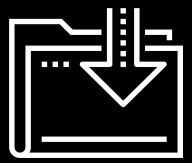
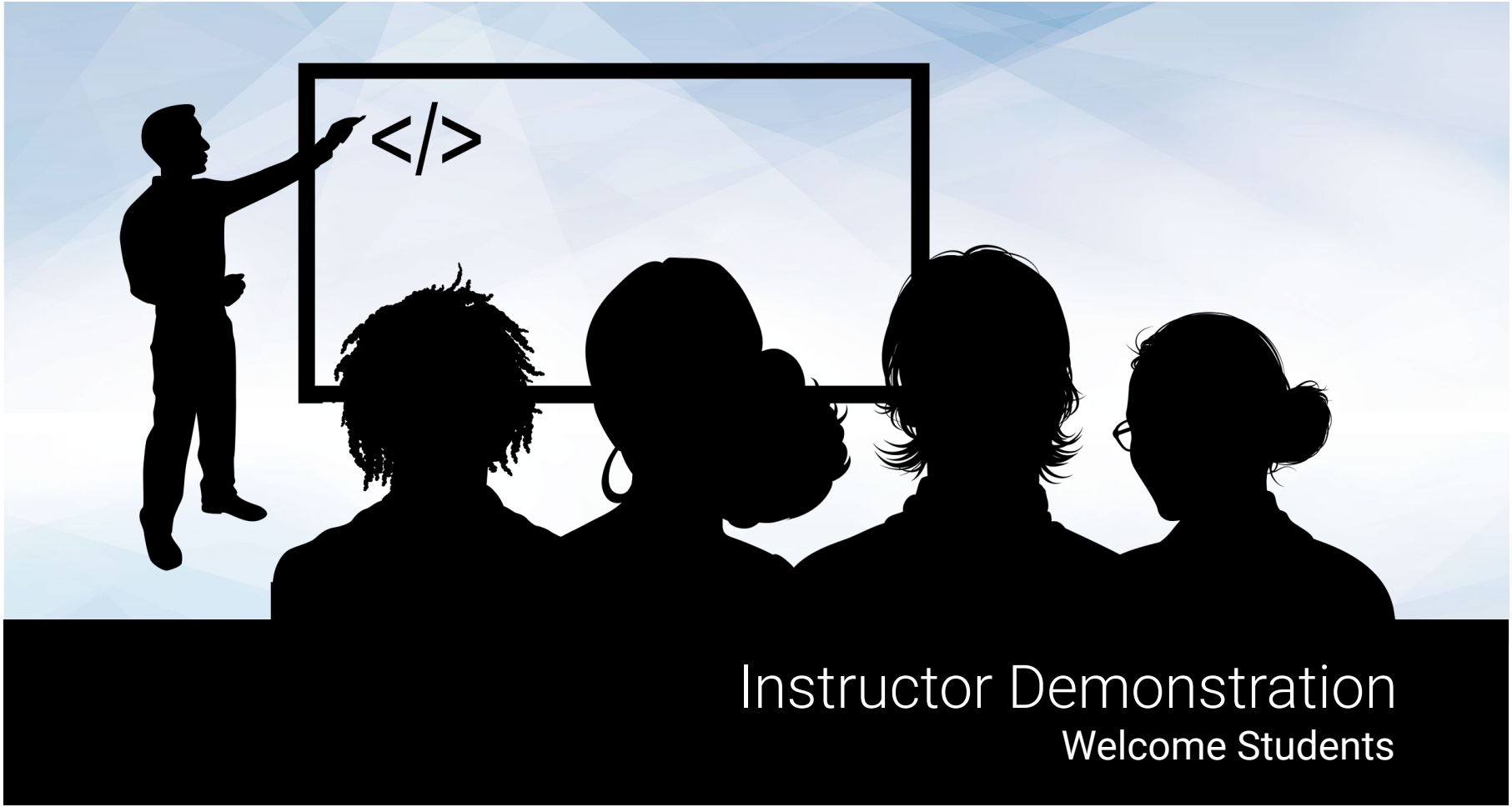
Project #2: Extract, Transform, and Load

Data Boot Camp

Lesson 13.1





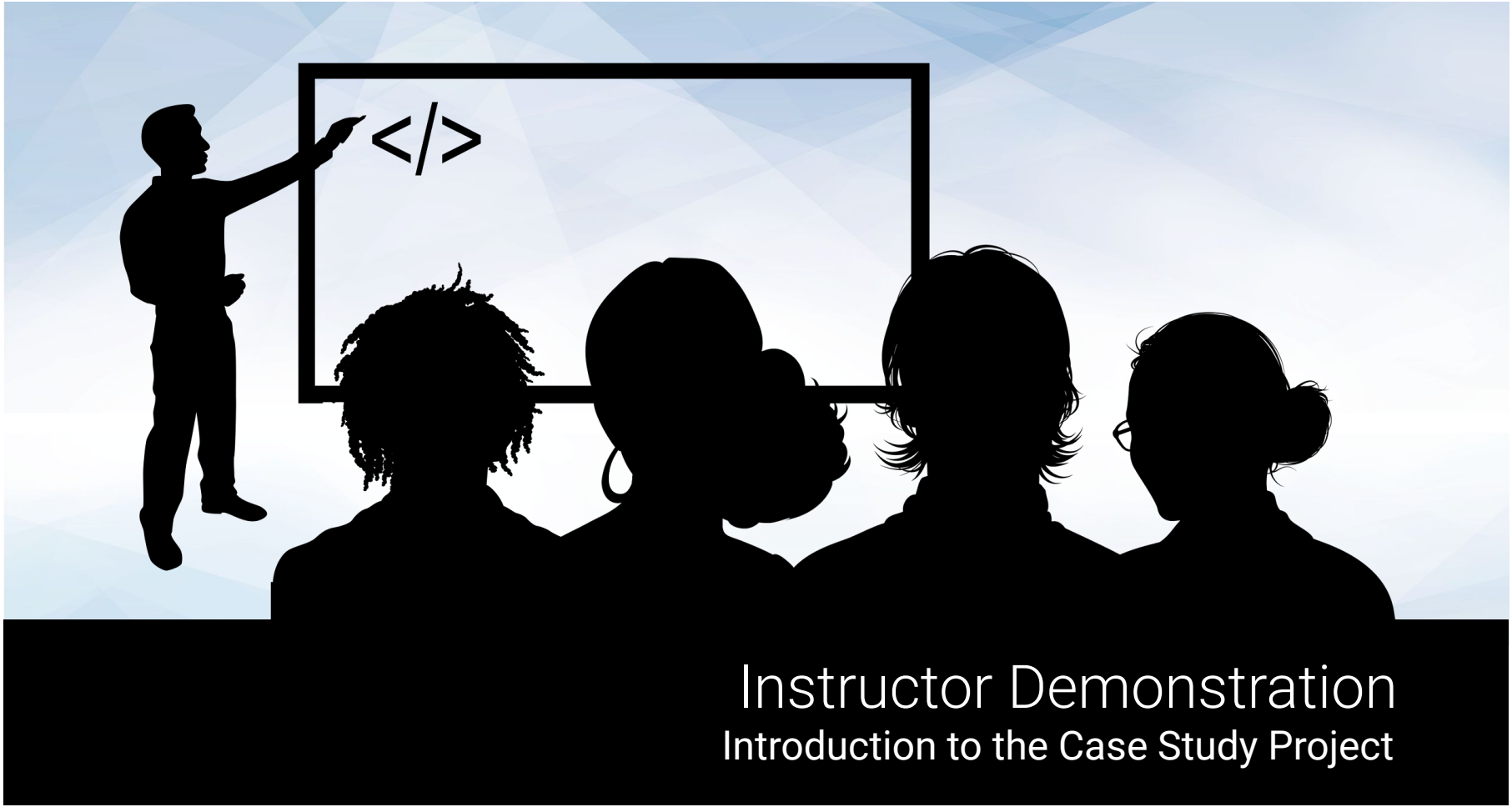
Instructor Demonstration

Welcome Students


WELCOME

The Week Ahead!

- **Day 1 (today):**
  - / Introduction to the ETL process: working through activities
  - Introduction to the ETL project
    - / Goals
    - / Requirements
  - Working toward a feasible project idea with:
    - / Instructors
    - / TAs
  - Submit project proposal
- **Day 2:**
  - / Working on projects
  - / Full assistance from instructors and TAs
- **Day 3:**
  - / Project due date: presentation!
  - / Discussion



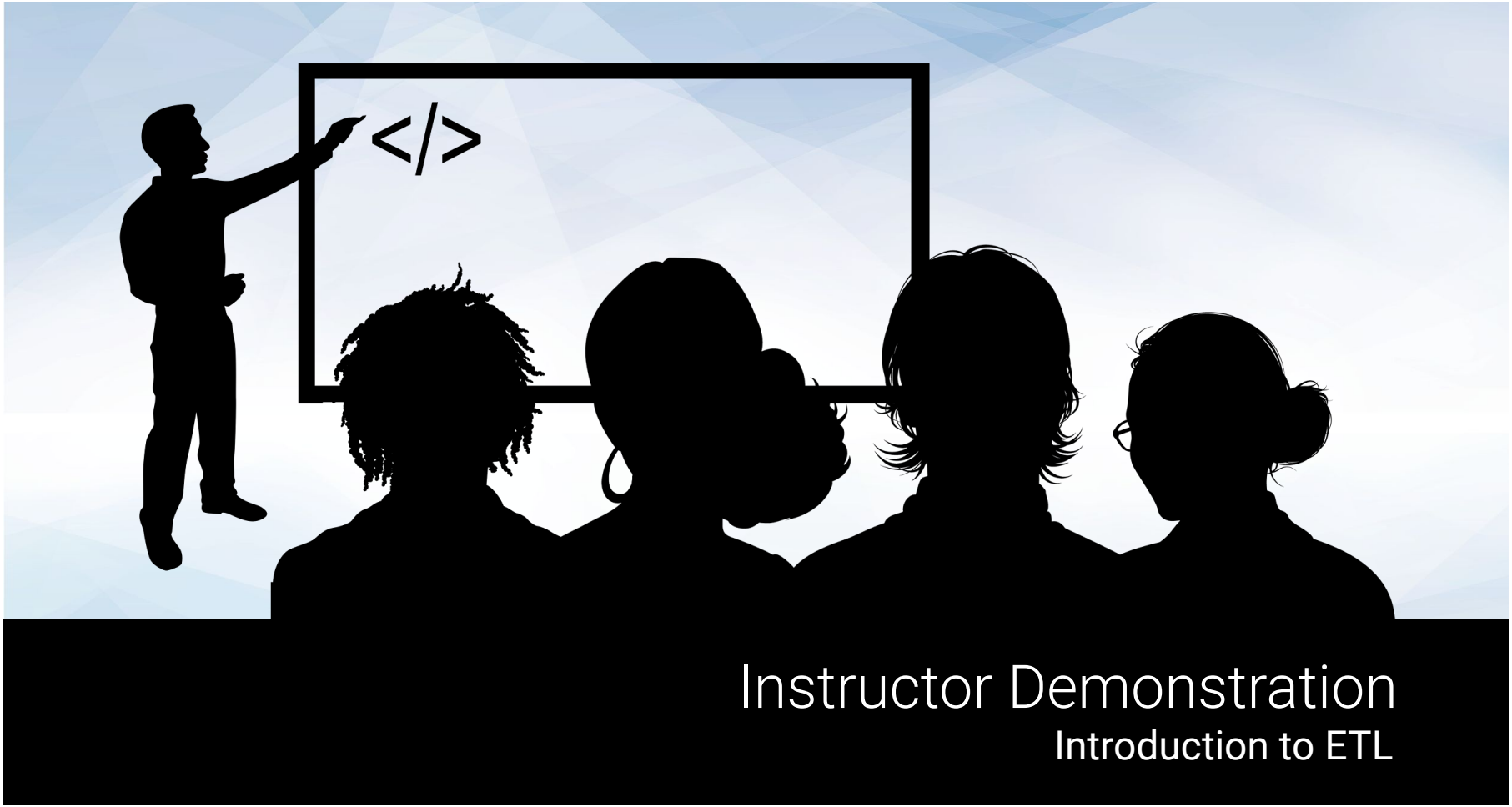
Instructor Demonstration

Introduction to the Case Study Project

Case Study Project Requirements

- You must have two (minimum) or more sources.
- Recommended sources:
  - Kaggle
  - Data.world
  - Google Dataset Search (https://datasetsearch.research.google.com/)
  - APIs may be used as an alternative source

- Once your datasets are identified, perform ETL and create documentation.
  - Documentation must have:
    - Datasets used and their sources
    - Types of data wrangling performed (data cleaning, joining, filtering, and aggregating)
    - The schemata used in the final production database



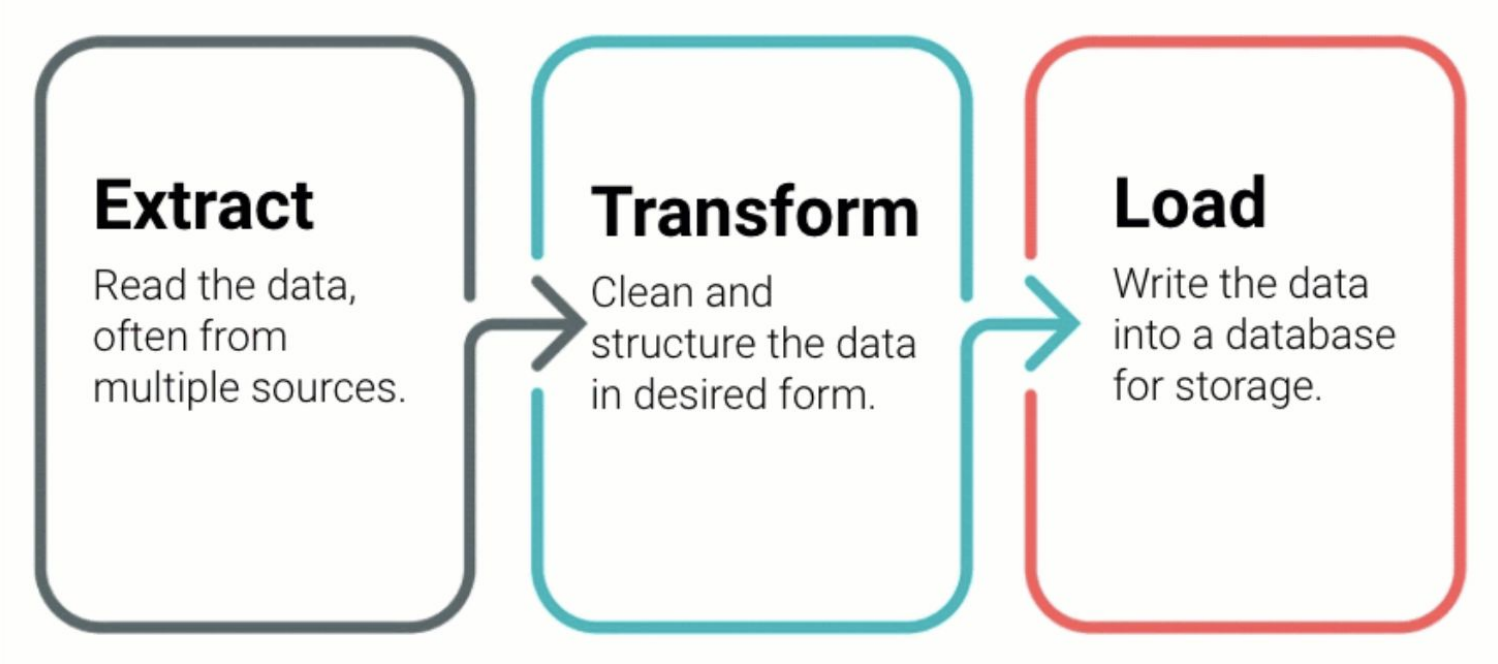
Instructor Demonstration

Introduction to ETL

Introduction to ETL

ETL

Data integration is an important part of working with data.

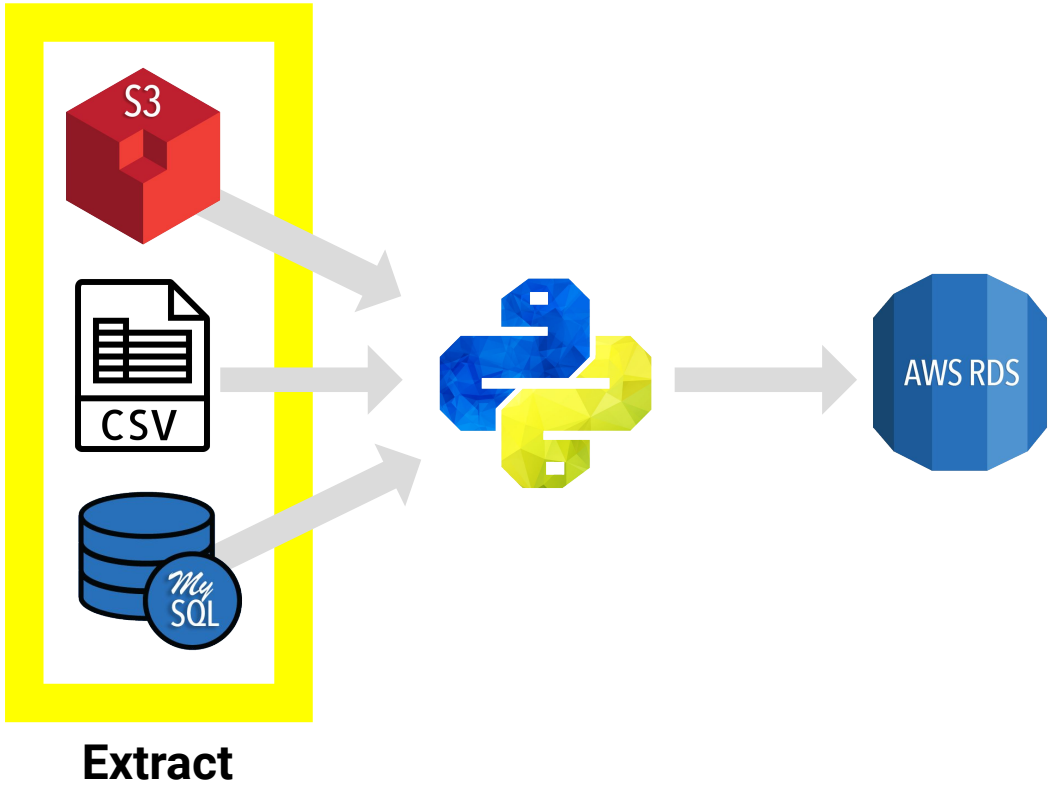


Introduction to ETL

Extract

Data may come from disparate sources, such as:

- CSV files
- JSON files
- HTML tables
- SQL databases
- Spreadsheets

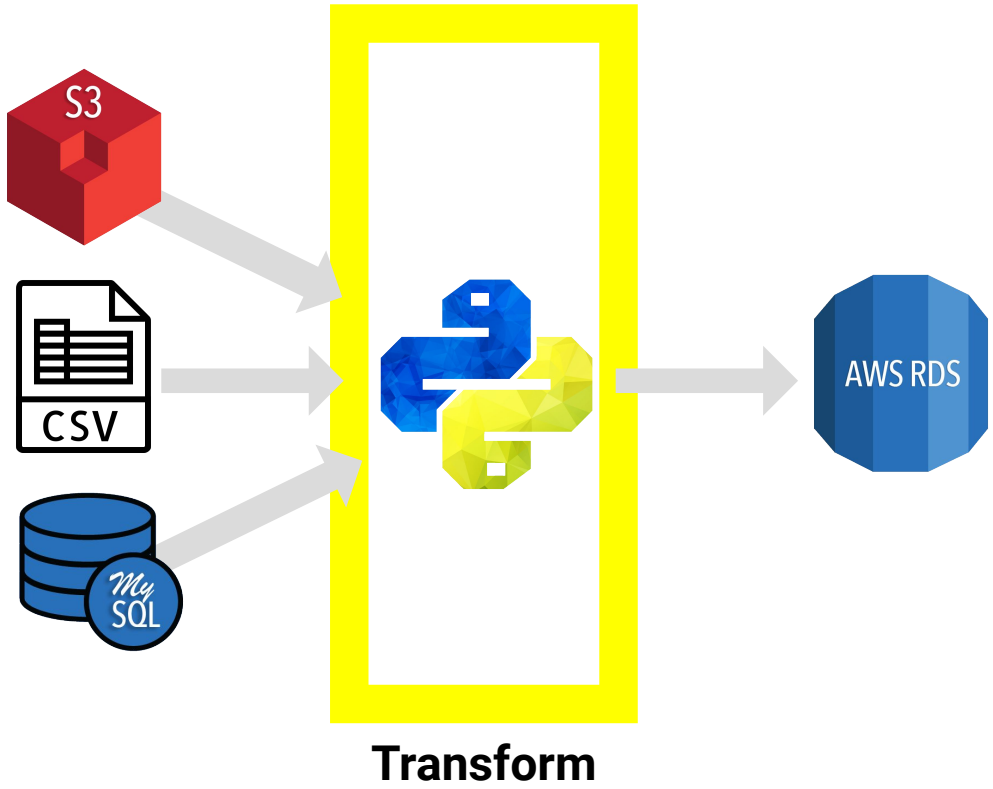


Introduction to ETL

Transform

Transform the data to suit business needs.
This may include:

- Data cleaning
- Summarization
- Selection
- Joining
- Filtering
- Aggregating



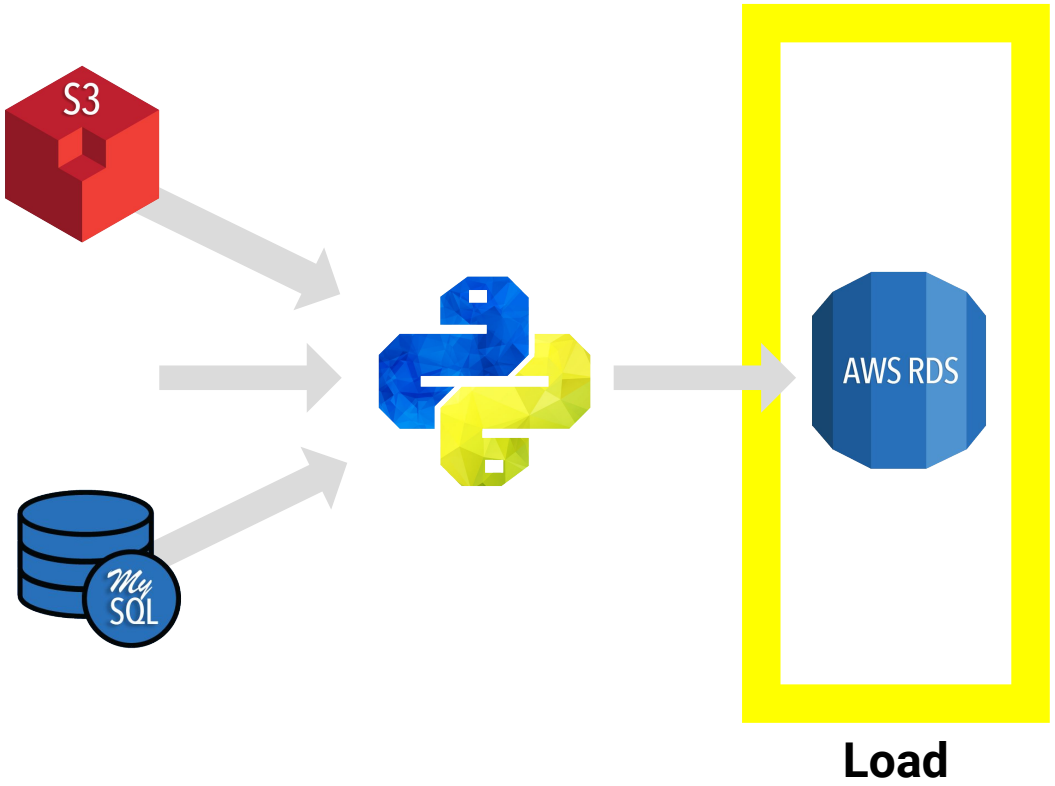
Note: We will use Python and Pandas for transformation, which can also be done with SQL or a specialized ETL tool.

Introduction to ETL

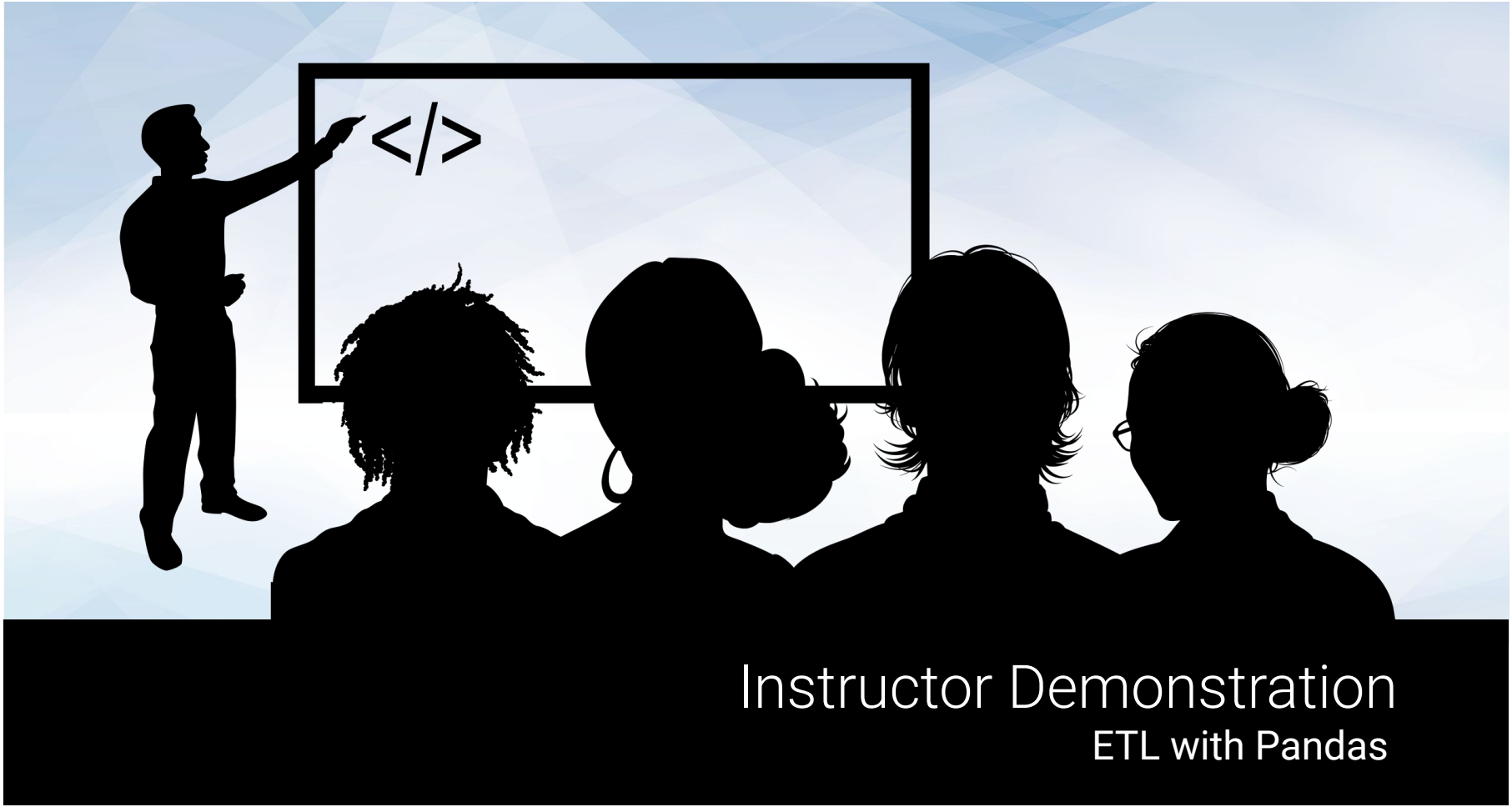
Load

Load the data into a final database that can be used for future analysis or business use.

- Can be a relational or non-relational database
- Can be local or in the cloud
- Can be a data lake or data warehouse



Questions?

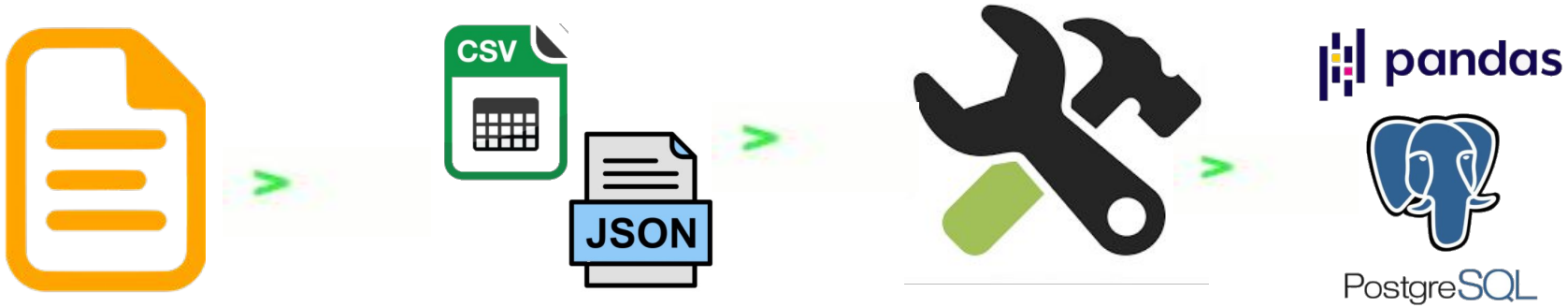


Instructor Demonstration

ETL with Pandas

ETL with Pandas

- Not limited to **Pandas**, the **ETL** process is performed with a variety of tools and file formats.
- For this demonstration, we will use the following:



Couple of things to prepare before we move forward! Let’s find out what they are!

ETL with Pandas

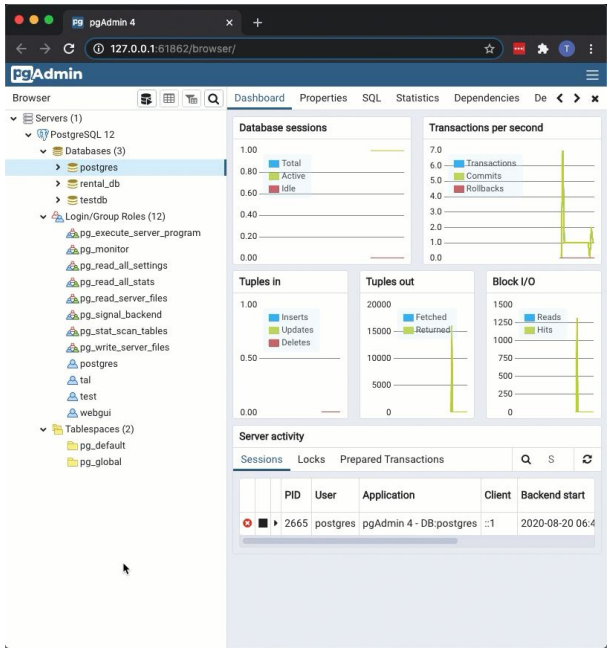
`pip install psycopg2`

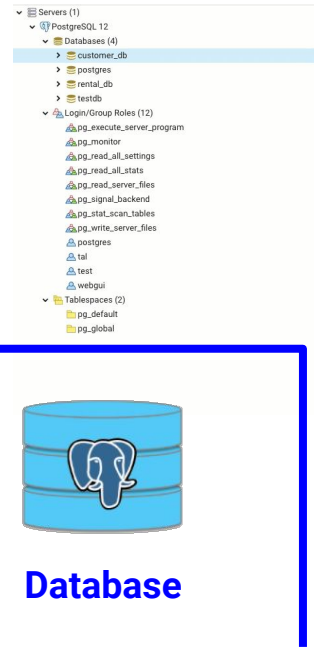
- ➔ `pip install psycopg2`
- ➔ Psycopg is a package—an adapter for Python that works as a wrapper for **libpq**, which is the official postgreSQL client library

ETL with Pandas

pgAdmin postgreSQL

➔ Second, we need to open pgAdmin 4 and connect to a local server. Once connected, we need to create a new **database** and **tables** accordingly.





Tables

```
1  CREATE TABLE customer_name (
2  id INT PRIMARY KEY,
3  first_name TEXT,
4  last_name TEXT
5  );
6
7  CREATE TABLE customer_location (
8  id INT PRIMARY KEY,
9  address TEXT,
10 us_state TEXT
11 );
```

ETL with Pandas

.ipynb (Jupyter Notebook)

- Pandas is the pivotal piece of the ETL process. In Pandas, we are going to extract the data, transform them, and them load back into data frames. Let's follow the code line by line and see how it is done.

Store CSV into DataFrame

```
In [2]: csv_file = "../Resources/customer_data.csv"
        customer_data_df = pd.read_csv(csv_file)
        customer_data_df.head()
```

Out[2]:

	id	first_name	last_name	email	gender	car
0	1	Benetta	Cancott	bcancott0@studiopress.com	Female	Scion
1	2	Lilyan	Cherry	lcherry1@deliciousdays.com	Female	Chrysler
2	3	Ezekiel	Benasik	ebenasik2@wikia.com	Male	Mercedes-Benz
3	4	Kennedy	Atlay	katlay3@so-net.ne.jp	Male	Buick
4	5	Sanford	Salmen	ssalmen4@reuters.com	Male	Lincoln

Create new data with select columns

```
In [3]: new_customer_data_df = customer_data_df[['id', 'first_name', 'last_name']].co
        py()
        new_customer_data_df.head()
```

Out[3]:

	id	first_name	last_name
0	1	Benetta	Cancott
1	2	Lilyan	Cherry
2	3	Ezekiel	Benasik
3	4	Kennedy	Atlay
4	5	Sanford	Salmen

➔ Cell 2 has data pulled from a CSV file and the data are assigned to a variable called `customer_data_df`.

➔ Cell 3 is returning a new data frame with only the necessary columns. The new data frame is assigned to a new variable as well.

ETL with Pandas

.ipynb (Jupyter Notebook)

Store JSON data into a DataFrame ¶

In [4]:
```
json_file = "../Resources/customer_location.json"
customer_location_df = pd.read_json(json_file)
customer_location_df.head()
```

Out[4]:

	address	id	latitude	longitude	us_state
0	043 Mockingbird Place	1	39.1682	-86.5186	Indiana
1	4 Prentice Point	2	41.0938	-85.0707	Indiana
2	46 Derek Junction	3	32.7673	-96.7776	Texas
3	11966 Old Shore Place	4	39.0350	-94.3567	Missouri
4	5 Evergreen Circle	5	40.7808	-73.9772	New York

Clean DataFrame

In [5]:
```
new_customer_location_df = customer_location_df[["id", "address", "us_stat
e"]].copy()
new_customer_location_df.head()
```

Out[5]:

	id	address	us_state
0	1	043 Mockingbird Place	Indiana
1	2	4 Prentice Point	Indiana
2	3	46 Derek Junction	Texas
3	4	11966 Old Shore Place	Missouri
4	5	5 Evergreen Circle	New York

➔ The same process of extracting and transforming the data is repeated with the JSON file as well.

ETL with Pandas

.ipynb (Jupyter Notebook)

Connect to local database

```
In [6]: rds_connection_string = "<insert user name>:<insert password>@localhost:5432/customer_db"
engine = create_engine(f'postgresql://{rds_connection_string}')
```

Check for tables

```
In [7]: engine.table_names()
Out[7]: ['customer_location', 'customer_name']
```

Use pandas to load csv converted DataFrame into database

```
In [8]: new_customer_data_df.to_sql(name='customer_name', con=engine, if_exists='append', index=False)
```

Use pandas to load json converted DataFrame into database

```
In [9]: new_customer_location_df.to_sql(name='customer_location', con=engine, if_exists='append', index=False)
```

- ➔ The following step is to connect to the local database. Once connected, we check the tables created earlier in the process.
- ➔ Next, we are dumping the newly created and trimmed data frames into the database.

ETL with Pandas

.ipynb (Jupyter Notebook)

Confirm data has been added by querying the customer_name table

- NOTE: can also check using pgAdmin

```
In [10]: pd.read_sql_query('select * from customer_name', con=engine).head()
```

Out[10]:

	id	first_name	last_name
0	1	Benetta	Cancott
1	2	Lilyan	Cherry
2	3	Ezekiel	Benasik
3	4	Kennedy	Atlay
4	5	Sanford	Salmen

Confirm data has been added by querying the customer_location table

```
In [11]: pd.read_sql_query('select * from customer_location', con=engine).head()
```

Out[11]:

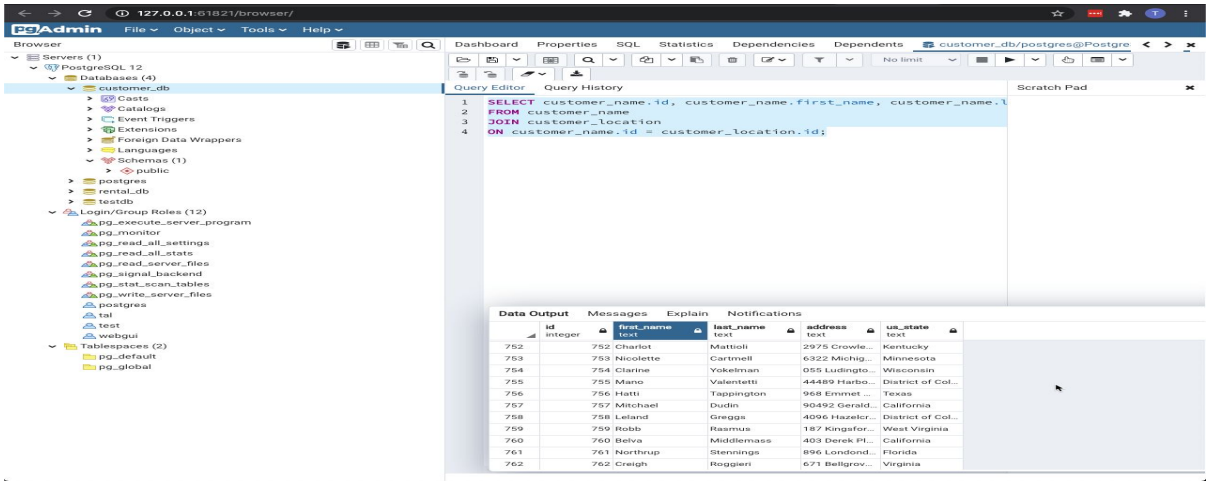
	id	address	us_state
0	1	043 Mockingbird Place	Indiana
1	2	4 Prentice Point	Indiana
2	3	46 Derek Junction	Texas
3	4	11966 Old Shore Place	Missouri
4	5	5 Evergreen Circle	New York

- ➔ At this point, all the data that we extracted and transformed are successfully loaded into our postgreSQL database.
- ➔ To double-check, as a best practice, we performed queries for both tables at the database.

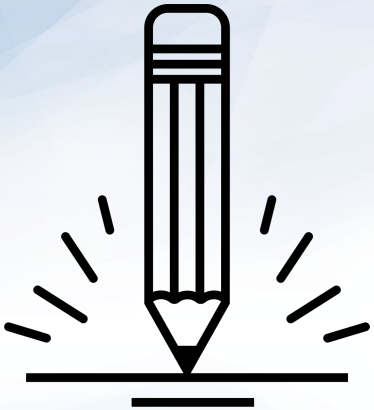
ETL with Pandas

pgAdmin postgreSQL

- The last piece of the process is coming back to pgAdmin to perform the join of the two tables we created.




```
SELECT customer_name.id, customer_name.first_name, customer_name.last_name, customer_location.address,
customer_location.us_state
FROM customer_name
JOIN customer_location
ON customer_name.id = customer_location.id;
```



Activity: Pandas ETL

In this activity, you will have the opportunity to perform your very first ETL process.

Suggested Time:
20 Minutes

Activity: Pandas ETL

Instructions

- Create a `customer_db` database in pgAdmin 4 and then create the following two tables within:
  - A premise table that contains the columns `id`, `premise_name` and `county_id`.
  - A county table that contains the columns `id`, `county_name`, `license_count` and `county_id`.
  - Be sure to assign a primary key, as Pandas will not be able to do so.
- In Jupyter Notebook, perform all ETL.

➔ **Extraction**
- Put each CSV into a Pandas DataFrame.

➔ **Transform**
- Copy only the columns needed into a new DataFrame.
- Rename columns to fit the tables created in the database.
- Handle any duplicates. **Hint:** Some locations have the same name, but each license number is unique.
- Set index to the previously created primary key.

Activity: Pandas ETL

Instructions

➔ **Load**

- Create a connection to database.
- Check for a successful connection to the database and confirm that the tables have been created.
- Append DataFrames to tables. Be sure to use the index set earlier.

- Confirm successful **load** by querying database.
- Join the two tables and select the `id` and `premise_name` from the `premise` table and `county_name` from the `county` table.

Time’s Up! Let’s Review.

Countdown timer

15:00

(with alarm)

Project Overview

Project Week! (This Week)

Day 1:

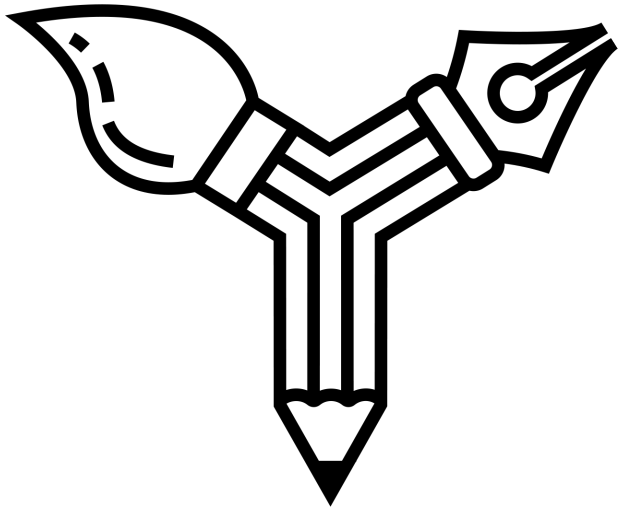
- Form groups (3-5 people each)
- Identify datasets
- Perform ETL on the data

Day 2:

- Database development

Day 3:

- Complete final report



Project Proposals

Team effort

Due to the short timeline, teamwork will be crucial to the success of this project! Work closely with your team through all phases of the project to ensure that there are no surprises at the end of the week.

Working in a group enables you to tackle more difficult problems than you'd be able to work on alone. In other words, working in a group allows you to work smart and dream big. Take advantage of it!

Project proposal

Before you start writing any code, remember that you only have one week to complete this project. View this project as a typical assignment from work. Imagine that a bunch of data came in and you and your team are tasked with migrating it to a production database.

Take advantage of your instructor and TA support during office hours and class project work time. They are a valuable resource and can help you stay on track.

Project 2: ETL

Data Cleanup and Analysis Requirements

You will also be responsible for:

- Citing the data sources from which you will extract.
- Extracting the data from their existing locations.
- Transforming the data (i.e. cleaning, joining, filtering, aggregating, etc).
- Loading the data to a database (relational or non-relational).

Report Requirements

You will also be responsible for preparing a formal report that covers:

Extract: your original data sources and how the data were formatted (CSV, JSON, pgAdmin 4, etc.).

Transform: what data cleaning or transformation was required.

Load: the final database, tables/collections, and why this was chosen.

Project Rubric

Rubric at a Glance

Categories for grading

Project proposal (20 points)

Technical report (20 points)

GitHub repository (20 points)

Data Suggestions

Data Suggestions

Feel free to ask us (the instructional staff) for input, but our general advice is to stick to data sources that:

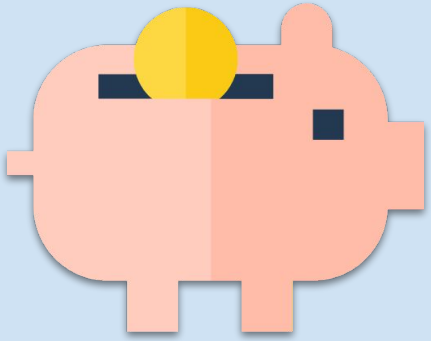
- Are sufficiently large.
- Have a consistent format.
- Ideally, contain more data than needed.
- Are well documented.

Choosing a Project Track

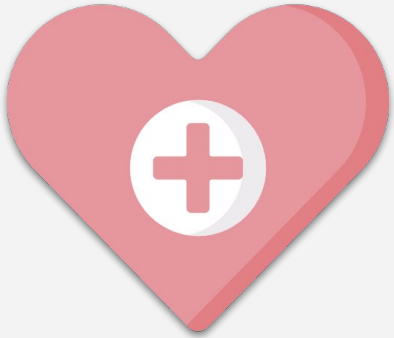
Choosing a Project Track

This project gives you the ability to focus your efforts within a specific industry.

Here are the specializations:



Finance



Healthcare

Custom

ETL and Finance

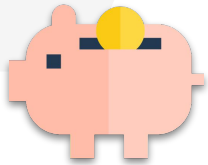
When to use ETL in finance

Current treasury benchmarks are at an all-time low, and a financial analyst has decided to study the last 30 years worth of rates.

After pulling historic data, they clean and explore them to perform analysis, with the intent of predicting future benchmark trends.

Once the history data have been collected, processed, and loaded into a database, the financial analyst turns their attention to current data. Using an API, they pull the most up-to-date information with the intent of adding it to their established database.

They've already extracted the new data, but before loading them into the existing database, they need to ensure that they have the correct format. Once the data are transformed, they can load them to the database and continue on with the analysis.



ETL and Healthcare

When to use ETL in healthcare

An analyst working at a major hospital is tasked with reviewing policies regarding the upcoming flu season. They're keeping the following questions in mind:

- How many patients does the hospital expect this year?
- How severe will the flu season be this year?
- Will there be regional differences? Similarities?

The analyst wants to collect data from different sources to analyze them and predict the future flu season.

Before combining the hospital's own data with regional data acquired elsewhere, the analyst will need to extract the new data, transform them to match the existing data, and then load them into the database.

ETL in the Wild

Customize it!

There are several other industries utilizing the ETL process as well.

- In marketing, analysts may acquire data from competitors to see how their product fares. Multiple data sources would need to be extracted, transformed, and loaded into a common database prior to analysis.
- An analyst working for a large retail chain finds themselves in charge of moving a legacy database into a cloud-based data warehouse.
- An entrepreneur has a big business idea but wants to get a feel for their product idea. They use web scraping and APIs to pull data from a variety of social media platforms with the intent of analyzing consumer reactions.

Time to divide into teams!

Questions?