NAME: Rishi Patel
PRN: 202401120070
ROLL NO: CS8-79
BATCH: CS84

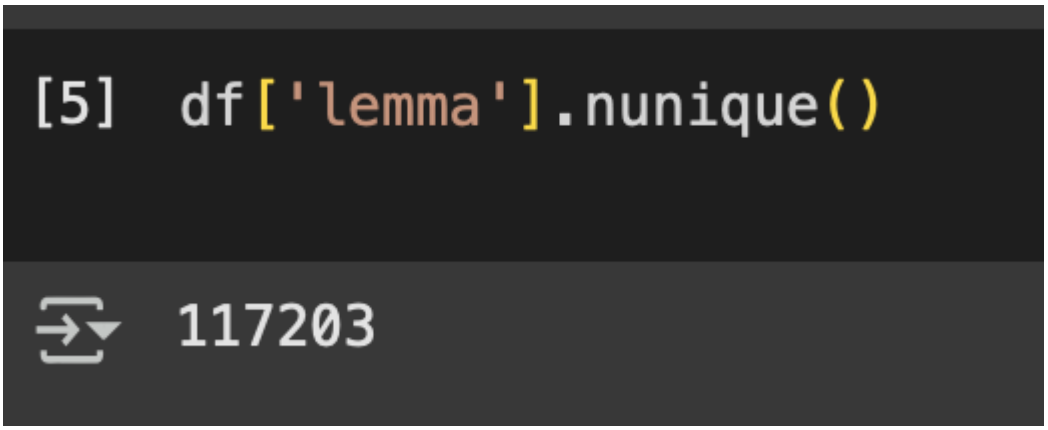
```
Import numpy as np
Import pandas as pd

df= pd.read_csv("/synonyms.csv")
df.head()
```

	lemma	part_of_speech	synonyms
0	.22-caliber	adjective	.22 caliber;.22 calibre;.22-calibre
1	.22-calibre	adjective	.22 caliber;.22-caliber;.22 calibre
2	.22 caliber	adjective	.22-caliber;.22 calibre;.22-calibre
3	.22 calibre	adjective	.22 caliber;.22-caliber;.22-calibre
4	.38-caliber	adjective	.38 caliber;.38 calibre;.38-calibre

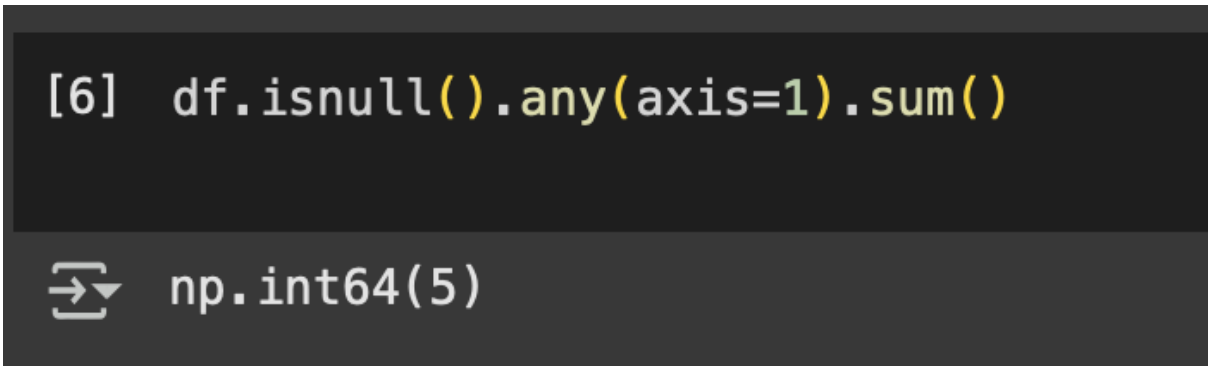
1. Count the total number of unique lemmas

   df['lemma'].nunique()



2. Identify how many rows has missing values in column

   df.isnull().any(axis=1).sum()



3. Find most common part of speech

   df['part_of_speech'].value_counts().idxmax()

4. Count total number of adjectives in dataset

   (df['part_of_speech'] == 'adjective').sum()

```
[8]  (df['part_of_speech'] == 'adjective').sum()

     np.int64(4274)
```

5. Find all entries where lemma has more than 3 synonyms

   df[df['synonyms'].str.count(';') > 2]

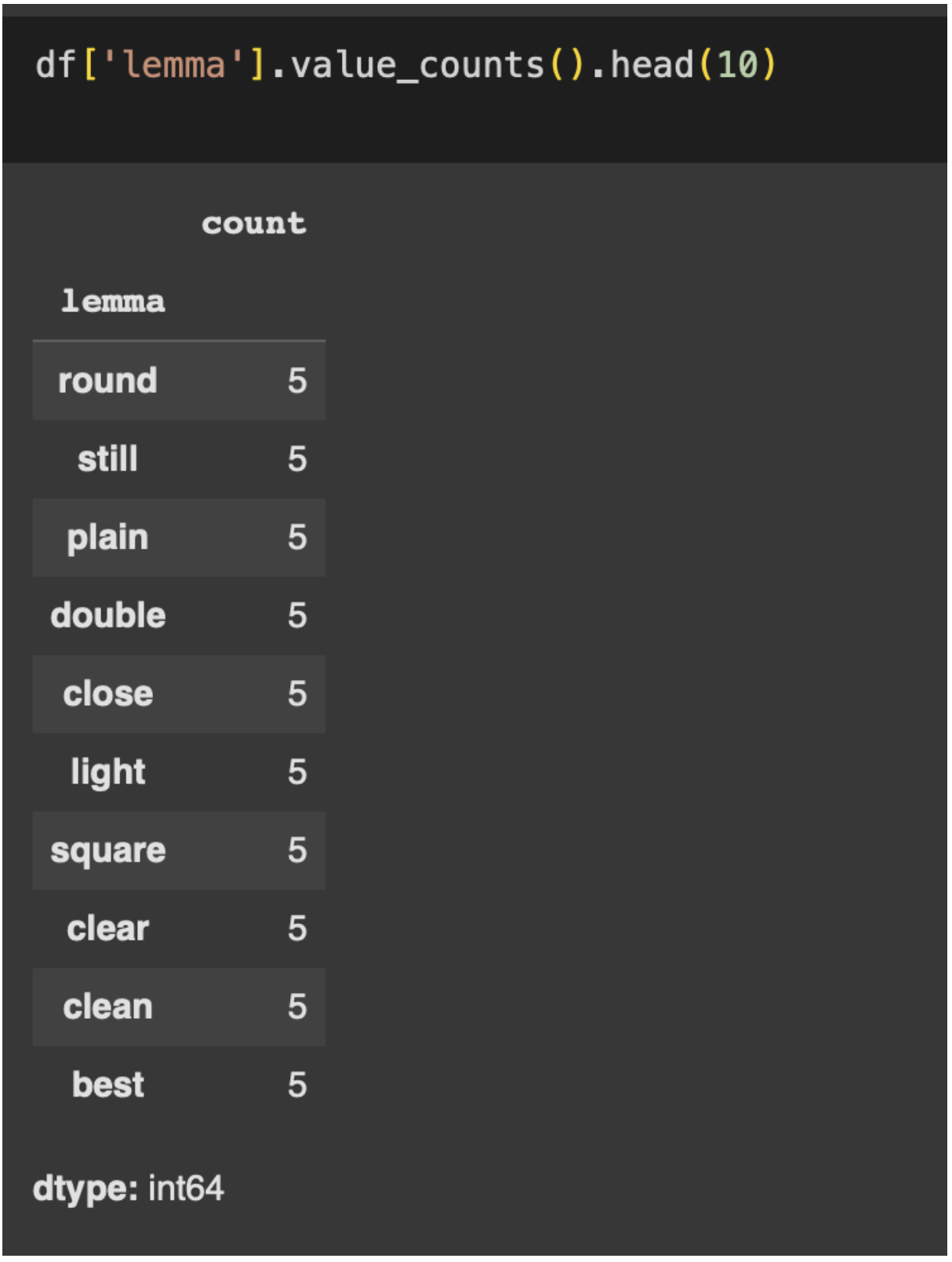
```
df[df['synonyms'].str.count(';') > 2]
```

	lemma	part_of_speech	synonyms
12	0	noun	zero;nought;cipher;cypher
14	1	noun	one;I;ace;single;unity
16	10	noun	ten;X;tenner;decade
19	100	noun	hundred;C;century;one C
21	1000	noun	thousand;one thousand;M;K;chiliad;G;grand;thou...
...	...	...	...
126989	yack away	verb	yack;jaw;rattle on;yap away
126990	yammer	verb	howl;wrawl;yowlIwhine;grizzle;yawp
126991	yap away	verb	yack;jaw;yack away;rattle on
126993	yaup	verb	howl;ululate;wail;roar;yawl
126995	yearn	verb	hanker;longlache;yen;pine;languish

29181 rows × 3 columns

6. Find the top 10 most common lemmas

   df['lemma'].value_counts().head(10)

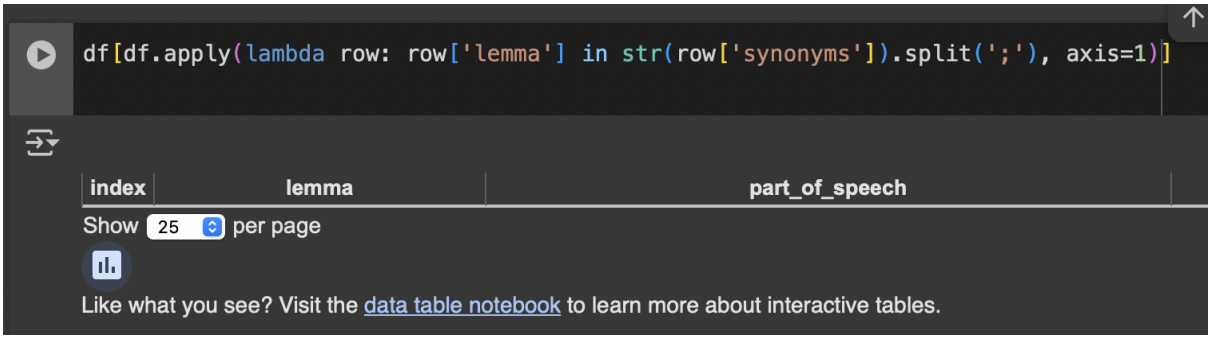


7. Count how many unique synonyms exist in the dataset

   df['part_of_speech'].unique()

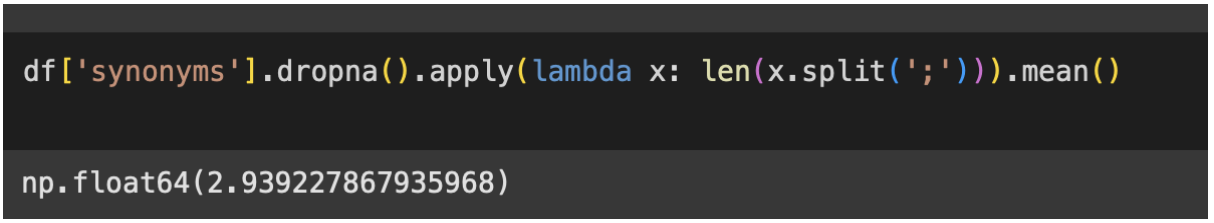
8. Find rows where lemma and any synonym are exactly the same

   df[df.apply(lambda row: row['lemma'] in str(row['synonyms']).split(';'), axis=1)]



9. Get average number of synonyms per entry

   df['synonyms'].dropna().apply(lambda x: len(x.split(';'))).mean()



10. Create a new column with the number of synonyms per lemma.

    df['synonym_count'] = df['synonyms'].dropna().apply(lambda x: len(x.split(';')))

11. Identify the lemma with the highest number of synonyms.

    df.loc[df['synonym_count'].idxmax()]

```
df.loc[df['synonym_count'].idxmax()]
```

	13930
lemma	passing
part_of_speech	verb
synonyms	pass;go through;go acrossltravel by;pass by;su...
synonym_count	85.0

dtype: object

12. Filter entries with missing lemmas.

    df[df['lemma'].isnull()]

    df[df['lemma'].isnull()]

```
df[df['lemma'].isnull()]
```

	lemma	part_of_speech	synonyms	synonym_count
13102	NaN	noun	nothing;nil;nix;nada;aught;cipher;cypher;goose...	13.0
13103	NaN	satellite	void	1.0
83831	NaN	noun	grandma;grandmother;granny;grannie;gran;nannal...	7.0

13. Determine how many adjectives have more than 5 synonyms.

    df[(df['part_of_speech'] == 'adjective') & (df['synonyms'].str.count(';') > 4)].shape[0]

```
[17] df[(df['part_of_speech'] == 'adjective') & (df['synonyms'].str.count(';') > 4)].shape[0]

     12
```

14. Replace missing lemmas with a placeholder string like "UNKNOWN".

    df['lemma'].fillna('UNKNOWN', inplace=True)

15. Count how many synonym entries contain a hyphen.

    df['synonyms'].dropna().str.contains('-').sum()

```
df['synonyms'].dropna().str.contains('-').sum()

np.int64(9821)
```

16. Find lemmas that appear as synonyms of other lemmas.

    lemmas_set = set(df['lemma'].dropna())
    synonyms_set = set(all_synonyms.dropna())
    lemmas_set & synonyms_set

```
lemmas_set = set(df['lemma'].dropna())
synonyms_set = set(df['synonyms'].dropna().str.cat(sep=';').split(';')) # Extract all synonyms from 'synonyms' colu
common_elements = lemmas_set & synonyms_set # Find common elements
print(common_elements)

{'shellac varnish', 'rosilla', 'pretence', 'dusk', 'burdensomeness', 'angoumois moth', 'psychic phenomena', 'unctio
```

17. Count the number of duplicate rows.

```
df.duplicated().sum()
```

```
np.int64(0)
```

18. Find the top 5 most common synonyms.

```
all_synonyms = df['synonyms'].dropna().str.cat(sep=';').split(';')
all_synonyms = pd.Series(all_synonyms) # Convert to Series for value_counts()
all_synonyms.value_counts().head(5)
```

```
all_synonyms = df['synonyms'].dropna().str.cat(sep=';').split(';')
all_synonyms = pd.Series(all_synonyms)
all_synonyms.value_counts().head(5)
```

	count
pass	138
take	136
go	127
get	127
break	122

dtype: int64

19. Group and count lemmas by part_of_speech.

    df.groupby('part_of_speech')['lemma'].count()

```
[33] df.groupby('part_of_speech')['lemma'].count()
```

part_of_speech	lemma
adjective	4274
adverb	2694
noun	94600
satellite	11752
verb	13681

dtype: int64

20. Find all lemmas whose synonyms include a number (e.g., "22", "100", etc.).

```
import re


mask = df['synonyms'].dropna().str.contains(r'\b\d+\b',
regex=True).reset_index(drop=True)


mask = mask.reindex(df.index, fill_value=False)


filtered_df = df[mask]

filtered_df
```

	lemma	part_of_speech	synonyms	synonym_count
0	.22-caliber	adjective	.22 caliber;.22 calibre;.22-calibre	3.0
1	.22-calibre	adjective	.22 caliber;.22-caliber;.22 calibre	3.0
2	.22 caliber	adjective	.22-caliber;.22 calibre;.22-calibre	3.0
3	.22 calibre	adjective	.22 caliber;.22-caliber;.22-calibre	3.0
4	.38-caliber	adjective	.38 caliber;.38 calibre;.38-calibre	3.0
...	...	...	...	...
120891	zimmer frame	noun	walker;Zimmer;Zimmer frame	3.0
120927	zircon	noun	zirconium silicate	1.0
120940	ziziphus jujuba	noun	jujube;jujube bush;Christ's-thorn;Jerusalem th...	5.0
120995	zoysia matrella	noun	Manila grass;Japanese carpet grass;Zoysia matr...	3.0
123359	flatten out	verb	flatten	1.0

1057 rows × 4 columns