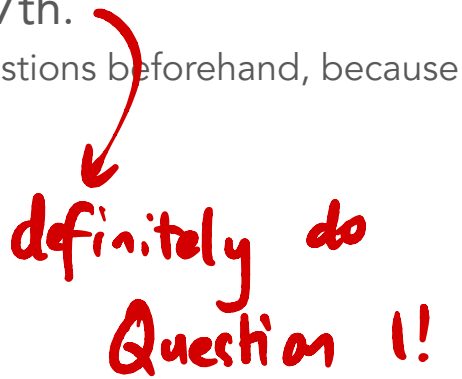
Lecture 13: Midterm Review

EECS 398-003: Practical Data Science, Fall 2024

practicaldsc.org • github.com/practicaldsc/fa24

Announcements

- The Midterm Exam is this Wednesday from 7-9PM. See this post on Ed for lots of details, including where to take it, what is covered, what to bring, and how to study.
- Homework 4 and 5 scores are available on Gradescope.
- There is no lecture on Thursday and no discussion on Friday.
- Homework 6 is due on Thursday, October 17th.
  - Work through the SQL and regular expressions questions beforehand, because the concepts are all in scope for the exam!
  - TF-IDF is in scope too, but we'll review that today.



Agenda

- We'll work through the review worksheet posted here: study.practicaldsc.org/mt-review-tuesday
- I'll post these annotated slides after lecture, and enable solutions on the study site for this worksheet after, too.
- The solutions + recording for Monday's review session are also posted.

ask questions: practicaldsc.org/q !

TF-IDF

and cosine similarity
and
bag of words

Problem 1

Nishant decides to look at reviews for the Catamaran Resort Hotel and Spa. TripAdvisor has 96 reviews for the hotel; of those 96, Nishant's favorite review was:

```
"close to the beach but far from the beach beach"
```

Problem 1.1

What is the TF of `"beach"` in Nishant's favorite review? Give your answer as a simplified fraction.

$$\text{term frequency} = \frac{\#\text{ of words in doc} = \text{"beach"}}{\#\text{ of words in doc}}$$

$$= \boxed{\frac{3}{10}}$$

Problem 1

tip: on exams, we will specify the log base

Nishant decides to look at reviews for the Catamaran Resort Hotel and Spa. TripAdvisor has 96 reviews for the hotel; of those 96, Nishant's favorite review was:

```
"close to the beach but far from the beach beach"
```

Problem 1.2

The TF-IDF of `"beach"` in Nishant's favorite review is $\frac{9}{10}$, when using a base-2 logarithm to compute the IDF. How many of the reviews on TripAdvisor for this hotel contain the term `"beach"`?

- 3
- 6
- 8
- 12
- 16
- 24
- 32

$$\text{TF-IDF} = \underbrace{\frac{\#\text{ of words in doc} = \text{"beach"}}{\#\text{ of words in doc}}}_{\text{TF} = \frac{3}{10} \text{ (last slide)}} \cdot \underbrace{\log_2\left(\frac{96}{C}\right)}_{\text{IDF}} = \frac{9}{10}$$

$$\frac{3}{10} \cdot \log_2\left(\frac{96}{C}\right) = \frac{9}{10} \Rightarrow \log_2\left(\frac{96}{C}\right) = 3 \Rightarrow 2^3 = \frac{96}{C}$$

$$\Rightarrow \boxed{C = 12}$$

Problem 2.1

What is the TF-IDF of the word `"hate"` in Song 0's title? Use base 2 in your logarithm, and give your answer as a simplified fraction.

$$\text{TF-IDF} = \text{TF} \times \text{IDF} = \frac{\#\text{ of words} = \text{"hate"} \text{ in } 0}{\#\text{ words in } 0} \cdot \log\left(\frac{\#\text{ songs}}{\#\text{ songs with "hate"}}\right)$$

$$= \frac{2}{12} \cdot \log_2\left(\frac{4}{2}\right)$$

$$= \frac{1}{6} \cdot 1$$

$$= \boxed{\frac{1}{6}}$$

	track_name
0	i hate you i love you i hate that i love you
1	love me like a love song
2	love you better
3	hate sosa

Problem 2.2

Which word in Song 0's title has the highest TF-IDF?

- "i"
- "hate"
- "you"
- "love"
- "that"
- Two or more words are tied for the highest TF-IDF in Song 0's title

$$\text{TF-IDF}(t,d) = \left(\text{prop of terms in } d == t\right) \cdot \text{how rare is } t?$$

$$\text{TF}(\text{"i"}, \text{song 0}) = \frac{4}{12} = \frac{1}{3} \leftarrow \text{max possible!}$$

$$\text{IDF}(\text{"i"}) = \log_2\left(\frac{4}{1}\right) \leftarrow \text{max possible!}$$

	track_name
0	i hate you i love you i hate that i love you
1	love me like a love song
2	love you better
3	hate sosa

word = term

Problem 2.3

Let $\text{tfidf}(t, d)$ be the TF-IDF of term t in document d, and let $\text{bow}(t, d)$ be the number of occurrences of term t in document d.

count

Select all correct answers below.

- ☐ If $\text{tfidf}(t, d) = 0$, then $\text{bow}(t, d) = 0$.
- ☑ If $\text{bow}(t, d) = 0$, then $\text{tfidf}(t, d) = 0$.
- ☐ Neither of the above statements are necessarily true.

2 ways TF-IDF = 0:

1) TF = 0
→ t never appears in d

2) IDF = 0:
→ t is in every doc!

$$\text{TF-IDF}(t,d) = \text{TF}(t,d) \cdot \text{IDF}(t)$$

$$\text{TF-IDF}(t,d) = \frac{\text{bow}(t,d)}{\#\text{ words in } d} \cdot \text{IDF}(t)$$

	track_name
0	i hate you i love you i hate that i love you
1	love me like a love song
2	love you better
3	hate sosa

Problem 2.4

$\|\vec{u}\| = \sqrt{u_1^2 + u_2^2 + \cdots + u_n^2}$

Below, we've encoded the corpus from the previous page using the bag-of-words model.

$\cos\text{sim}(\vec{u}, \vec{v}) = \frac{\vec{u} \cdot \vec{v}}{\|\vec{u}\| \|\vec{v}\|}$

$= \vec{u} \cdot \vec{v}$

only because all rows here have $\|\vec{u}\| = 1$!

	better	hate	like	love	me	song	sosa	that	you
0	0	0.47	0	0.47	0	0	0	0.24	0.71
1	0	0	0.38	0.76	0.38	0.38	0	0	0
2	0.58	0	0	0.58	0	0	0	0	0.58
3	0	0.71	0	0	0	0	0.71	0	0

Note that in the above DataFrame, each row has been normalized to have a length of 1 (i.e. $|\vec{v}| = 1$ for all four row vectors).

Which song's title has the highest cosine similarity with Song 0's title?

- Song 1
- Song 2
- Song 3

let's take 3 dot products!

0→1 : $0.47 \cdot 0.76$

0→2 : $0.47 \cdot 0.58 + 0.71 \cdot 0.58$

0→3 : $0.47 \cdot 0.71$

which is largest?

	track_name
0	i hate you i love you i hate that i love you
1	love me like a love song
2	love you better
3	hate sosa

let's take 3 dot products!

$0 \to 1$: $0.47 \cdot 0.76$ < $0.5 \cdot 0.8 = 0.4$

$0 \to 2$: $0.47 \cdot 0.58 + 0.71 \cdot 0.58$ > $0.4 \cdot 0.5 + 0.7 \cdot 0.5$

$0.2 + 0.35 = 0.55$

$0 \to 3$: ~~$0.47 \cdot 0.71$~~ ← rule out, because $0.71 < 0.76$

which is largest?

Merging

Problem 3

The DataFrame `dogs`, contains one row for every registered pet dog in Zurich, Switzerland in 2017.

	owner_id	owner_age	owner_sex	district	primary_breed
0	4215	41-50	f	8	Bergamasker
1	4215	41-50	f	8	Border Collie
2	6071	61-70	m	3	Cocker Spaniel
3	123237	21-30	f	7	Sheltie
4	135726	11-20	f	11	Pinscher

In this question, assume that there are more than 12 districts in `dogs`.

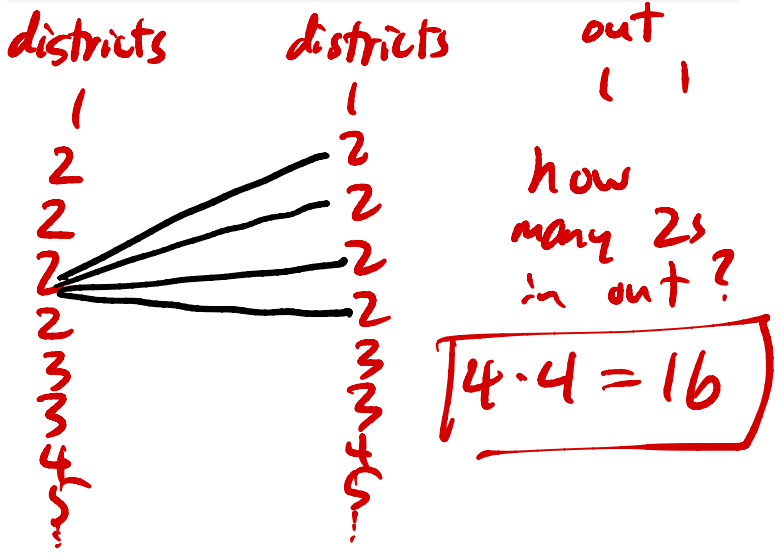
Suppose we merge the `dogs` DataFrame with itself as follows.

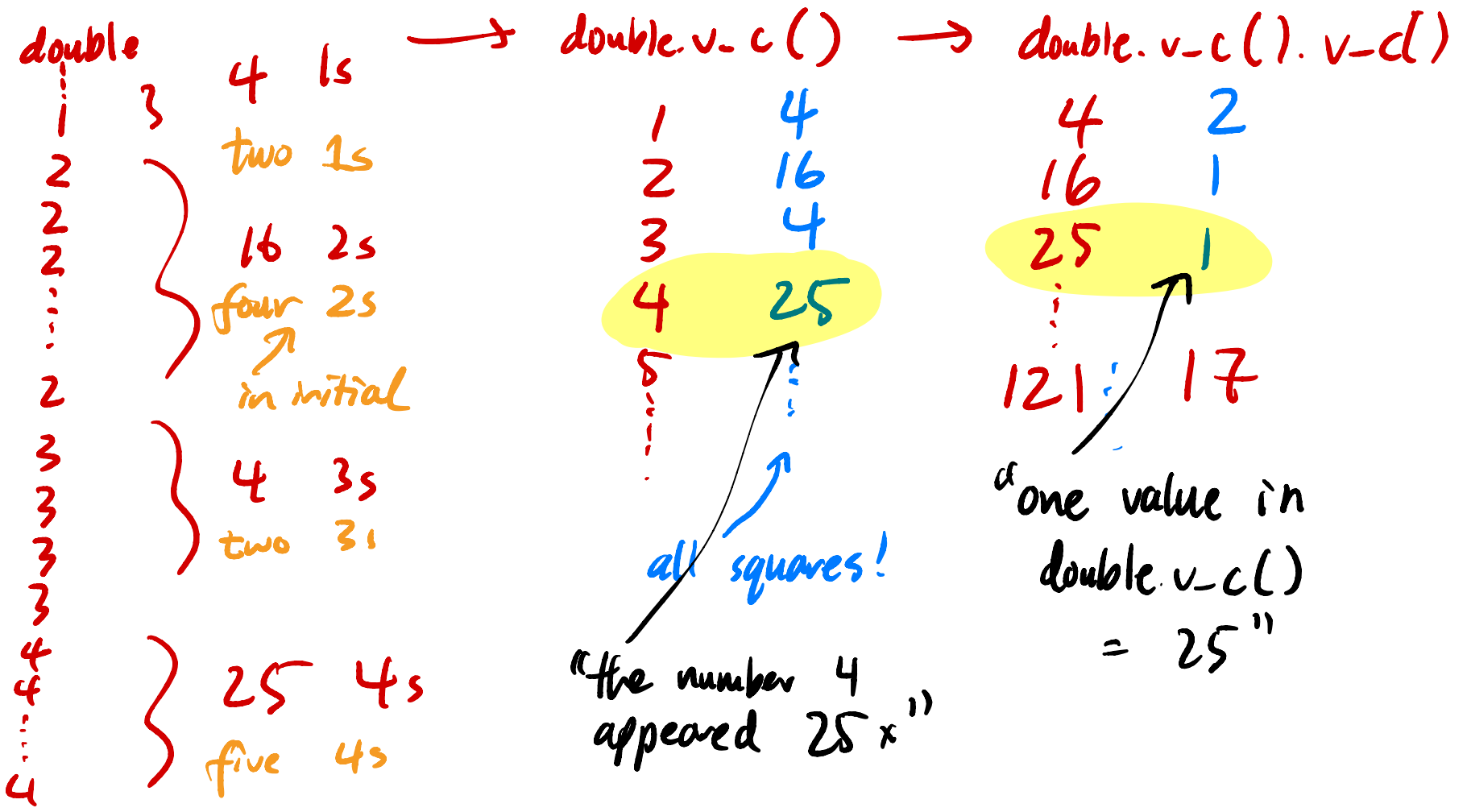
```
# on="x" is the same as specifying both left_on="x" and right_on="x".
double = dogs.merge(dogs, on="district")

# sort_index sorts a Series in increasing order of its index.
square = double["district"].value_counts().value_counts().sort_index()
```

The first few rows of `square` are shown below.

```
1     5500
4      215
9       40
```

all we need for 3.1




double
4 1s
two 1s
16 2s
four 2s
in initial
4 3s
two 3s
25 4s
five 4s
double.v_c()
4
16
25
all squares!
"the number 4 appeared 25x"
double.v_c().v_c()
"one value in double.v_c() = 25"

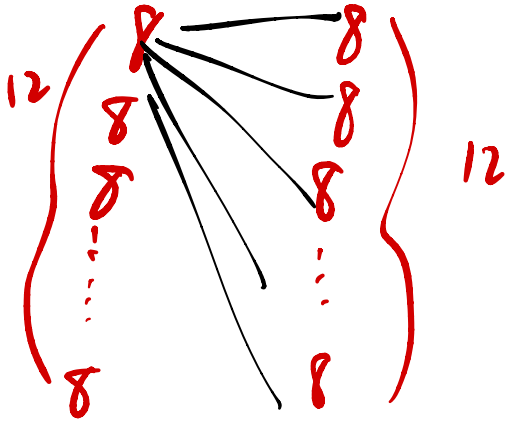
Problem 3.1

In `dogs`, there are 12 rows with a `"district"` of `8`. How many rows of `double` have a `"district"` of `8`? Give your answer as a positive integer.

In this question, assume that there are more than 12 districts in `dogs`.

Suppose we merge the `dogs` DataFrame with itself as follows.

```
# on="x" is the same as specifying both left_on="x" and right_on="x".
double = dogs.merge(dogs, on="district")
```



$12 \cdot 12 = 144$

Problem 3.2

What does the following expression evaluate to? Give your answer as a positive integer.

```
dogs.groupby("district").filter(lambda df: df.shape[0] == 3).shape[0]
```

Hint: Unlike in 5.1, your answer to 5.2 depends on the values in `square`.

In this question, assume that there are more than 12 districts in `dogs`.

Suppose we merge the `dogs` DataFrame with itself as follows.

```
# on="x" is the same as specifying both left_on="x" and right_on="x".
double = dogs.merge(dogs, on="district")

# sort_index sorts a Series in increasing order of its index.
square = double["district"].value_counts().value_counts().sort_index()
```

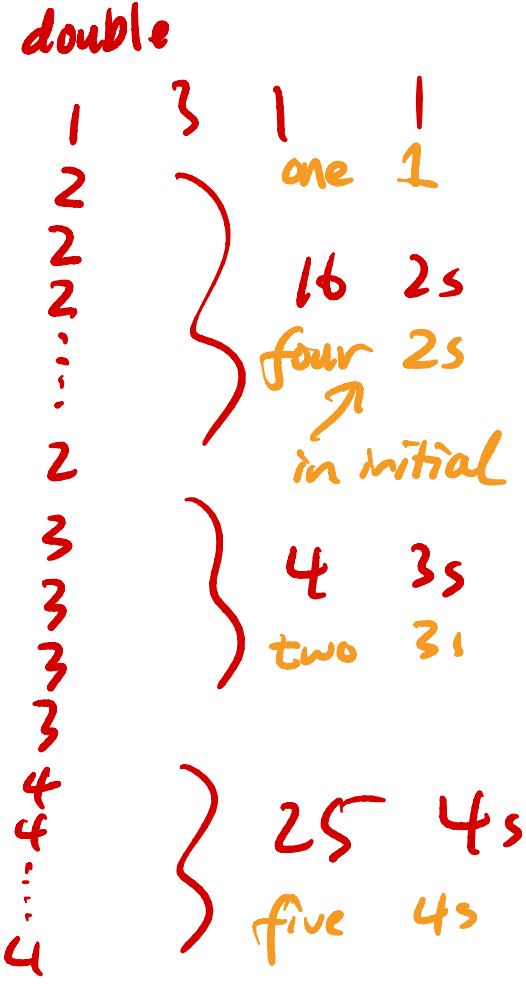
The first few rows of `square` are shown below.

1	5500
4	215
9	40

→ 5500 districts appeared once in dogs

→ 215 districts appeared twice in dogs

→ 40 districts appeared 3x in dogs



```
dogs.groupby("district").filter(lambda df: df.shape[0]==3).shape[0]
```

```
.groupby("col").size()
["col"].value_counts()
```

does this district appear exactly 3x?

→ 40 districts appear exactly 3x

→ total rows = $40 \cdot 3 = \boxed{120}$

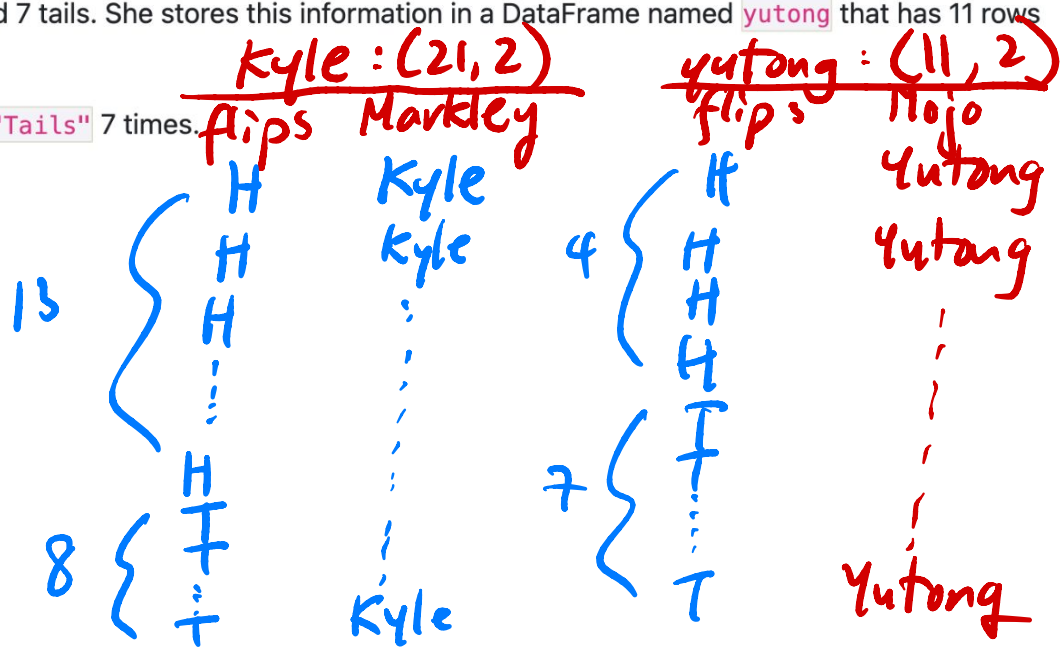
Problem 4

Kyle flips the coin 21 times and sees 13 heads and 8 tails. He stores this information in a DataFrame named `kyle` that has 21 rows and 2 columns, such that:

- The `"flips"` column contains `"Heads"` 13 times and `"Tails"` 8 times.
- The `"Markley"` column contains `"Kyle"` 21 times.

Then, Yutong flips the coin 11 times and sees 4 heads and 7 tails. She stores this information in a DataFrame named `yutong` that has 11 rows and 2 columns, such that:

- The `"flips"` column contains `"Heads"` 4 times and `"Tails"` 7 times.
- The `"MoJo"` column contains `"Yutong"` 11 times.



Problem 4.1

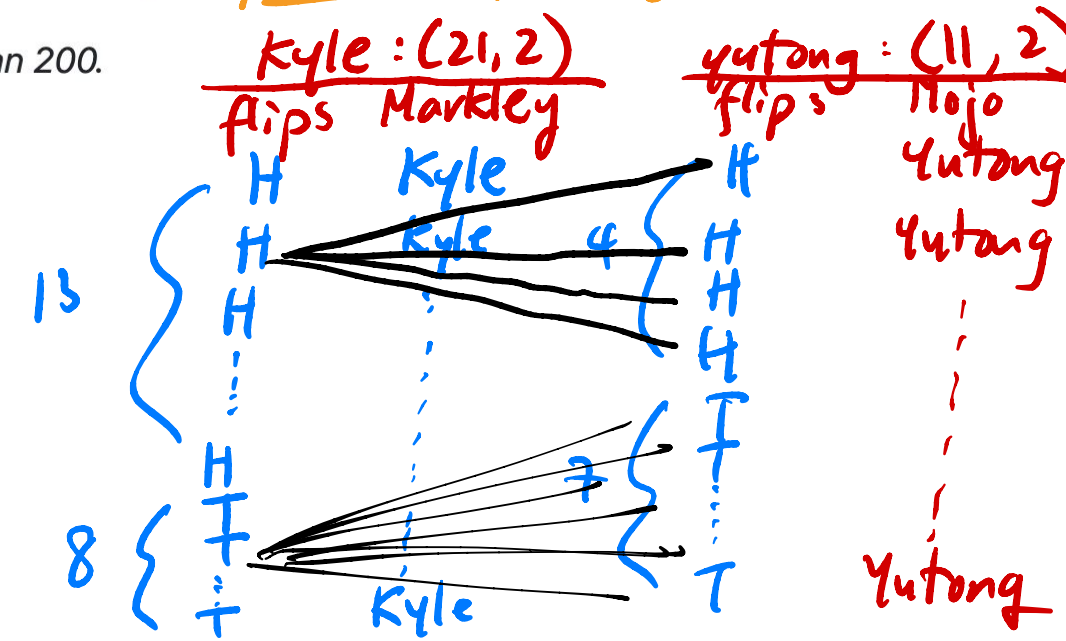
output df: (13)(4) + (7)(8)
= 52 + 56 = 108

How many rows are in the following DataFrame? Give your answer as an integer.

```
kyle.merge(yutong, on="flips")
```

here, left=right=outer=inner merge!!!

Hint: The answer is less than 200.



Problem 4.2

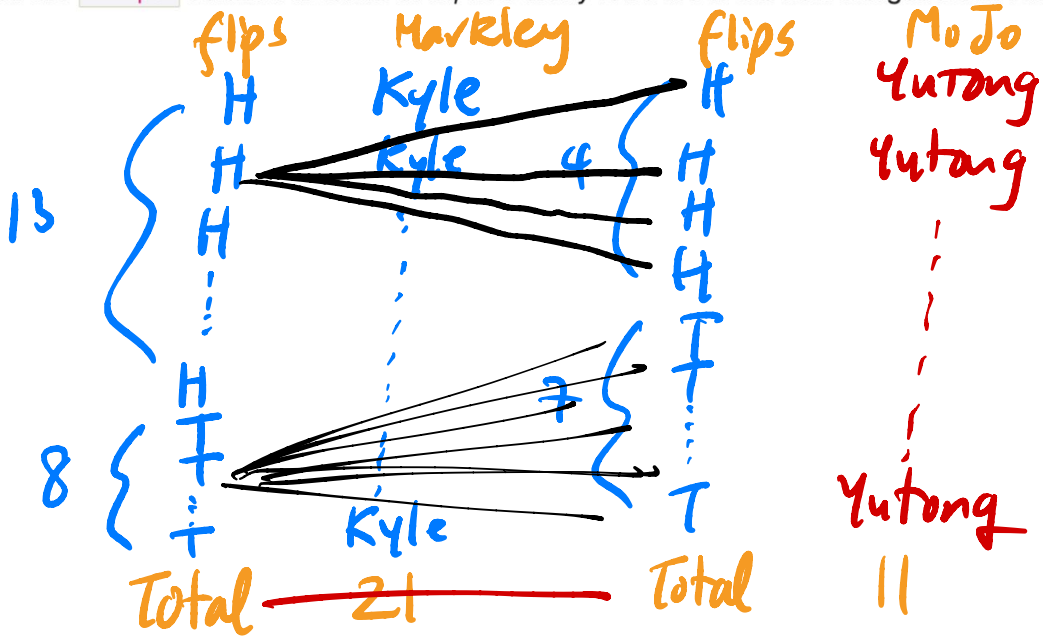
all this does is add 1 × 1 = 1 new row!

Let A be your answer to the previous part. Now, suppose that:

- `kyle` contains an additional row, whose `"flips"` value is `"Total"` and whose `"Markley"` value is 21.
- `yutong` contains an additional row, whose `"flips"` value is `"Total"` and whose `"MoJo"` value is 11.

Suppose we again merge `kyle` and `yutong` on the `"flips"` column. In terms of A, how many rows are in the new merged DataFrame?

- A
- $A + 1$
- $A + 2$
- $A + 4$
- $A + 231$



Problem 5

Suppose the DataFrame `today` consists of 15 rows — 3 rows for each of 5 different `"artist_names"`. For each artist, it contains the `"track_name"` for their three most-streamed songs today. For instance, there may be one row for `"olivia rodrigo"` and `"favorite crime"`, one row for `"olivia rodrigo"` and `"drivers license"`, and one row for `"olivia rodrigo"` and `"deja vu"`.

Another DataFrame, `genres`, is shown below in its entirety.

today → artist_names track_name

genres

	artist_names	genre
0	harry styles	Pop
1	olivia rodrigo	Pop
2	glass animals	Alternative
3	drake	Hip-Hop/Rap
4	doja cat	Hip-Hop/Rap

5 artists
3 rows per artist

Problem 5.1

Suppose we perform an **inner** merge between `today` and `genres` on `"artist_names"`. If the five `"artist_names"` in `today` are the same as the five `"artist_names"` in `genres`, what fraction of the rows in the merged DataFrame will contain `"Pop"` in the `"genre"` column? Give your answer as a simplified fraction.

today artist_names track_names

hs
hs
hs
ov
ov
ov
ga
ga
ga
dc

merged:

Pop
Pop
Pop
pop
pop
alt

15 rows

6 = pop

$\Rightarrow \frac{6}{15} = \frac{2}{5}$

Another DataFrame, `genres`, is shown below in its entirety.

	artist_names	genre
0	harry styles	Pop
1	olivia rodrigo	Pop
2	glass animals	Alternative
3	drake	Hip-Hop/Rap
4	doja cat	Hip-Hop/Rap

Problem 5.2

Suppose we perform an **inner** merge between `today` and `genres` on `"artist_names"`. Furthermore, suppose that the only overlapping `"artist_names"` between `today` and `genres` are `"drake"` and `"olivia rodrigo"`. What fraction of the rows in the merged DataFrame will contain `"Pop"` in the `"genre"` column? Give your answer as a simplified fraction.

output:

olivia pop
olivia pop
olivia pop
drake hip hop
drake hip hop
drake hip hop

→ 6 rows, 3 = "pop"

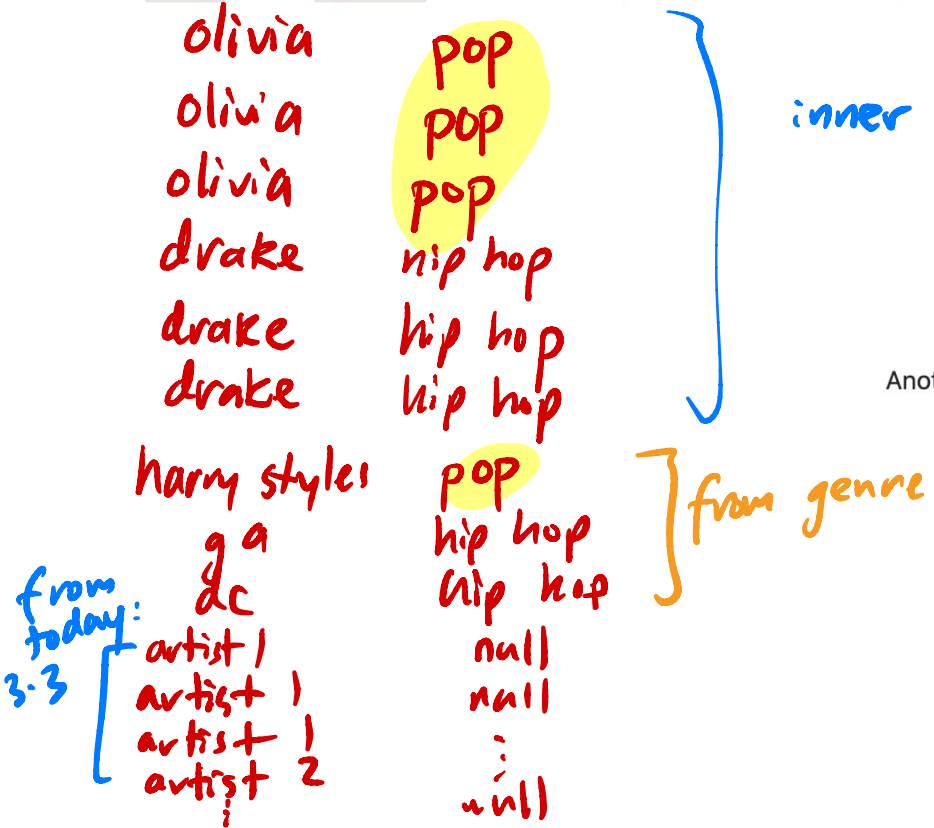
$\Rightarrow \frac{3}{6} = \frac{1}{2}$

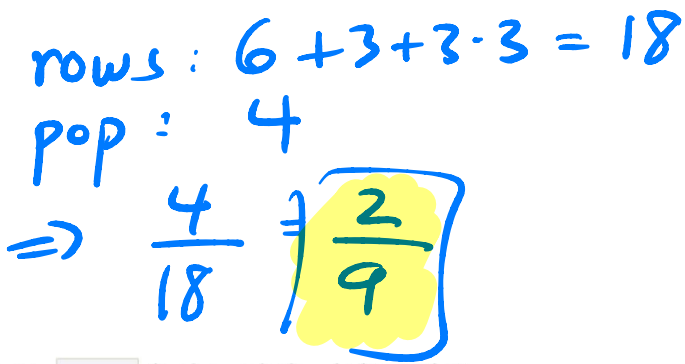
Another DataFrame, `genres`, is shown below in its entirety.

	artist_names	genre
0	harry styles	Pop
1	olivia rodrigo	Pop
2	glass animals	Alternative
3	drake	Hip-Hop/Rap
4	doja cat	Hip-Hop/Rap

Problem 5.3

Suppose we perform an **outer** merge between `today` and `genres` on `"artist_names"`. Furthermore, suppose that the only overlapping `"artist_names"` between `today` and `genres` are `"drake"` and `"olivia rodrigo"`. What fraction of the rows in the merged DataFrame will contain `"Pop"` in the `"genre"` column? Give your answer as a simplified fraction.





Another DataFrame, `genres`, is shown below in its entirety.

	artist_names	genre
0	harry styles	Pop
1	olivia rodrigo	Pop
2	glass animals	Alternative
3	drake	Hip-Hop/Rap
4	doja cat	Hip-Hop/Rap