

Name - NetID

Homework 1

Homework Instructions

For questions that require code, please create a code chunk directly below the question and type your code there. Your knitted pdf will show both your code and your output. You are encouraged to knit your file as you work to check that your coding and formatting is done so appropriately.

For written responses or multiple choice questions, please bold your (selected) answer.

Grading Details

All questions will be graded full credit (1 point), half credit (0.5 point) or no credit (0 points).

Full credit responses should have the correct response and the appropriate code. Half credit responses will have a reasonable attempt (typically no more than one small error or oversight), and no credit responses will be blank or attempts or attempts with at least one significant error.

Exercise 1

```
# starter  
x = 1:100
```

Calculate

$$\sum_{i=1}^n \ln(x_i).$$

That is, sum the log of each element of **x**.

Hint: The simplest way to do this is with two functions, one nested within the other.

```
# solution  
sum(log(x))
```

```
## [1] 363.7394
```

Exercise 2

Create a vector that spans the range 12 to 15.4, and that goes by intervals of 0.2. Name this vector **h**

Please be sure to also run **h** at the end so that your vector is printed.

```
# solution  
h = seq(from = 12, to = 15.4, by = 0.2)  
# or alternatively...  
#h = seq(12, 15.4, 0.2)  
  
h
```

```
## [1] 12.0 12.2 12.4 12.6 12.8 13.0 13.2 13.4 13.6 13.8 14.0 14.2 14.4 14.6 14.8
## [16] 15.0 15.2 15.4
```

Exercise 3

```
#starter
labels = c("A","B","C")
```

Create a new vector named `full_labels` that repeats the sequence `labels` until this new vector has a length of 90

Print `full_labels`

```
#solution
new_labels = rep(labels, 30)
new_labels
```

```
## [1] "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C" "A"
## [20] "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B"
## [39] "C" "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C"
## [58] "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C" "A"
## [77] "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C" "A" "B" "C"
```

Exercise 4

```
# starter
x = 1:100
```

Create a new vector `y`, which adds 5 to the elements stored in odd indices of `x` and subtracts 10 from the elements stored even indices of `x`.

Hint: Try creating a second vector and then take advantage of vectorized operations to complete this arithmetic.

Finally, print `y`

```
#solution
add = c(5,-10)

z = rep(add, 50)
#Note that this step above isn't actually necessary...we could just directly do y = x + add (since R wi
y = x + z

y
```

```
## [1] 6 -8 8 -6 10 -4 12 -2 14 0 16 2 18 4 20 6 22 8
## [19] 24 10 26 12 28 14 30 16 32 18 34 20 36 22 38 24 40 26
## [37] 42 28 44 30 46 32 48 34 50 36 52 38 54 40 56 42 58 44
## [55] 60 46 62 48 64 50 66 52 68 54 70 56 72 58 74 60 76 62
## [73] 78 64 80 66 82 68 84 70 86 72 88 74 90 76 92 78 94 80
## [91] 96 82 98 84 100 86 102 88 104 90
```

Exercise 5

Which of the following objects in R is technically a vector, but each entry of that vector is a set of one or more entries, and each element can be of a different type?

Please **bold** your answer

A Matrix

A Function

A List

A Data Frame

A Tibble

Exercise 6

```
# starter
set.seed(42)
data = data.frame(
  x = rep("A",8),
  y = c(1, 1, 2, 3, 5, 8, 13, 21),
  z = sample(8)
)
data
```

```
##   x  y z
## 1 A  1 1
## 2 A  1 5
## 3 A  2 8
## 4 A  3 6
## 5 A  5 2
## 6 A  8 4
## 7 A 13 3
## 8 A 21 7
```

The above code block creates a data frame, stored as the variable `data`.

Extract the third entry of the vector `y`.

Hint: There is more than one way to do this. Any method that produces the correct result is fine.

```
# solution
data$y[3]
```

```
## [1] 2
```

```
#or alternatively...
data[[2]][3]
```

```
## [1] 2
```

Exercise 7

```
# starter
set.seed(42)
a = rnorm(100, mean = 0, sd = 10)
```

Write a function called `f` that has a single argument `input`. We want this function to input a vector (that may have a mixture of positive and negative values) and produce a vector where all negative elements are replaced with 0.

Hint: The `ifelse()` function could be useful here.

Write and save your function, then run your function with vector `a` as your input and save the result to a new variable named `x_pos`, and then extract the first five elements of `x_pos`

```
#solution
f = function(input = 42) {
  ifelse(input < 0, 0, input)
}

x_pos = f(input = a)
x_pos[1:5]
```

```
## [1] 13.709584 0.000000 3.631284 6.328626 4.042683
```

Exercise 8

Create three vectors as follows. Each should have a length of 30.

`x1`: Each element should be the value 1 `x2`: The first 30 square numbers, starting from 1 (so 1, 4, 9, etc.)

`x3`: A sequence running from 2 to 60 by 2's.

Then create a new vector named `y`, such that each entry of `y` is the sum of the entries of `x1`, `x2`, and `x3` for that same position.

Print `y`

```
# solution
x1 = rep(1, 30)
x2 = (1:30) ^ 2
x3 = seq(2, 60, 2)
y = x1 + x2 + x3
y
```

```
## [1] 4 9 16 25 36 49 64 81 100 121 144 169 196 225 256 289 324 361 400
## [20] 441 484 529 576 625 676 729 784 841 900 961
```

Exercise 9

(Continued from Exercise 8) Create a matrix `X` with columns `x1`, `x2`, and `x3`.

Hint: Use the `column bind` function.

Report the sum of the elements in row 17 of this matrix.

```
# solution
X = cbind(x1, x2, x3)

sum(X[17,])
```

```
## [1] 324
```

```
#or alternatively
```

```
X[17,1] + X[17,2] + X[17,3]
```

```
## x1
## 324
```

Exercise 10

```
#starter_0
#If you do not have this already, install ggplot2
#install.packages("ggplot2")
```

```
#starter
library(ggplot2)
mpg
```

```
## # A tibble: 234 x 11
##   manufacturer model   displ  year  cyl trans  drv    cty   hwy fl    class
##   <chr>          <chr>   <dbl> <int> <int> <chr>  <chr> <int> <int> <chr> <chr>
## 1 audi          a4       1.8  1999    4 auto(l~ f      18    29 p     comp~
## 2 audi          a4       1.8  1999    4 manual~ f      21    29 p     comp~
## 3 audi          a4       2    2008    4 manual~ f      20    31 p     comp~
## 4 audi          a4       2    2008    4 auto(a~ f      21    30 p     comp~
## 5 audi          a4       2.8  1999    6 auto(l~ f      16    26 p     comp~
## 6 audi          a4       2.8  1999    6 manual~ f      18    26 p     comp~
## 7 audi          a4       3.1  2008    6 auto(a~ f      18    27 p     comp~
## 8 audi          a4 quat~  1.8  1999    4 manual~ 4      18    26 p     comp~
## 9 audi          a4 quat~  1.8  1999    4 auto(l~ 4      16    25 p     comp~
## 10 audi         a4 quat~  2    2008    4 manual~ 4      20    28 p     comp~
## # ... with 224 more rows
```

mpg is a tibble (a data frame with some additionally nice properties) displaying information about 38 car models released between 1999 and 2008. Find more information about this data and the variables here:

```
?mpg
```

Create a subsetting tibble named `mpg_eff` that contains only vehicles with a highway fuel efficiency of 30 or above.

Hint: Use bracket notation, and remember you just need to filter rows, but you are selecting all of the columns.

Then use an R command to report on how many observations are in `mpg_eff`

```
#solution
mpg_eff = mpg[mpg$hwy >= 30,]
nrow(mpg_eff)
```

```
## [1] 26
```

Exercise 11 (Bonus)

(Continued from Exercise 10) Which manufacturer makes the vehicle with the highest **city** fuel efficiency? (Yes, you should use R code to find it!)

Hint: For full credit, your answer should be produced from only one line of code, but you can still get half credit if done across several lines of code.

```
#solution
mpg[mpg$cty == max(mpg$cty),1]
```

```
## # A tibble: 1 x 1
##   manufacturer
##   <chr>
## 1 volkswagen
```

```
#or alternatively
```

```
mpg[which.max(mpg$cty),1]
```

```
## # A tibble: 1 x 1
##   manufacturer
##   <chr>
## 1 volkswagen
```

Exercise 12 (Bonus)

(Continued from Exercises 10 and 11) Let's say that a local law requires city vehicles to have a city mpg rate of at least 20 to be sold within city limits. Create a function that will produce TRUE if the inputted value is 20 or above, and FALSE if the inputted value is below 20.

Create the function, run it with the mpg data and name the output **pass**, and then column bind this new pass variable to the mpg data and call this new data **mpg_pass**.

You are encouraged to print the results for your own sake to check, but please delete the code to avoid printing the full results in your knitted pdf

Finally, run a subset of the mpg_pass data that only includes vehicles that don't pass the standard. No need to save it under a new name first—just print it.

```
#solution
g = function(mpg) {
  ifelse(mpg < 20, FALSE, TRUE)
}

pass = g(mpg = mpg$cty)
mpg_pass = cbind(mpg, pass)
mpg_pass[mpg_pass$pass == FALSE,]
```

##	manufacturer	model	displ	year	cyl	trans	drv	cty	hwy
## 1	audi	a4	1.8	1999	4	auto(15)	f	18	29
## 5	audi	a4	2.8	1999	6	auto(15)	f	16	26
## 6	audi	a4	2.8	1999	6	manual(m5)	f	18	26
## 7	audi	a4	3.1	2008	6	auto(av)	f	18	27
## 8	audi	a4 quattro	1.8	1999	4	manual(m5)	4	18	26
## 9	audi	a4 quattro	1.8	1999	4	auto(15)	4	16	25
## 11	audi	a4 quattro	2.0	2008	4	auto(s6)	4	19	27
## 12	audi	a4 quattro	2.8	1999	6	auto(15)	4	15	25
## 13	audi	a4 quattro	2.8	1999	6	manual(m5)	4	17	25
## 14	audi	a4 quattro	3.1	2008	6	auto(s6)	4	17	25
## 15	audi	a4 quattro	3.1	2008	6	manual(m6)	4	15	25
## 16	audi	a6 quattro	2.8	1999	6	auto(15)	4	15	24
## 17	audi	a6 quattro	3.1	2008	6	auto(s6)	4	17	25
## 18	audi	a6 quattro	4.2	2008	8	auto(s6)	4	16	23
## 19	chevrolet	c1500 suburban 2wd	5.3	2008	8	auto(14)	r	14	20
## 20	chevrolet	c1500 suburban 2wd	5.3	2008	8	auto(14)	r	11	15
## 21	chevrolet	c1500 suburban 2wd	5.3	2008	8	auto(14)	r	14	20
## 22	chevrolet	c1500 suburban 2wd	5.7	1999	8	auto(14)	r	13	17
## 23	chevrolet	c1500 suburban 2wd	6.0	2008	8	auto(14)	r	12	17
## 24	chevrolet	corvette	5.7	1999	8	manual(m6)	r	16	26
## 25	chevrolet	corvette	5.7	1999	8	auto(14)	r	15	23
## 26	chevrolet	corvette	6.2	2008	8	manual(m6)	r	16	26
## 27	chevrolet	corvette	6.2	2008	8	auto(s6)	r	15	25
## 28	chevrolet	corvette	7.0	2008	8	manual(m6)	r	15	24
## 29	chevrolet	k1500 tahoe 4wd	5.3	2008	8	auto(14)	4	14	19
## 30	chevrolet	k1500 tahoe 4wd	5.3	2008	8	auto(14)	4	11	14
## 31	chevrolet	k1500 tahoe 4wd	5.7	1999	8	auto(14)	4	11	15
## 32	chevrolet	k1500 tahoe 4wd	6.5	1999	8	auto(14)	4	14	17
## 33	chevrolet	malibu	2.4	1999	4	auto(14)	f	19	27
## 35	chevrolet	malibu	3.1	1999	6	auto(14)	f	18	26
## 36	chevrolet	malibu	3.5	2008	6	auto(14)	f	18	29
## 37	chevrolet	malibu	3.6	2008	6	auto(s6)	f	17	26
## 38	dodge	caravan 2wd	2.4	1999	4	auto(13)	f	18	24
## 39	dodge	caravan 2wd	3.0	1999	6	auto(14)	f	17	24
## 40	dodge	caravan 2wd	3.3	1999	6	auto(14)	f	16	22
## 41	dodge	caravan 2wd	3.3	1999	6	auto(14)	f	16	22
## 42	dodge	caravan 2wd	3.3	2008	6	auto(14)	f	17	24
## 43	dodge	caravan 2wd	3.3	2008	6	auto(14)	f	17	24
## 44	dodge	caravan 2wd	3.3	2008	6	auto(14)	f	11	17
## 45	dodge	caravan 2wd	3.8	1999	6	auto(14)	f	15	22
## 46	dodge	caravan 2wd	3.8	1999	6	auto(14)	f	15	21
## 47	dodge	caravan 2wd	3.8	2008	6	auto(16)	f	16	23
## 48	dodge	caravan 2wd	4.0	2008	6	auto(16)	f	16	23
## 49	dodge	dakota pickup 4wd	3.7	2008	6	manual(m6)	4	15	19
## 50	dodge	dakota pickup 4wd	3.7	2008	6	auto(14)	4	14	18
## 51	dodge	dakota pickup 4wd	3.9	1999	6	auto(14)	4	13	17
## 52	dodge	dakota pickup 4wd	3.9	1999	6	manual(m5)	4	14	17
## 53	dodge	dakota pickup 4wd	4.7	2008	8	auto(15)	4	14	19
## 54	dodge	dakota pickup 4wd	4.7	2008	8	auto(15)	4	14	19
## 55	dodge	dakota pickup 4wd	4.7	2008	8	auto(15)	4	9	12
## 56	dodge	dakota pickup 4wd	5.2	1999	8	manual(m5)	4	11	17
## 57	dodge	dakota pickup 4wd	5.2	1999	8	auto(14)	4	11	15
## 58	dodge	durango 4wd	3.9	1999	6	auto(14)	4	13	17

## 59	dodge	durango	4wd	4.7	2008	8	auto(15)	4	13	17
## 60	dodge	durango	4wd	4.7	2008	8	auto(15)	4	9	12
## 61	dodge	durango	4wd	4.7	2008	8	auto(15)	4	13	17
## 62	dodge	durango	4wd	5.2	1999	8	auto(14)	4	11	16
## 63	dodge	durango	4wd	5.7	2008	8	auto(15)	4	13	18
## 64	dodge	durango	4wd	5.9	1999	8	auto(14)	4	11	15
## 65	dodge	ram 1500	pickup 4wd	4.7	2008	8	manual(m6)	4	12	16
## 66	dodge	ram 1500	pickup 4wd	4.7	2008	8	auto(15)	4	9	12
## 67	dodge	ram 1500	pickup 4wd	4.7	2008	8	auto(15)	4	13	17
## 68	dodge	ram 1500	pickup 4wd	4.7	2008	8	auto(15)	4	13	17
## 69	dodge	ram 1500	pickup 4wd	4.7	2008	8	manual(m6)	4	12	16
## 70	dodge	ram 1500	pickup 4wd	4.7	2008	8	manual(m6)	4	9	12
## 71	dodge	ram 1500	pickup 4wd	5.2	1999	8	auto(14)	4	11	15
## 72	dodge	ram 1500	pickup 4wd	5.2	1999	8	manual(m5)	4	11	16
## 73	dodge	ram 1500	pickup 4wd	5.7	2008	8	auto(15)	4	13	17
## 74	dodge	ram 1500	pickup 4wd	5.9	1999	8	auto(14)	4	11	15
## 75	ford	expedition	2wd	4.6	1999	8	auto(14)	r	11	17
## 76	ford	expedition	2wd	5.4	1999	8	auto(14)	r	11	17
## 77	ford	expedition	2wd	5.4	2008	8	auto(16)	r	12	18
## 78	ford	explorer	4wd	4.0	1999	6	auto(15)	4	14	17
## 79	ford	explorer	4wd	4.0	1999	6	manual(m5)	4	15	19
## 80	ford	explorer	4wd	4.0	1999	6	auto(15)	4	14	17
## 81	ford	explorer	4wd	4.0	2008	6	auto(15)	4	13	19
## 82	ford	explorer	4wd	4.6	2008	8	auto(16)	4	13	19
## 83	ford	explorer	4wd	5.0	1999	8	auto(14)	4	13	17
## 84	ford	f150 pickup	4wd	4.2	1999	6	auto(14)	4	14	17
## 85	ford	f150 pickup	4wd	4.2	1999	6	manual(m5)	4	14	17
## 86	ford	f150 pickup	4wd	4.6	1999	8	manual(m5)	4	13	16
## 87	ford	f150 pickup	4wd	4.6	1999	8	auto(14)	4	13	16
## 88	ford	f150 pickup	4wd	4.6	2008	8	auto(14)	4	13	17
## 89	ford	f150 pickup	4wd	5.4	1999	8	auto(14)	4	11	15
## 90	ford	f150 pickup	4wd	5.4	2008	8	auto(14)	4	13	17
## 91	ford	mustang		3.8	1999	6	manual(m5)	r	18	26
## 92	ford	mustang		3.8	1999	6	auto(14)	r	18	25
## 93	ford	mustang		4.0	2008	6	manual(m5)	r	17	26
## 94	ford	mustang		4.0	2008	6	auto(15)	r	16	24
## 95	ford	mustang		4.6	1999	8	auto(14)	r	15	21
## 96	ford	mustang		4.6	1999	8	manual(m5)	r	15	22
## 97	ford	mustang		4.6	2008	8	manual(m5)	r	15	23
## 98	ford	mustang		4.6	2008	8	auto(15)	r	15	22
## 99	ford	mustang		5.4	2008	8	manual(m6)	r	14	20
## 109	hyundai	sonata		2.4	1999	4	auto(14)	f	18	26
## 110	hyundai	sonata		2.4	1999	4	manual(m5)	f	18	27
## 113	hyundai	sonata		2.5	1999	6	auto(14)	f	18	26
## 114	hyundai	sonata		2.5	1999	6	manual(m5)	f	18	26
## 115	hyundai	sonata		3.3	2008	6	auto(15)	f	19	28
## 116	hyundai	tiburon		2.0	1999	4	auto(14)	f	19	26
## 117	hyundai	tiburon		2.0	1999	4	manual(m5)	f	19	29
## 120	hyundai	tiburon		2.7	2008	6	auto(14)	f	17	24
## 121	hyundai	tiburon		2.7	2008	6	manual(m6)	f	16	24
## 122	hyundai	tiburon		2.7	2008	6	manual(m5)	f	17	24
## 123	jeep	grand cherokee	4wd	3.0	2008	6	auto(15)	4	17	22
## 124	jeep	grand cherokee	4wd	3.7	2008	6	auto(15)	4	15	19
## 125	jeep	grand cherokee	4wd	4.0	1999	6	auto(14)	4	15	20

## 126	jeep	grand cherokee 4wd	4.7	1999	8	auto(14)	4	14	17
## 127	jeep	grand cherokee 4wd	4.7	2008	8	auto(15)	4	9	12
## 128	jeep	grand cherokee 4wd	4.7	2008	8	auto(15)	4	14	19
## 129	jeep	grand cherokee 4wd	5.7	2008	8	auto(15)	4	13	18
## 130	jeep	grand cherokee 4wd	6.1	2008	8	auto(15)	4	11	14
## 131	land rover	range rover	4.0	1999	8	auto(14)	4	11	15
## 132	land rover	range rover	4.2	2008	8	auto(s6)	4	12	18
## 133	land rover	range rover	4.4	2008	8	auto(s6)	4	12	18
## 134	land rover	range rover	4.6	1999	8	auto(14)	4	11	15
## 135	lincoln	navigator 2wd	5.4	1999	8	auto(14)	r	11	17
## 136	lincoln	navigator 2wd	5.4	1999	8	auto(14)	r	11	16
## 137	lincoln	navigator 2wd	5.4	2008	8	auto(16)	r	12	18
## 138	mercury	mountaineer 4wd	4.0	1999	6	auto(15)	4	14	17
## 139	mercury	mountaineer 4wd	4.0	2008	6	auto(15)	4	13	19
## 140	mercury	mountaineer 4wd	4.6	2008	8	auto(16)	4	13	19
## 141	mercury	mountaineer 4wd	5.0	1999	8	auto(14)	4	13	17
## 143	nissan	altima	2.4	1999	4	auto(14)	f	19	27
## 146	nissan	altima	3.5	2008	6	manual(m6)	f	19	27
## 147	nissan	altima	3.5	2008	6	auto(av)	f	19	26
## 148	nissan	maxima	3.0	1999	6	auto(14)	f	18	26
## 149	nissan	maxima	3.0	1999	6	manual(m5)	f	19	25
## 150	nissan	maxima	3.5	2008	6	auto(av)	f	19	25
## 151	nissan	pathfinder 4wd	3.3	1999	6	auto(14)	4	14	17
## 152	nissan	pathfinder 4wd	3.3	1999	6	manual(m5)	4	15	17
## 153	nissan	pathfinder 4wd	4.0	2008	6	auto(15)	4	14	20
## 154	nissan	pathfinder 4wd	5.6	2008	8	auto(s5)	4	12	18
## 155	pontiac	grand prix	3.1	1999	6	auto(14)	f	18	26
## 156	pontiac	grand prix	3.8	1999	6	auto(14)	f	16	26
## 157	pontiac	grand prix	3.8	1999	6	auto(14)	f	17	27
## 158	pontiac	grand prix	3.8	2008	6	auto(14)	f	18	28
## 159	pontiac	grand prix	5.3	2008	8	auto(s4)	f	16	25
## 160	subaru	forester awd	2.5	1999	4	manual(m5)	4	18	25
## 161	subaru	forester awd	2.5	1999	4	auto(14)	4	18	24
## 163	subaru	forester awd	2.5	2008	4	manual(m5)	4	19	25
## 165	subaru	forester awd	2.5	2008	4	auto(14)	4	18	23
## 167	subaru	impreza awd	2.2	1999	4	manual(m5)	4	19	26
## 168	subaru	impreza awd	2.5	1999	4	manual(m5)	4	19	26
## 169	subaru	impreza awd	2.5	1999	4	auto(14)	4	19	26
## 172	subaru	impreza awd	2.5	2008	4	manual(m5)	4	19	25
## 174	toyota	4runner 4wd	2.7	1999	4	manual(m5)	4	15	20
## 175	toyota	4runner 4wd	2.7	1999	4	auto(14)	4	16	20
## 176	toyota	4runner 4wd	3.4	1999	6	auto(14)	4	15	19
## 177	toyota	4runner 4wd	3.4	1999	6	manual(m5)	4	15	17
## 178	toyota	4runner 4wd	4.0	2008	6	auto(15)	4	16	20
## 179	toyota	4runner 4wd	4.7	2008	8	auto(15)	4	14	17
## 184	toyota	camry	3.0	1999	6	auto(14)	f	18	26
## 185	toyota	camry	3.0	1999	6	manual(m5)	f	18	26
## 186	toyota	camry	3.5	2008	6	auto(s6)	f	19	28
## 191	toyota	camry solara	3.0	1999	6	auto(14)	f	18	26
## 192	toyota	camry solara	3.0	1999	6	manual(m5)	f	18	26
## 193	toyota	camry solara	3.3	2008	6	auto(s5)	f	18	27
## 199	toyota	land cruiser wagon 4wd	4.7	1999	8	auto(14)	4	11	15
## 200	toyota	land cruiser wagon 4wd	5.7	2008	8	auto(s6)	4	13	18
## 201	toyota	toyota tacoma 4wd	2.7	1999	4	manual(m5)	4	15	20

## 202	toyota	toyota tacoma 4wd	2.7 1999	4	auto(l4)	4	16	20
## 203	toyota	toyota tacoma 4wd	2.7 2008	4	manual(m5)	4	17	22
## 204	toyota	toyota tacoma 4wd	3.4 1999	6	manual(m5)	4	15	17
## 205	toyota	toyota tacoma 4wd	3.4 1999	6	auto(l4)	4	15	19
## 206	toyota	toyota tacoma 4wd	4.0 2008	6	manual(m6)	4	15	18
## 207	toyota	toyota tacoma 4wd	4.0 2008	6	auto(l5)	4	16	20
## 209	volkswagen	gti	2.0 1999	4	auto(l4)	f	19	26
## 212	volkswagen	gti	2.8 1999	6	manual(m5)	f	17	24
## 215	volkswagen	jetta	2.0 1999	4	auto(l4)	f	19	26
## 220	volkswagen	jetta	2.8 1999	6	auto(l4)	f	16	23
## 221	volkswagen	jetta	2.8 1999	6	manual(m5)	f	17	24
## 225	volkswagen	new beetle	2.0 1999	4	auto(l4)	f	19	26
## 229	volkswagen	passat	1.8 1999	4	auto(l5)	f	18	29
## 230	volkswagen	passat	2.0 2008	4	auto(s6)	f	19	28
## 232	volkswagen	passat	2.8 1999	6	auto(l5)	f	16	26
## 233	volkswagen	passat	2.8 1999	6	manual(m5)	f	18	26
## 234	volkswagen	passat	3.6 2008	6	auto(s6)	f	17	26
##	fl	class	pass					
## 1	p	compact	FALSE					
## 5	p	compact	FALSE					
## 6	p	compact	FALSE					
## 7	p	compact	FALSE					
## 8	p	compact	FALSE					
## 9	p	compact	FALSE					
## 11	p	compact	FALSE					
## 12	p	compact	FALSE					
## 13	p	compact	FALSE					
## 14	p	compact	FALSE					
## 15	p	compact	FALSE					
## 16	p	midsize	FALSE					
## 17	p	midsize	FALSE					
## 18	p	midsize	FALSE					
## 19	r	suv	FALSE					
## 20	e	suv	FALSE					
## 21	r	suv	FALSE					
## 22	r	suv	FALSE					
## 23	r	suv	FALSE					
## 24	p	2seater	FALSE					
## 25	p	2seater	FALSE					
## 26	p	2seater	FALSE					
## 27	p	2seater	FALSE					
## 28	p	2seater	FALSE					
## 29	r	suv	FALSE					
## 30	e	suv	FALSE					
## 31	r	suv	FALSE					
## 32	d	suv	FALSE					
## 33	r	midsize	FALSE					
## 35	r	midsize	FALSE					
## 36	r	midsize	FALSE					
## 37	r	midsize	FALSE					
## 38	r	minivan	FALSE					
## 39	r	minivan	FALSE					
## 40	r	minivan	FALSE					
## 41	r	minivan	FALSE					

```

## 42  r   minivan FALSE
## 43  r   minivan FALSE
## 44  e   minivan FALSE
## 45  r   minivan FALSE
## 46  r   minivan FALSE
## 47  r   minivan FALSE
## 48  r   minivan FALSE
## 49  r   pickup  FALSE
## 50  r   pickup  FALSE
## 51  r   pickup  FALSE
## 52  r   pickup  FALSE
## 53  r   pickup  FALSE
## 54  r   pickup  FALSE
## 55  e   pickup  FALSE
## 56  r   pickup  FALSE
## 57  r   pickup  FALSE
## 58  r       suv  FALSE
## 59  r       suv  FALSE
## 60  e       suv  FALSE
## 61  r       suv  FALSE
## 62  r       suv  FALSE
## 63  r       suv  FALSE
## 64  r       suv  FALSE
## 65  r   pickup  FALSE
## 66  e   pickup  FALSE
## 67  r   pickup  FALSE
## 68  r   pickup  FALSE
## 69  r   pickup  FALSE
## 70  e   pickup  FALSE
## 71  r   pickup  FALSE
## 72  r   pickup  FALSE
## 73  r   pickup  FALSE
## 74  r   pickup  FALSE
## 75  r       suv  FALSE
## 76  r       suv  FALSE
## 77  r       suv  FALSE
## 78  r       suv  FALSE
## 79  r       suv  FALSE
## 80  r       suv  FALSE
## 81  r       suv  FALSE
## 82  r       suv  FALSE
## 83  r       suv  FALSE
## 84  r   pickup  FALSE
## 85  r   pickup  FALSE
## 86  r   pickup  FALSE
## 87  r   pickup  FALSE
## 88  r   pickup  FALSE
## 89  r   pickup  FALSE
## 90  r   pickup  FALSE
## 91  r subcompact FALSE
## 92  r subcompact FALSE
## 93  r subcompact FALSE
## 94  r subcompact FALSE
## 95  r subcompact FALSE

```

```

## 96  r subcompact FALSE
## 97  r subcompact FALSE
## 98  r subcompact FALSE
## 99  p subcompact FALSE
## 109 r   midsize  FALSE
## 110 r   midsize  FALSE
## 113 r   midsize  FALSE
## 114 r   midsize  FALSE
## 115 r   midsize  FALSE
## 116 r subcompact FALSE
## 117 r subcompact FALSE
## 120 r subcompact FALSE
## 121 r subcompact FALSE
## 122 r subcompact FALSE
## 123 d       suv  FALSE
## 124 r       suv  FALSE
## 125 r       suv  FALSE
## 126 r       suv  FALSE
## 127 e       suv  FALSE
## 128 r       suv  FALSE
## 129 r       suv  FALSE
## 130 p       suv  FALSE
## 131 p       suv  FALSE
## 132 r       suv  FALSE
## 133 r       suv  FALSE
## 134 p       suv  FALSE
## 135 r       suv  FALSE
## 136 p       suv  FALSE
## 137 r       suv  FALSE
## 138 r       suv  FALSE
## 139 r       suv  FALSE
## 140 r       suv  FALSE
## 141 r       suv  FALSE
## 143 r   compact FALSE
## 146 p   midsize FALSE
## 147 p   midsize FALSE
## 148 r   midsize FALSE
## 149 r   midsize FALSE
## 150 p   midsize FALSE
## 151 r       suv  FALSE
## 152 r       suv  FALSE
## 153 p       suv  FALSE
## 154 p       suv  FALSE
## 155 r   midsize FALSE
## 156 p   midsize FALSE
## 157 r   midsize FALSE
## 158 r   midsize FALSE
## 159 p   midsize FALSE
## 160 r       suv  FALSE
## 161 r       suv  FALSE
## 163 p       suv  FALSE
## 165 p       suv  FALSE
## 167 r subcompact FALSE
## 168 r subcompact FALSE

```

```

## 169 r subcompact FALSE
## 172 p compact FALSE
## 174 r suv FALSE
## 175 r suv FALSE
## 176 r suv FALSE
## 177 r suv FALSE
## 178 r suv FALSE
## 179 r suv FALSE
## 184 r midsize FALSE
## 185 r midsize FALSE
## 186 r midsize FALSE
## 191 r compact FALSE
## 192 r compact FALSE
## 193 r compact FALSE
## 199 r suv FALSE
## 200 r suv FALSE
## 201 r pickup FALSE
## 202 r pickup FALSE
## 203 r pickup FALSE
## 204 r pickup FALSE
## 205 r pickup FALSE
## 206 r pickup FALSE
## 207 r pickup FALSE
## 209 r compact FALSE
## 212 r compact FALSE
## 215 r compact FALSE
## 220 r compact FALSE
## 221 r compact FALSE
## 225 r subcompact FALSE
## 229 p midsize FALSE
## 230 p midsize FALSE
## 232 p midsize FALSE
## 233 p midsize FALSE
## 234 p midsize FALSE

```

```

#library(dplyr)
#mpg_pass = as_tibble(mpg_pass)
#mpg_pass[mpg_pass$pass == FALSE,]

```