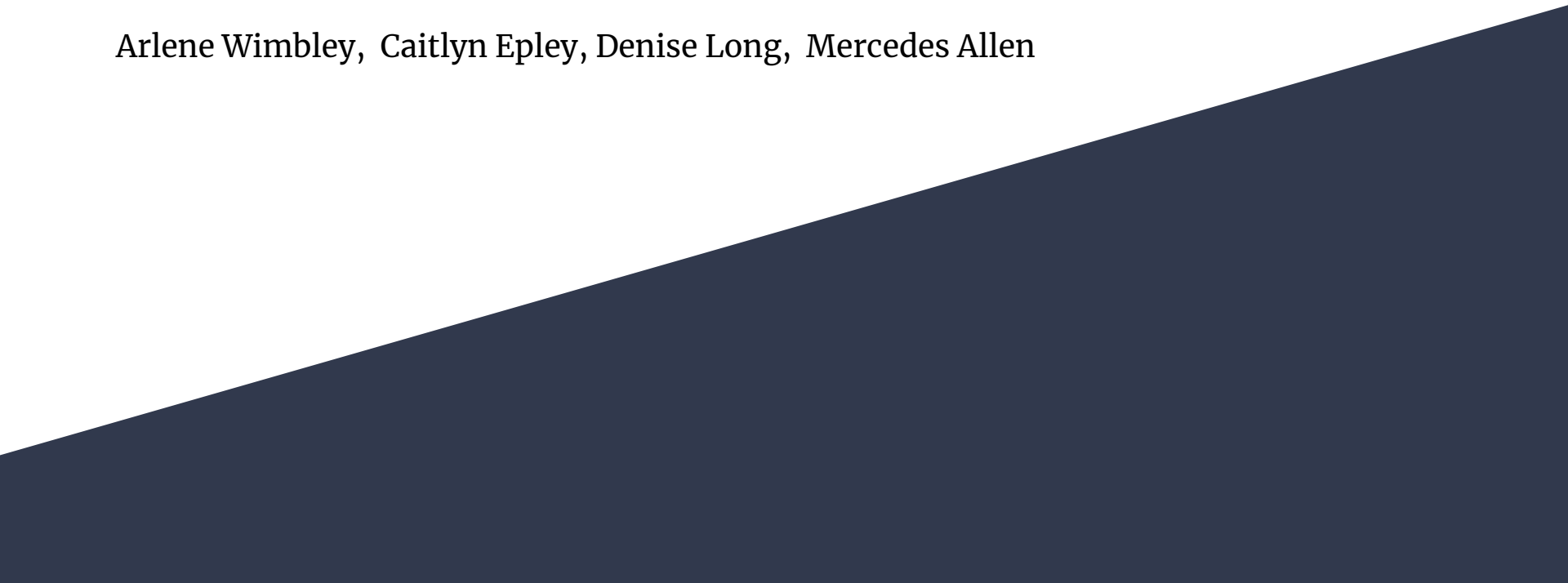


Project 3 - Data Engineering Track

Arlene Wimbley, Caitlyn Epley, Denise Long, Mercedes Allen

A dark blue diagonal gradient bar that starts from the bottom left corner and extends towards the top right corner, covering the lower half of the slide.

What is the Library of Congress?



The Library of Congress is the largest library in the world, with millions of books, films and video, audio recordings, photographs, newspapers, maps and manuscripts in its collections. The Library is the main research arm of the U.S. Congress and the home of the U.S. Copyright Office.

Dataset of World Digital Library of Congress

<https://www.loc.gov/item/2020446966/>

- We condensed this dataset because it was too large to upload to GitHub. We removed some of the rows of data as well as some of the 'less significant' columns
- Our final dataset had 23,861 rows and 99 columns

ETHICAL CONSIDERATION FOR DATA ENGINEERING – BEST PRACTICES

- Ownership
- Transparency
- Privacy
- Intention
- Outcomes

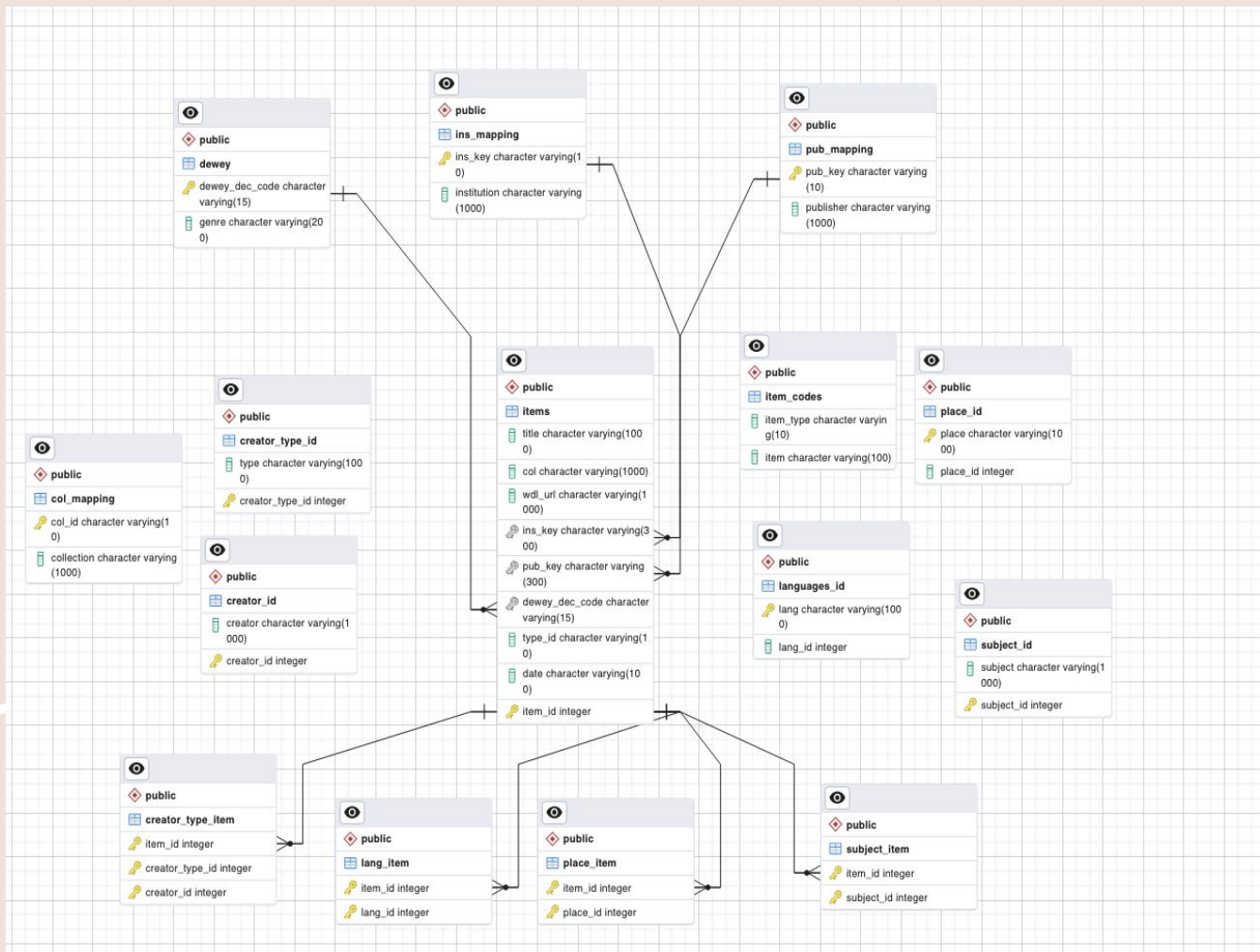
The dataset we used for our project did not contain any personally identifiable information (PII) other than the names of the creators of the items included in the dataset. However, since there wasn't any other PII (like contact information, addresses, or social security numbers) we did not take any special measures to protect the creator's information.

We chose to use a relational database because the data from our source file had some structure to it already. Each item had very similar characteristics and shared most of the features from the dataset, so it made more sense to use a relational database with a rigid structure rather than a non-relational database like NoSQL. To implement the relational database we used Postgres in the form of pgAdmin4 because everyone in the group had already used it before in class and there was no cost associated with it.

Database Choice & Reasoning

ERD

Postgres/pgAdmin4
Relational Database



ETL Process



Dask Python Library

#Demonstrating the addition of a library that has not been taught in class. However, will continue with Pandas due to time constraint

```
import dask

if dask.__version__:
    print(f"Dask version {dask.__version__} is installed.")
else:
    print("Dask is not installed. You can install it using 'conda install dask' or 'pip install dask[complete]'.")
```

Dask version 2023.6.0 is installed.

#Using DASK to read the data, demonstrating use of the library
#Reading in the source file for the project in DASK

```
file_path = 'Resources/wdl_data_en_reduced.csv'
# Specifying encoding as 'ISO-8859-1' to read file
dd_loc_df = dd.read_csv(file_path, encoding='ISO-8859-1')
```


ETL Workflows

- We needed to split the large flat file (the dataset from loc.gov) into relations that could be used in our database
- To do this, we used ETL workflows to read the dataset into Pandas Dataframes, manipulate it, and export the final product into corresponding .csv files

	Subject	item_id	subject_id
0	Antietam, Battle of, Maryland, 1862	0	0
1	Antietam, Battle of, Maryland, 1862	11930	0
2	Generals	1011	1
3	Generals	1016	1
4	Generals	1017	1
...
113573	Popular culture	23859	4640
113574	Mohammad Daoud, Sardar, 1909-1978	11920	4641
113575	Mohammad Daoud, Sardar, 1909-1978	23850	4641
113576	Israel-Arab War, 1973	11929	4642
113577	Israel-Arab War, 1973	23859	4642

113578 rows × 3 columns

	Creator	item_id	type
0	Gardner, Alexander, 1821-1882	0	Photographer
1	Vargas, Max T., 1874-1959	1	Photographer
2	Sandberg, Bob	27	Photographer
3	Brumfield, William Craft, 1944-	31	Photographer
4	Ferrez, Marc, 1843-1923	41	Photographer
...
26749	Vahhāj, Sirāj al-Dīn	23850	Lead
26750	Vahhāj, Sirāj al-Dīn	23851	Lead
26751	Vahhāj, Sirāj al-Dīn	23852	Lead
26752	Vahhāj, Sirāj al-Dīn	23853	Lead
26753	Vahhāj, Sirāj al-Dīn	23854	Lead

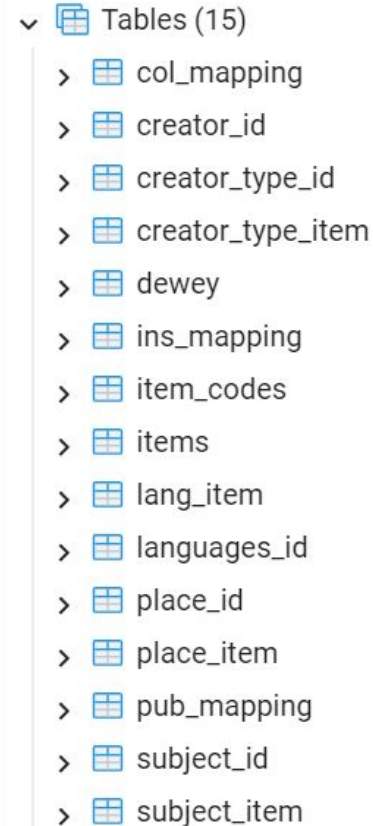
26754 rows × 3 columns

Final .csv files

- Dewey.csv
- col_mapping_new.csv
- creator_id.csv
- creator_type_id.csv
- creator_type_item.csv
- ins_mapping_new.csv
- item_codes.csv
- items.csv (Note: this file has '|' as the delimiter, rather than ',')
- items_csv (this is the same as the items.csv, except that this file has ',' as the delimiter)
- lang_item.csv
- languages_id.csv
- place_id.csv
- place_item.csv
- pub_mapping_new.csv
- subject_id.csv
- subject_item.csv

Implementing a Database in pgAdmin4

- To implement the database, we created SQL schema to create tables and then imported our final .csv files into the tables
- Once the data is imported, SQL statements can be used to view, manipulate, and edit the data



Ex Query

```
4 select title, subject from items i join subject_item si on i.item_id = si.item_id
5 left join subject_id s on s.subject_id = si.subject_id
```

Data Output

Messages

Notifications



	title character varying (1000)	subject character varying (1000)
69	Special Edition of Universal Current Events: Late...	Military camps
70	Special Edition of Universal Current Events: Late...	Military camps
71	View of the Encampment of the Corn Exchange ...	Military camps
72	View of the Encampment of the Corn Exchange ...	Military camps
73	Grand Review	Military camps
74	Grand Review	Military camps
75	President George Washington	Presidents
76	President Abraham Lincoln	Presidents
77	President Millard Fillmore	Presidents
78	President Jefferson Davis, Confederate States of...	Presidents
79	President George Washington	Presidents
80	President Abraham Lincoln	Presidents
81	President Millard Fillmore	Presidents
82	President Jefferson Davis, Confederate States of...	Presidents
83	Special Edition of Universal Current Events: Late...	Tents

Flask API

Available Routes:

[/api/v1.0/col_mapping](#)

[/api/v1.0/creator_id](#)

[/api/v1.0/creator_type_id](#)

[/api/v1.0/creator_type_item](#)

[/api/v1.0/dewey](#)

[/api/v1.0/ins_mapping](#)

[/api/v1.0/item_codes](#)

[/api/v1.0/items](#)

[/api/v1.0/lang_item](#)

[/api/v1.0/languages_id](#)

[/api/v1.0/place_id](#)

[/api/v1.0/place_item](#)

[/api/v1.0/pub_mapping](#)

[/api/v1.0/subject_id](#)

[/api/v1.0/subject_item](#)

JSON	Raw Data	Headers
Save	Copy	Collapse All Expand All (slow) Filter JSON
▶ 0:		{-}
▶ 1:		{-}
▶ 2:		{-}
▼ 3:		
col:		"col_3"
date:		"1862"
dewey_dec_code:		"769.792"
ins_key:		"ins_1"
item_id:		3
pub_key:		null
title:		"Warrior Asahina Kobayashi"
type_id:		"prph"
wdl_url:		" https://www.wdl.org/en/item/4 "
▶ 4:		{-}
▶ 5:		{-}

CONCLUSION

In summation, pertinent information was extracted from the Library of Congress database to draw aggregate findings and design a database useful for the future utilization.

- *ETL workflows were implemented.*
- *SQL, Pandas, JSON, Flask API*
- *Dask Library*
- *JavaScript, Python Scripts*
- *Jupyter Notebook*
- *Final data was uploaded into Postgres to create a relational Database to aid in making it more user friendly.*

Sources

- Dask documentation: <https://docs.dask.org/en/stable/>
- Dataset source: <https://www.loc.gov/item/2020446966/>