ 110. ’1856561’ 111. ’4000000’ 112. ’200000’
113. ’25000’ 114. ’2000000’ 115. ’-’ 116. ’118000’ 117. ’250000’ 118. ’5844811’ 119. ’-’ 120. ’-’
121. ’3000000’ 122. ’7000000’ 123. ’60000000’ 124. ’-’ 125. ’383000’ 126. ’2941000’ 127. ’165000000’
128. ’250000’ 129. ’-’ 130. ’275000’ 131. ’33457’ 132. ’966728’ 133. ’85000000’ 134. ’27644420’
135. ’20000’ 136. ’58000’ 137. ’6000’ 138. ’-’ 139. ’939000’ 140. ’-’ 141. ’300000’ 142. ’95875000’
143. ’80000’ 144. ’-’ 145. ’-’ 146. ’-’ 147. ’78089000’ 148. ’25000’ 149. ’-’ 150. ’2274716’ 151. ’25654560’
152. ’-’ 153. ’17800000’ 154. ’5000000’ 155. ’1000000’ 156. ’7650000’ 157. ’1543920’ 158. ’4100000’
159. ’1623640’ 160. ’-’ 161. ’16000000’ 162. ’-’ 163. ’-’ 164. ’-’ 165. ’54096’ 166. ’18951’ 167. ’2203975’
168. ’6700000’ 169. ’-’ 170. ’1573976’ 171. ’25000’ 172. ’1629549’ 173. ’1600000’ 174. ’3750000’
175. ’1719583’ 176. ’75000’ 177. ’133620’ 178. ’500000’ 179. ’5740000’ 180. ’1465000’ 181. ’28000’
182. ’250000’ 183. ’380000’ 184. ’503000’ 185. ’4500000’ 186. ’1200000’ 187. ’7500’ 188. ’240000’
189. ’35000000’ 190. ’909940’ 191. ’19500000’ 192. ’1250000’ 193. ’-’ 194. ’50000’ 195. ’2670000’
196. ’530000’ 197. ’732064’ 198. ’200000’ 199. ’7436500’ 200. ’2065000’ 201. 202. ’-’ 203. ’1700000’
204. ’9000000’ 205. ’8000000’ 206. ’700000’ 207. ’4000000’ 208. ’271028’ 209. ’-’ 210. ’1500000’ 211. ’-’
212. ’325000’ 213. ’450000’ 214. ’4000000’ 215. ’20849997’ 216. ’13000000’ 217. ’-’ 218. ’164744’
219. ’50000’ 220. ’2000000’ 221. ’1700000’ 222. ’97940002’ 223. ’100000’ 224. ’200000’ 225. ’1600500’
226. ’-’ 227. ’-’ 228. ’800000’ 229. ’580000’ 230. ’167062156’ 231. ’100000’ 232. ’48600’ 233. ’1500000’
234. ’2388400’ 235. ’210000’ 236. ’3250000’ 237. ’5500000’ 238. ’-’ 239. ’638550’ 240. ’900000’
241. ’19000008’ 242. ’-’ 243. ’65000’ 244. ’53800000’ 245. ’2040342’ 246. ’7000000’ 247. ’45693276’
248. ’-’ 249. ’1200000’ 250. ’-’ 251. ’-’ 252. ’700000’ 253. ’15000000’ 254. ’500000’ 255. ’3800000’
256. ’1286600’ 257. ’1242619’ 258. ’580000’ 259. ’-’ 260. ’2600000’ 261. ’-’ 262. ’2060000’
263. ’1600000’ 264. ’15500000’ 265. ’30000000’ 266. ’7926240’ 267. ’1000000’ 268. ’150000’ 269. ’-’
270. ’-’ 271. ’-’ 272. ’2900000’ 273. ’-’ 274. ’22200000’ 275. ’1300000’ 276. ’21500000’ 277. ’7000000’
278. ’2475000’ 279. ’7130000’ 280. ’2150110’ 281. ’500000’ 282. ’3350000’ 283. ’-’ 284. ’87000000’
285. ’88000’ 286. ’-’ 287. ’4000000’ 288. ’75000’ 289. ’50000’ 290. ’750000’ 291. ’13387373’ 292. ’-’
293. ’-’ 294. ’20000’ 295. ’61600000’ 296. ’750000’ 297. ’110000’ 298. ’59139710’ 299. ’16500000’
300. ’56628771’ 301. ’392000’ 302. ’2251000’ 303. ’-’ 304. ’142355891’ 305. ’1000000’ 306. ’32166307’
307. ’10000000’ 308. ’3300000’ 309. ’120000000’ 310. ’-’ 311. ’900000’ 312. ’-’ 313. ’650000’
314. ’21300000’ 315. ’133437465’ 316. ’4220018’ 317. ’2444963’ 318. ’122987972’ 319. ’38000000’
320. ’90000’ 321. ’1500000’ 322. ’2059308’ 323. ’-’ 324. ’12000000’ 325. ’-’ 326. ’18000000’
327. ’3896936’ 328. ’150000’ 329. ’350000’ 330. ’40000’ 331. ’30000’ 332. ’4000000’ 333. ’100000’

```
334. ’200000’ 335. ’80000’ 336. ’1650000’ 337. ’138600000’ 338. ’1070000’ 339. ’150000’ 340. ’-’
341. ’140000’ 342. ’28300000’ 343. ’1700000’ 344. ’100000’ 345. ’500000’ 346. ’7720000’ 347. ’50000’
348. ’12000000’ 349. ’8200000’ 350. ’127500000’ 351. ’700000’ 352. ’1240000’ 353. ’-’ 354. ’11146457’
355. ’-’ 356. ’-’ 357. ’-’ 358. ’25000’ 359. ’500000’ 360. ’1503926’ 361. ’1000000’ 362. ’2700000’
363. ’-’ 364. ’12000’ 365. ’650000’ 366. ’1515151’ 367. ’55200000’ 368. ’20000000’ 369. ’250000’
370. ’651000’ 371. ’3500000’ 372. ’190000’ 373. ’100000’ 374. ’870000’ 375. ’3845100’ 376. ’55750000’
377. ’-’ 378. ’9000000’ 379. ’3384225’ 380. ’800000’ 381. ’75000’ 382. ’666154’ 383. ’12039999’ 384. ’-’
385. ’2257464’ 386. ’38900000’ 387. ’-’ 388. ’-’ 389. ’3805520’ 390. ’866550786’ 391. ’25000000’
392. ’14750000’ 393. ’34275015’ 394. ’15419877’ 395. ’1510500’ 396. ’2686600’ 397. ’320000’
398. ’1587301’ 399. ’97398’ 400. ’9300000’ 401. ’45000000’
```

N. You are now ready to convert urlComps$funding_new to numeric using as.numeric() again. Calculate the average funding amount for **urlComps**. If you get “NA,” try using the na.rm=TRUE argument from problem I.

[18]:

```r
urlComps$numeric.funding_total_usd <- as.numeric(urlComps$numeric.
↪funding_total_usd) #overwrite all values as num
mean(urlComps$numeric.funding_total_usd, na.rm=TRUE) #find mean, removing the␣
↪NA value
```

```
18321551.4738696
```

1.5 Part 5: Create a function to automate the process from L-N:

O. The following function should work most of the time. Make sure to run this code before trying to test it. That is how you make the new function known to R. **Add comments to each line explaining what it does:**

[21]:

```r
convertCharToNum <- function(char_string) { #initializes function with parameter
  step1 <-␣
↪stri_replace_all_charclass(urlComps$funding_total_usd,"\\p{WHITE_SPACE}",␣
↪"") #removes spaces from all strings
  step2 <- as.numeric(step1) #coerces space-less strings to numeric
  return(step2) #returns the result of step 2
}
```

P. Run your new function on the **funding_total_usd** variable in **urlComps**:

[24]:

```r
convertCharToNum(urlComps$funding_total_usd) #test function
```

```
Warning message in convertCharToNum(urlComps$funding_total_usd):
"NAs introduced by coercion"
```

```
1. 1750000 2. 4e+06 3. 40000 4. 1500000 5. 1200000 6. 7e+06 7. 4912393 8. 2e+06 9. <NA>
10. 41250 11. 10600000 12. 40000 13. <NA> 14. 1750000 15. 2050000 16. 40000 17. 5e+05
18. <NA> 19. 2535000 20. 4962651 21. 4059079 22. 1e+07 23. 3e+06 24. 3e+06 25. 1250000
26. 3.5e+07 27. 50000 28. 1600000 29. 1e+05 30. <NA> 31. 3e+06 32. 77500 33. <NA>
34. 2e+07 35. 1800000 36. <NA> 37. <NA> 38. 270000 39. 794000 40. 650267 41. <NA>
42. <NA> 43. 378812 44. 130636 45. 619494 46. <NA> 47. 66450000 48. <NA> 49. 7e+05
50. 5500000 51. 5e+05 52. 6204822 53. 3400000 54. 2e+05 55. 50000 56. 1.4e+07 57. 84440319
```

58. 2500000 59. 9478511 60. <NA> 61. 2e+08 62. 40000 63. <NA> 64. <NA> 65. <NA>
66. 40000 67. 43811 68. 50000 69. <NA> 70. 1450000 71. 2100000 72. 4163132 73. 170000
74. 3e+07 75. 5e+05 76. <NA> 77. 2572969 78. <NA> 79. 156000 80. 30000 81. 1e+07
82. 1.17e+08 83. 750000 84. 1500000 85. 1660000 86. 41250 87. 204189 88. 150000 89. 29750000
90. 1236454 91. 2800000 92. 1464128 93. 2580000 94. 3.25e+08 95. 1e+08 96. 14984750 97. 250000
98. 1610541 99. 615000 100. 111949900 101. 50000 102. 1209701 103. 3452941 104. 6e+05
105. 4e+05 106. <NA> 107. 1.6e+07 108. 50000 109. <NA> 110. 1856561 111. 4e+06 112. 2e+05
113. 25000 114. 2e+06 115. <NA> 116. 118000 117. 250000 118. 5844811 119. <NA> 120. <NA>
121. 3e+06 122. 7e+06 123. 6e+07 124. <NA> 125. 383000 126. 2941000 127. 1.65e+08 128. 250000
129. <NA> 130. 275000 131. 33457 132. 966728 133. 8.5e+07 134. 27644420 135. 20000 136. 58000
137. 6000 138. <NA> 139. 939000 140. <NA> 141. 3e+05 142. 95875000 143. 80000 144. <NA>
145. <NA> 146. <NA> 147. 78089000 148. 25000 149. <NA> 150. 2274716 151. 25654560
152. <NA> 153. 17800000 154. 5e+06 155. 1e+06 156. 7650000 157. 1543920 158. 4100000
159. 1623640 160. <NA> 161. 1.6e+07 162. <NA> 163. <NA> 164. <NA> 165. 54096 166. 18951
167. 2203975 168. 6700000 169. <NA> 170. 1573976 171. 25000 172. 1629549 173. 1600000
174. 3750000 175. 1719583 176. 75000 177. 133620 178. 5e+05 179. 5740000 180. 1465000 181. 28000
182. 250000 183. 380000 184. 503000 185. 4500000 186. 1200000 187. 7500 188. 240000 189. 3.5e+07
190. 909940 191. 19500000 192. 1250000 193. <NA> 194. 50000 195. 2670000 196. 530000
197. 732064 198. 2e+05 199. 7436500 200. 2065000 201. 202. <NA> 203. 1700000 204. 9e+06
205. 8e+06 206. 7e+05 207. 4e+06 208. 271028 209. <NA> 210. 1500000 211. <NA> 212. 325000
213. 450000 214. 4e+06 215. 20849997 216. 1.3e+07 217. <NA> 218. 164744 219. 50000 220. 2e+06
221. 1700000 222. 97940002 223. 1e+05 224. 2e+05 225. 1600500 226. <NA> 227. <NA> 228. 8e+05
229. 580000 230. 167062156 231. 1e+05 232. 48600 233. 1500000 234. 2388400 235. 210000
236. 3250000 237. 5500000 238. <NA> 239. 638550 240. 9e+05 241. 19000008 242. <NA> 243. 65000
244. 53800000 245. 2040342 246. 7e+06 247. 45693276 248. <NA> 249. 1200000 250. <NA>
251. <NA> 252. 7e+05 253. 1.5e+07 254. 5e+05 255. 3800000 256. 1286600 257. 1242619
258. 580000 259. <NA> 260. 2600000 261. <NA> 262. 2060000 263. 1600000 264. 15500000
265. 3e+07 266. 7926240 267. 1e+06 268. 150000 269. <NA> 270. <NA> 271. <NA> 272. 2900000
273. <NA> 274. 22200000 275. 1300000 276. 21500000 277. 7e+06 278. 2475000 279. 7130000
280. 2150110 281. 5e+05 282. 3350000 283. <NA> 284. 8.7e+07 285. 88000 286. <NA> 287. 4e+06
288. 75000 289. 50000 290. 750000 291. 13387373 292. <NA> 293. <NA> 294. 20000 295. 61600000
296. 750000 297. 110000 298. 59139710 299. 16500000 300. 56628771 301. 392000 302. 2251000
303. <NA> 304. 142355891 305. 1e+06 306. 32166307 307. 1e+07 308. 3300000 309. 1.2e+08
310. <NA> 311. 9e+05 312. <NA> 313. 650000 314. 21300000 315. 133437465 316. 4220018
317. 2444963 318. 122987972 319. 3.8e+07 320. 90000 321. 1500000 322. 2059308 323. <NA>
324. 1.2e+07 325. <NA> 326. 1.8e+07 327. 3896936 328. 150000 329. 350000 330. 40000 331. 30000
332. 4e+06 333. 1e+05 334. 2e+05 335. 80000 336. 1650000 337. 138600000 338. 1070000 339. 150000
340. <NA> 341. 140000 342. 28300000 343. 1700000 344. 1e+05 345. 5e+05 346. 7720000 347. 50000
348. 1.2e+07 349. 8200000 350. 127500000 351. 7e+05 352. 1240000 353. <NA> 354. 11146457
355. <NA> 356. <NA> 357. <NA> 358. 25000 359. 5e+05 360. 1503926 361. 1e+06 362. 2700000
363. <NA> 364. 12000 365. 650000 366. 1515151 367. 55200000 368. 2e+07 369. 250000 370. 651000
371. 3500000 372. 190000 373. 1e+05 374. 870000 375. 3845100 376. 55750000 377. <NA>
378. 9e+06 379. 3384225 380. 8e+05 381. 75000 382. 666154 383. 12039999 384. <NA> 385. 2257464
386. 38900000 387. <NA> 388. <NA> 389. 3805520 390. 866550786 391. 2.5e+07 392. 14750000
393. 34275015 394. 15419877 395. 1510500 396. 2686600 397. 320000 398. 1587301 399. 97398
400. 9300000 401. 4.5e+07

Q. Assign the result of P to a variable in the dataframe:

[25]:

```
urlComps$funding_total_num <- convertCharToNum(urlComps$funding_total_usd) #uses function created to convert faster
```

```
Warning message in convertCharToNum(urlComps$funding_total_usd):
"NAs introduced by coercion"
```

Calculate the average of this new variable (you may need to use the rm.na=TRUE argument again). Is it the same as the value you got in N? Explain.

[27]:

```
mean(urlComps$numeric.funding_total_usd, na.rm=TRUE) #previous answer moved down to compare
mean(urlComps$funding_total_num, na.rm=TRUE) #new mean calculated

#yes, it is the same value I got in N because a function isn't changing the numeral values of the strings,
#it is only formatting them for me
```

```
18321551.4738696
```

```
18321551.4738696
```

[ ]:

```

```