DEALING WITH DATA

SPRING 2019: INFO-GB.2346.30

PROFESSOR GUTHRIE COLLIN

TEACHING FELLOW AJINKYA WALIMBE

NYU STERN

CLASS 11-A:
GROUP PROJECT PRESENTATION PREP

MAY 2, 2019

GROUP PROJECT PRESENTATION PREP

1. Group Project Submission Instructions
2. Pipeline Review
3. Class Project and Example Submissions:
    1. ZIP file
    2. README.txt
4. Grading the Group Project and Individual Contributions
    1. Instructor Grading Considerations
    2. Student Scoring
5. Presentation Order for May 9 (Python)

GROUP PROJECT SUBMISSION INSTRUCTIONS

- One member of each group must submit both of these files to the NYU Classes *Group Project* Assignment:
  1. A ZIP file of the group project's raw data, Python code, SQL database & SQL code (if SQL was used), cleaned/transformed data, and final activated data (before visualization), and a README.txt file
  2. A PDF of the group slide deck that was presented to the class

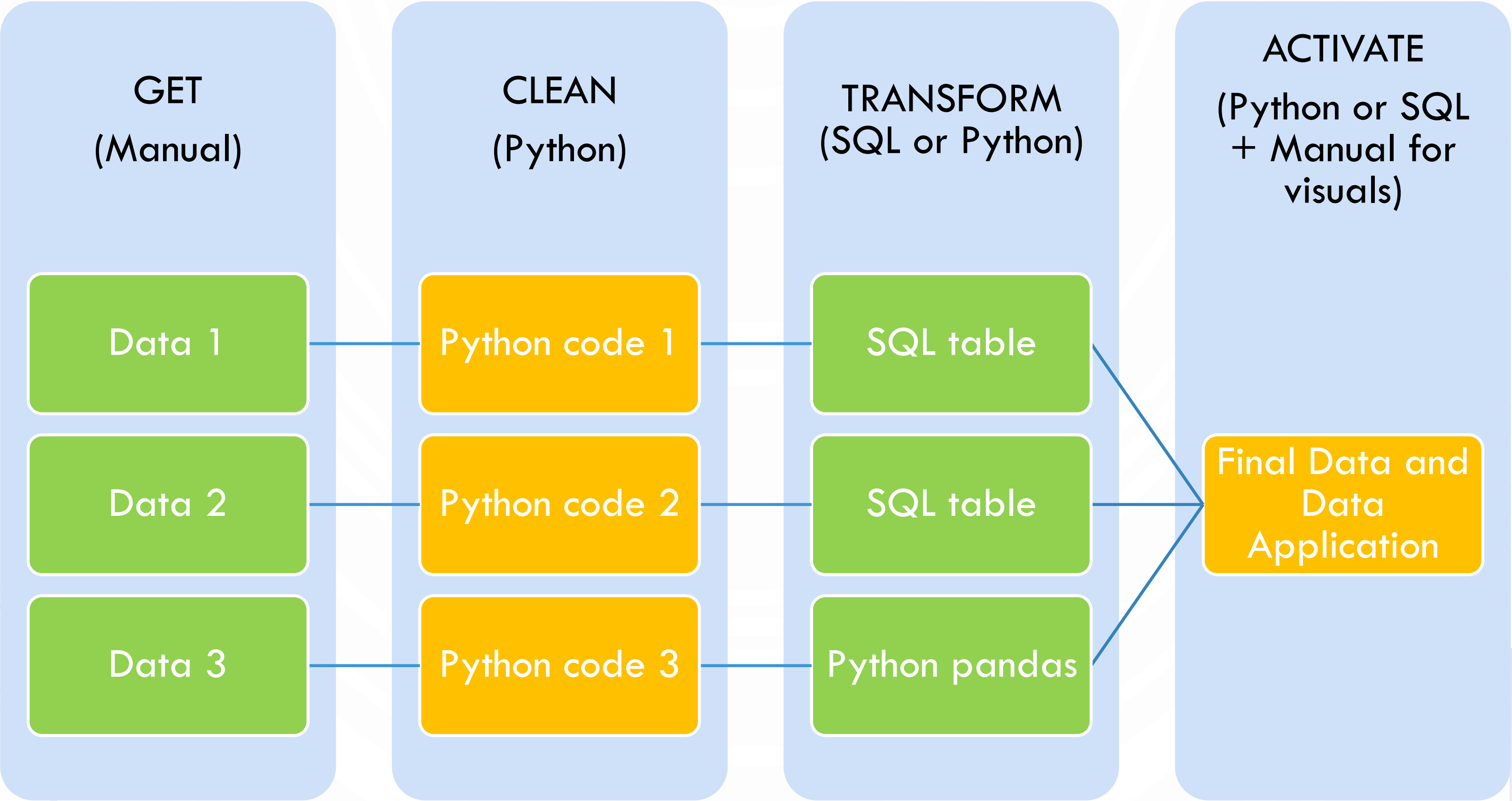
SUBMISSION INSTRUCTION: IMPORTANT NOTE ABOUT FILE SIZE

- **NYU Classes has a 600 MB upload limit!**
- If your group project ZIP file is larger than 600 MB, you must:
  - upload your group project's code to Google Drive here: https://drive.google.com/open?id=1jVy54Tm3XDQGu8rZW6XRpsFMVvqdFtYj
  - And, make a note in the NYU Classes assignment that you have uploaded the project to Google Drive.
- **EACH GROUP HAS THEIR OWN FOLDER ON GOOGLE DRIVE WITH GROUP-ONLY EDITING PERMISSIONS**

SUBMISSION INSTRUCTIONS: ZIP FILE'S README.TXT FILE

- The group project ZIP file MUST have a README.txt file
- This file MUST provide step-by-step instructions for recreating your project by using the code you wrote to clean and transform raw data using Python and SQL.
- This file MUST also provide links/references to your raw data sources.

GROUP PROJECT PIPELINE



GET

(Manual)

CLEAN

(Python)

TRANSFORM
(SQL or Python)

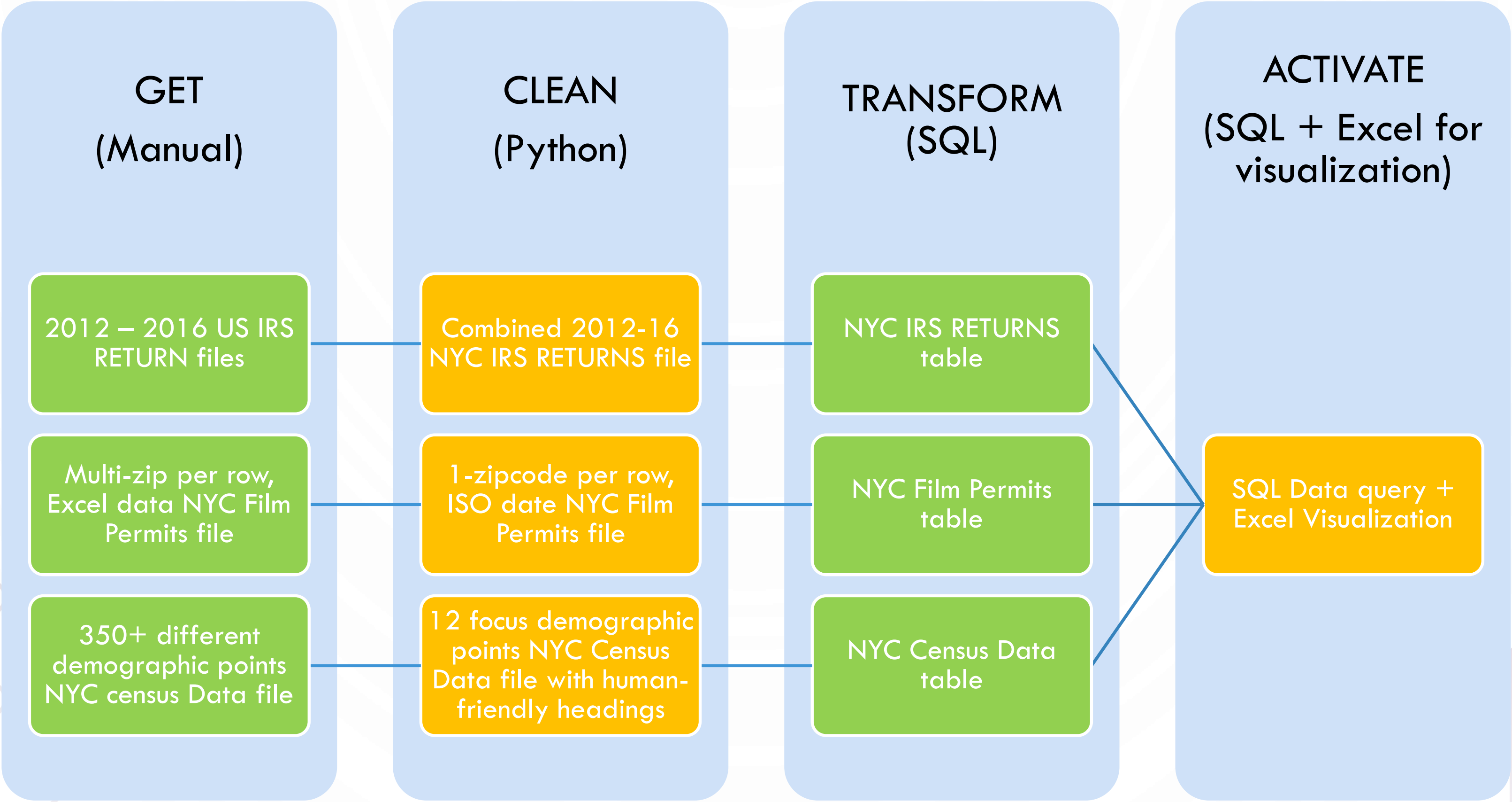
ACTIVATE

(Python or SQL
+ Manual for
visuals)

CLASS PROJECT:
BALANCED NYC MEDIA REPRESENTATION

- **User:** Commissioner for NYC's Media & Entertainment Agency: https://www1.nyc.gov/site/mome/index.page
- **Decision Problem:** Is NYC represented accurately in films based on the permits we issue?

CLASS PROJECT PIPELINE



GET

(Manual)

2012 – 2016 US IRS RETURN files

Multi-zip per row, Excel data NYC Film Permits file

350+ different demographic points NYC census Data file

CLEAN

(Python)

Combined 2012-16 NYC IRS RETURNS file

1-zipcode per row, ISO date NYC Film Permits file

12 focus demographic points NYC Census Data file with human-friendly headings

TRANSFORM (SQL)

NYC IRS RETURNS table

NYC Film Permits table

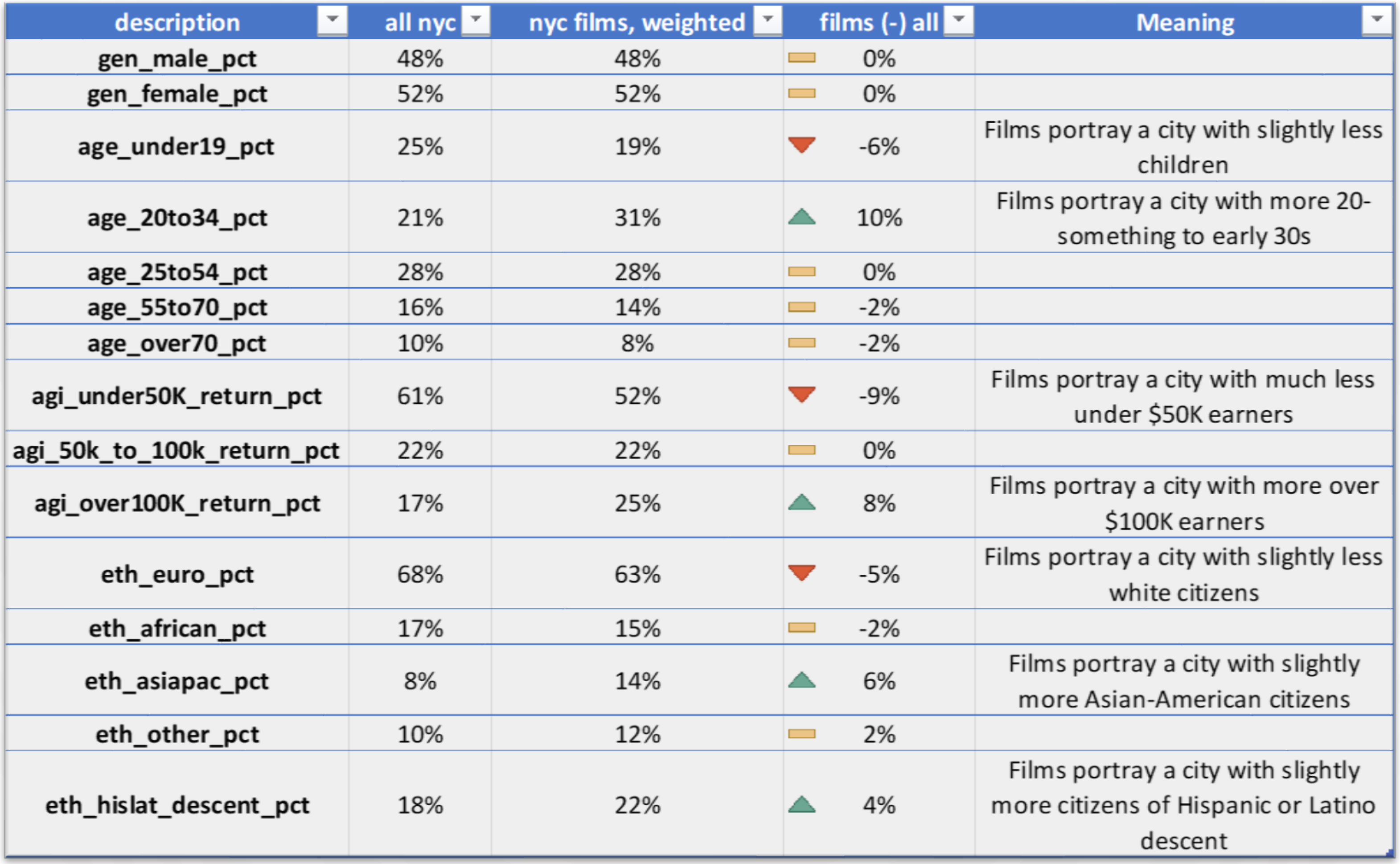
NYC Census Data table

ACTIVATE

(SQL + Excel for visualization)

SQL Data query + Excel Visualization

CLASS PROJECT RESULT: FILMED NYC LOOKED DIFFERENT FROM REAL NYC: 2012-2015

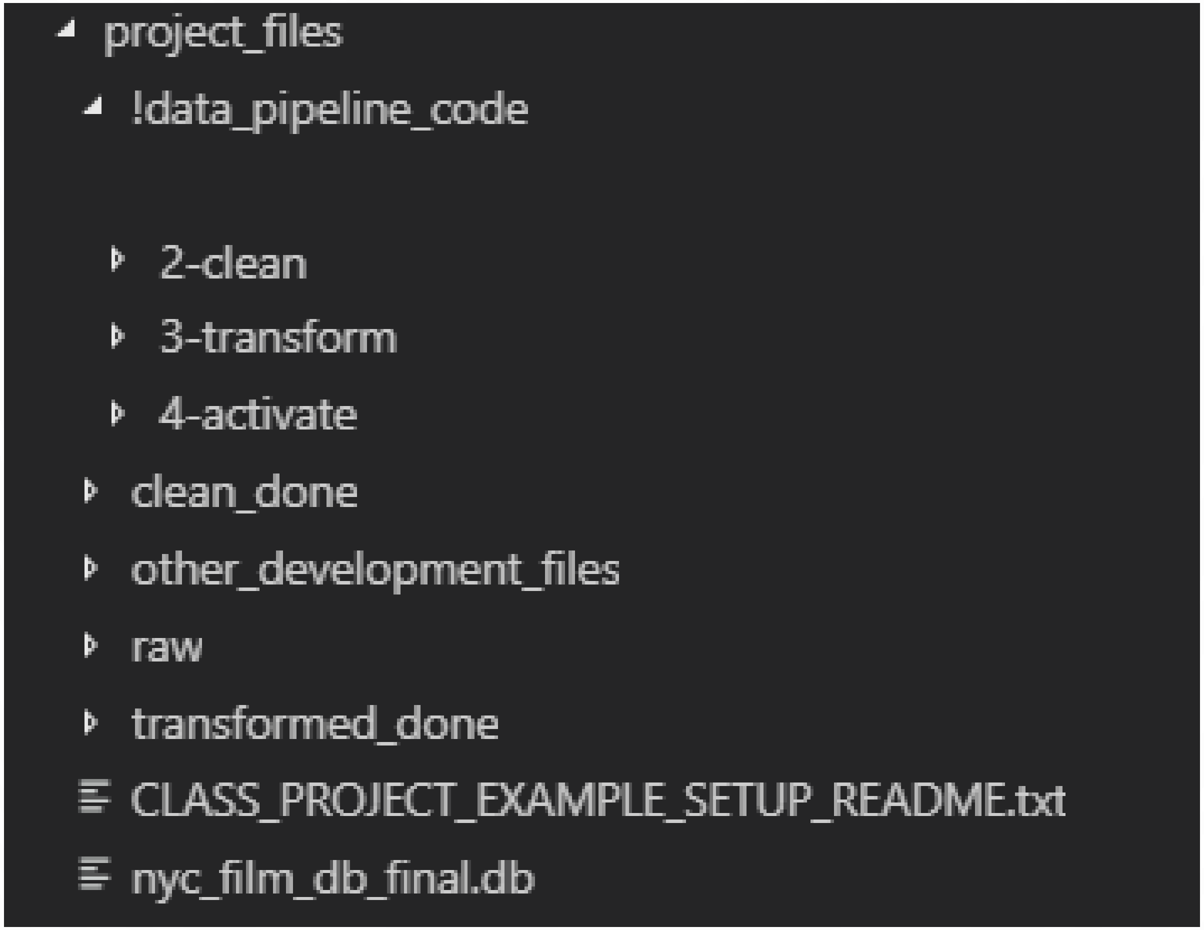


description	all nyc	nyc films, weighted	films (-) all	Meaning
gen_male_pct	48%	48%	▬ 0%	
gen_female_pct	52%	52%	▬ 0%	
age_under19_pct	25%	19%	▼ -6%	Films portray a city with slightly less children
age_20to34_pct	21%	31%	▲ 10%	Films portray a city with more 20-something to early 30s
age_25to54_pct	28%	28%	▬ 0%	
age_55to70_pct	16%	14%	▬ -2%	
age_over70_pct	10%	8%	▬ -2%	
agi_under50K_return_pct	61%	52%	▼ -9%	Films portray a city with much less under $50K earners
agi_50k_to_100k_return_pct	22%	22%	▬ 0%	
agi_over100K_return_pct	17%	25%	▲ 8%	Films portray a city with more over $100K earners
eth_euro_pct	68%	63%	▼ -5%	Films portray a city with slightly less white citizens
eth_african_pct	17%	15%	▬ -2%	
eth_asiapac_pct	8%	14%	▲ 6%	Films portray a city with slightly more Asian-American citizens
eth_other_pct	10%	12%	▬ 2%	
eth_hislat_descent_pct	18%	22%	▲ 4%	Films portray a city with slightly more citizens of Hispanic or Latino descent

CLASS PROJECT CODE FILES: ZIP FILE

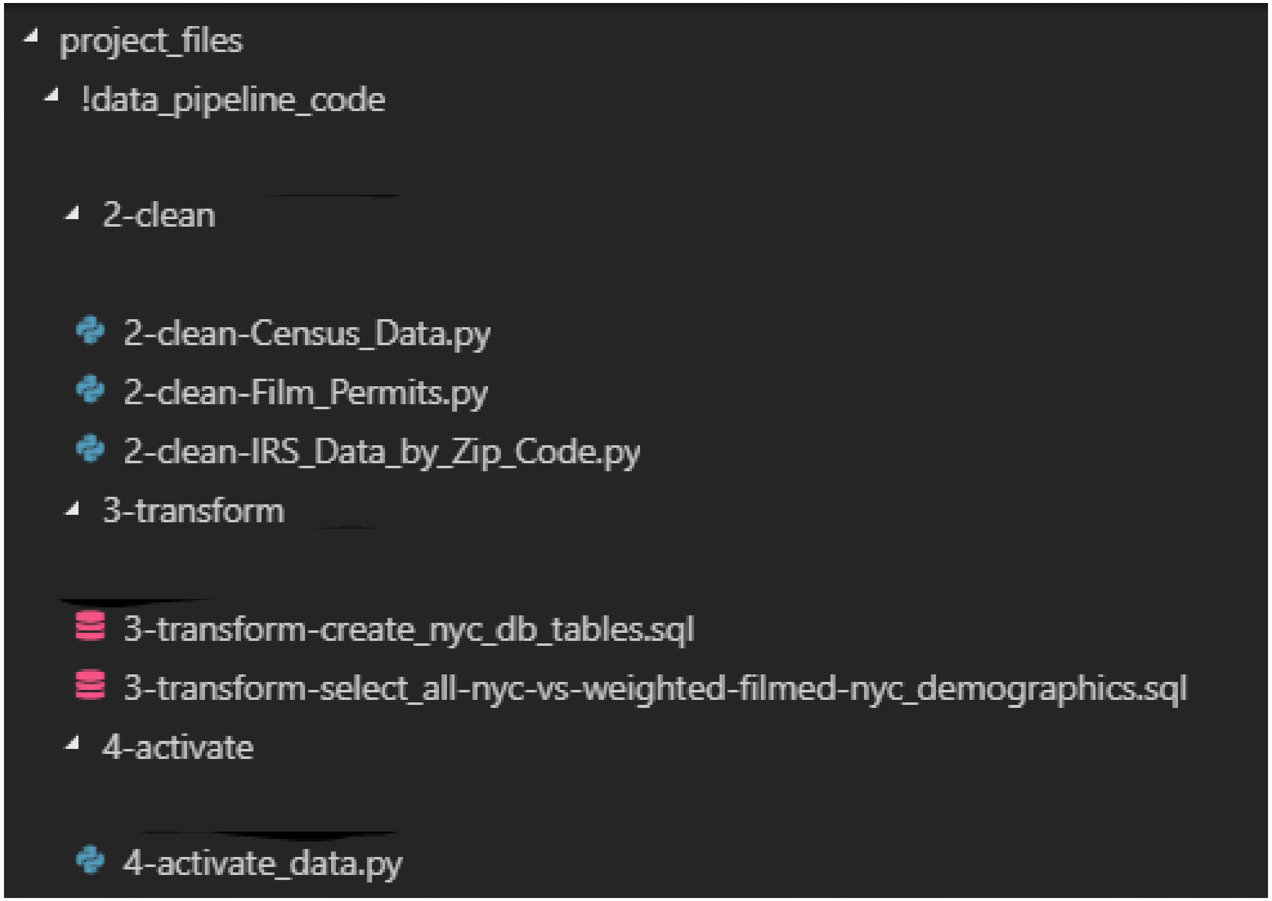
- Example file on NYU Classes: https://newclasses.nyu.edu/portal/site/bf237b8c-f90d-4880-aea4-59ac6b5300e9/page/ea88fcbe-f3a0-4633-98a7-7c3f90490a7f

CLASS PROJECT CODE FILES: ZIP FILE CONTENTS PT 1



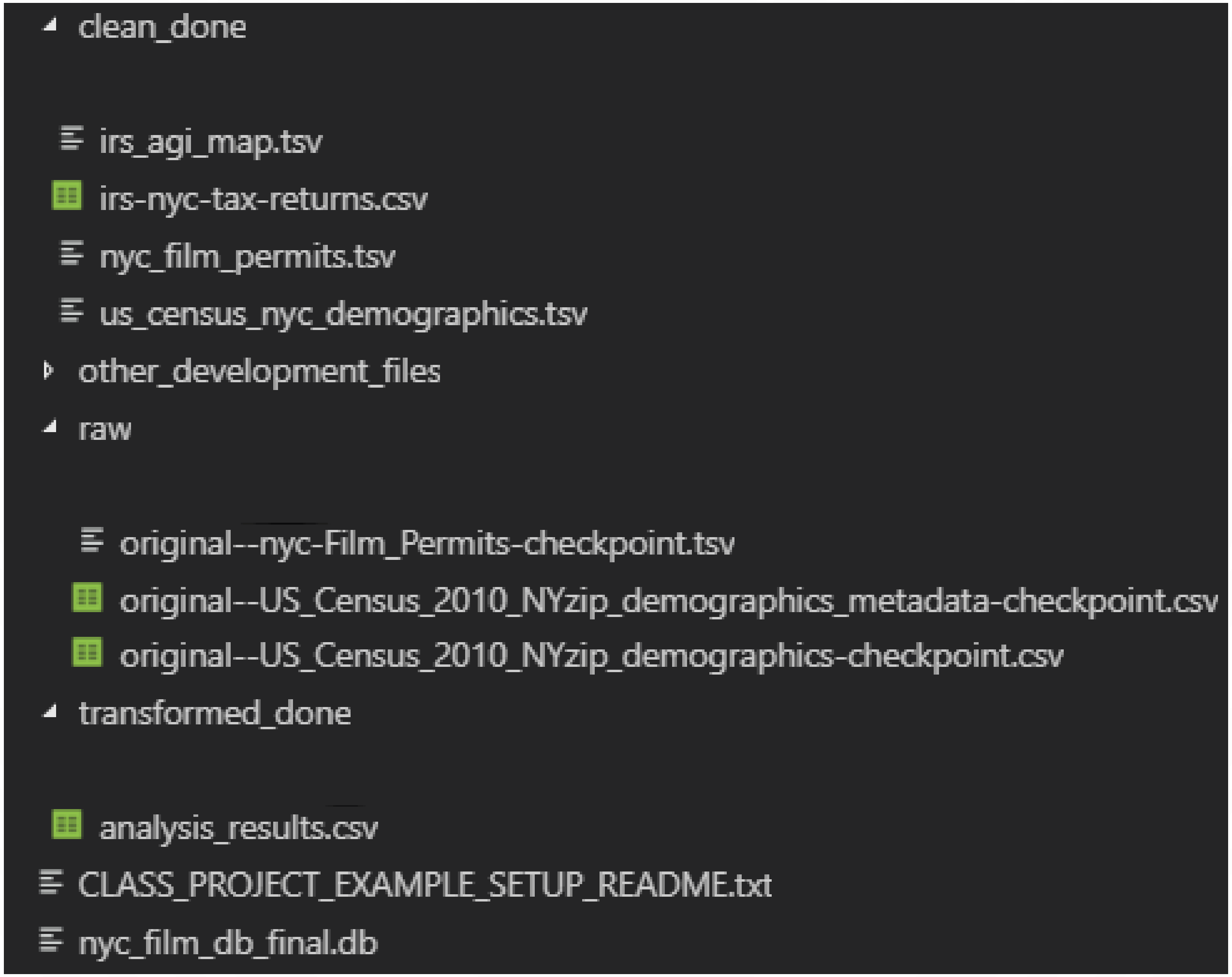
```
project_files
  !data_pipeline_code


    2-clean
    3-transform
    4-activate
  clean_done
  other_development_files
  raw
  transformed_done
  CLASS_PROJECT_EXAMPLE_SETUP_README.txt
  nyc_film_db_final.db
```

CLASS PROJECT CODE FILES: ZIP FILE CONTENTS PT 2



```
project_files
  !data_pipeline_code
    [redacted]
    2-clean
      [redacted]
      2-clean-Census_Data.py
      2-clean-Film_Permits.py
      2-clean-IRS_Data_by_Zip_Code.py
    3-transform
      [redacted]
      3-transform-create_nyc_db_tables.sql
      3-transform-select_all-nyc-vs-weighted-filmed-nyc_demographics.sql
    4-activate
      [redacted]
      4-activate_data.py
```

CLASS PROJECT CODE FILES: ZIP FILE CONTENTS PT 3



```
◢ clean_done

   irs_agi_map.tsv
   irs-nyc-tax-returns.csv
   nyc_film_permits.tsv
   us_census_nyc_demographics.tsv
▷ other_development_files
◢ raw

     original--nyc-Film_Permits-checkpoint.tsv
     original--US_Census_2010_NYzip_demographics_metadata-checkpoint.csv
     original--US_Census_2010_NYzip_demographics-checkpoint.csv
◢ transformed_done

   analysis_results.csv
 CLASS_PROJECT_EXAMPLE_SETUP_README.txt
 nyc_film_db_final.db
```

CLASS PROJECT CODE FILES: README.TXT FILE

- Example README.TXT file on NYU Classes: https://newclasses.nyu.edu/access/content/group/bf237b8c-f90d-4880-aea4-59ac6b5300e9/Group%20Project%20Resources/CLASS_PROJECT_EXAMPLE_SETUP_README.txt

CLASS PROJECT CODE FILES: README.TXT FILE CONTENTS, PT 1

1. Raw files and source mapping:

Raw files uploaded to Google Drive here: https://drive.google.com/open?id=14RWzpX0DQ7pTgdb_eUadiG6UBmP0sTZ-

You must download zip file, unzip, and copy those files into the "raw" directory

Sources for each raw file:
Neighborhood Film Permits (NYC.gov)
'original--nyc-Film_Permits.tsv' ==> https://data.cityofnewyork.us/City-Government/Film-Permits/tg4x-b46p

Neighborhood resident income data (IRS)
'original--irs-2012zpallagi.csv', 'original--irs-2013zpallagi.csv', 'original--irs-2014zpallagi.csv', 'original--irs-2015zpallagi.csv', 'original--irs-2016zpallagi.csv' ==> https://www.irs.gov/statistics/soi-tax-stats-individual-income-tax-statistics-zip-code-data-soi

(Note: the file "irs_agi_map.tsv" was created manually by reading IRS documentation.)

Neighborhood resident demographics (US Census 2010)
'original--US_Census_2010_NYzip_demographics.csv' ==> https://factfinder.census.gov/faces/affhelp/jsf/pages/metadata.xhtml?lang=en&type=dataset&id=dataset.en.DEC_10_SF1

NOTE: Used Census search to create an NYC-only data set of this file ==> https://factfinder.census.gov/faces/nav/jsf/pages/searchresults.xhtml?refresh=t

CLASS PROJECT CODE FILES: README.TXT FILE CONTENTS, PT 2

2. Run each of the Python scripts at "!data-pipeline_code/2-clean/2-clean*.py" files to clean the raw files and create the "clean" files in the clean_done folder.

3A. Launch SQLiteStudio. Create a database named "nyc_film_db_final.db" and place it in the project folder (it should be found in the same folder as "!data_pipeline_code","clean_done","raw",etc). Use the contents of the "!data-pipeline_code/3-transform/3-transform-create_nyc_db_tables.sql" file to create the SQLite database using SQLiteStudio. Then import the data for each table from "clean_done" using SQLiteStudio using the table to data mapping shown below:

irs_agi_map ==> 'irs_agi_map.tsv'

irs_nyc_tax_returns ==> 'irs-nyc-tax-returns.csv'

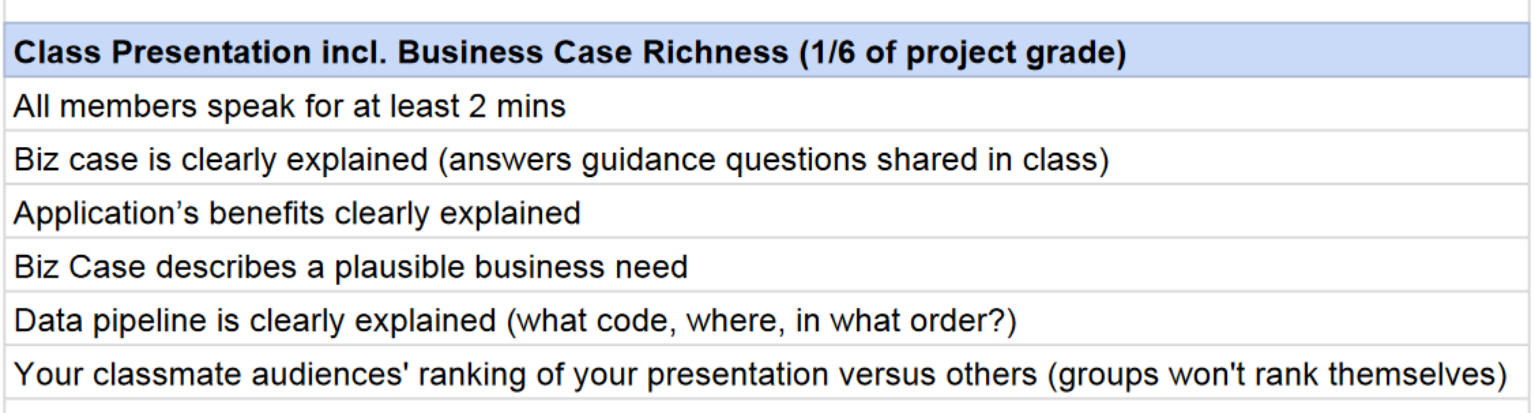
nyc_film_permits ==> 'nyc_film_permits.tsv'

nyc_census_data ==> 'us_census_nyc_demographics.tsv'

4. Run the Python script at "!data-pipeline_code/4-activate/4-activate_data.py". The nyc_film_db_final.db database must be in the main project folder to work correctly. NOTE: this Python script uses the SQL code found at "!data_pipeline_code/3-transform/3-transform-select_all-nyc-vs-weighted-filmed-nyc_demographics.sql" to create an analysis of the data comparing all NYC demographics to Filmed NYC demographics, and writes the results to a new file.

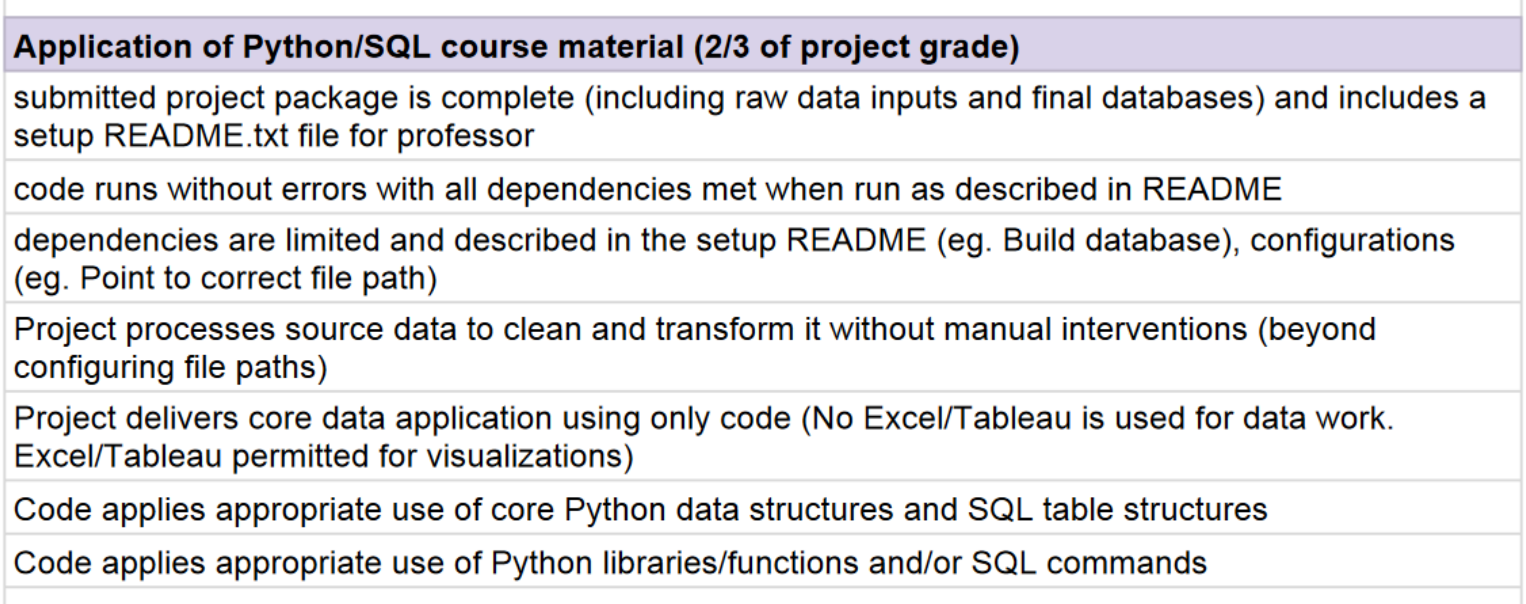
5. If desired, use Excel, Tableau or another tool to visualize the data outputted by step #4.

INSTRUCTOR GRADING CONSIDERATIONS: PRESENTATION (5 OF 30 POINTS)



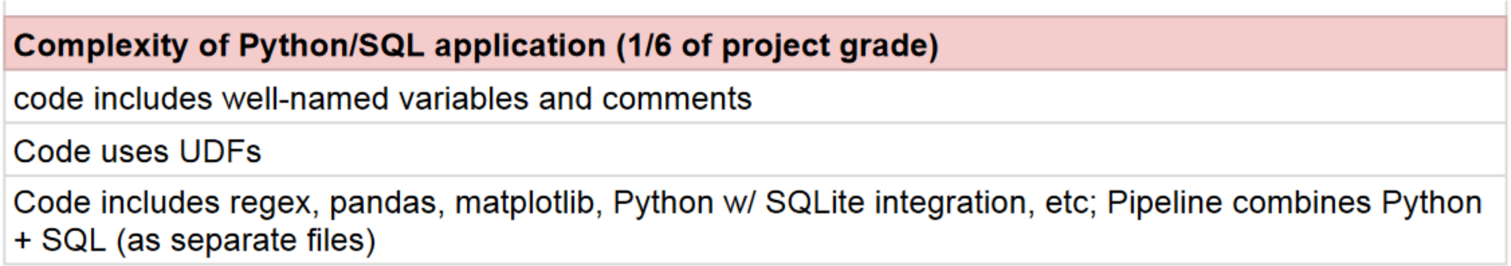
Class Presentation incl. Business Case Richness (1/6 of project grade)
All members speak for at least 2 mins
Biz case is clearly explained (answers guidance questions shared in class)
Application's benefits clearly explained
Biz Case describes a plausible business need
Data pipeline is clearly explained (what code, where, in what order?)
Your classmate audiences' ranking of your presentation versus others (groups won't rank themselves)

INSTRUCTOR GRADING CONSIDERATIONS: CORE MATERIAL (20 OF 30 POINTS)



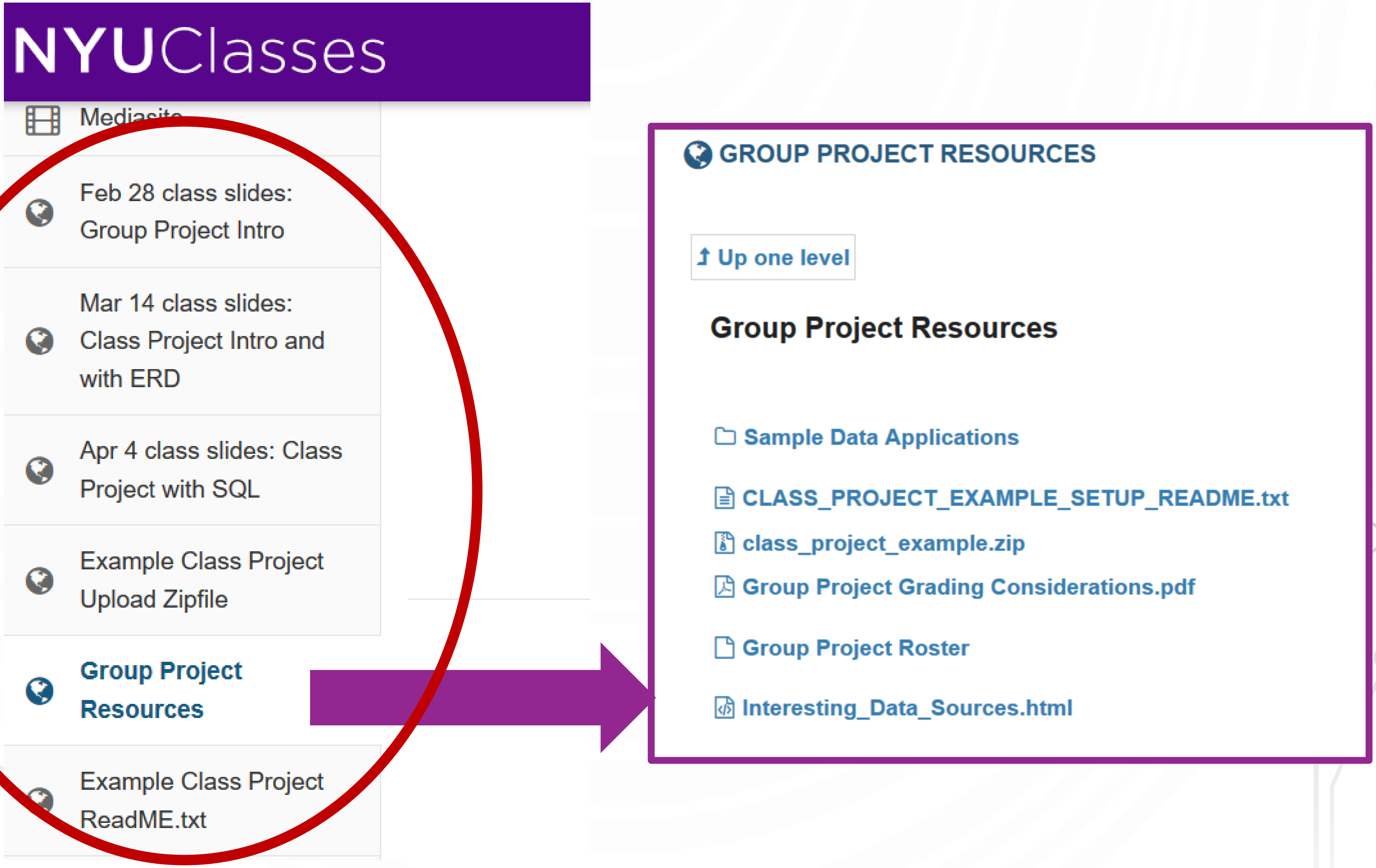
Application of Python/SQL course material (2/3 of project grade)
submitted project package is complete (including raw data inputs and final databases) and includes a setup README.txt file for professor
code runs without errors with all dependencies met when run as described in README
dependencies are limited and described in the setup README (eg. Build database), configurations (eg. Point to correct file path)
Project processes source data to clean and transform it without manual interventions (beyond configuring file paths)
Project delivers core data application using only code (No Excel/Tableau is used for data work. Excel/Tableau permitted for visualizations)
Code applies appropriate use of core Python data structures and SQL table structures
Code applies appropriate use of Python libraries/functions and/or SQL commands

INSTRUCTOR GRADING CONSIDERATIONS: CODE COMPLEXITY (5 OF 30 POINTS)



Complexity of Python/SQL application (1/6 of project grade)
code includes well-named variables and comments
Code uses UDFs
Code includes regex, pandas, matplotlib, Python w/ SQLite integration, etc; Pipeline combines Python + SQL (as separate files)

GROUP PROJECT RESOURCES QUICK LINKS



NYUClasses

Mediasite

Feb 28 class slides: Group Project Intro

Mar 14 class slides: Class Project Intro and with ERD

Apr 4 class slides: Class Project with SQL

Example Class Project Upload Zipfile

Group Project Resources

Example Class Project ReadME.txt

GROUP PROJECT RESOURCES

Up one level

Group Project Resources

- Sample Data Applications
- CLASS_PROJECT_EXAMPLE_SETUP_README.txt
- class_project_example.zip
- Group Project Grading Considerations.pdf
- Group Project Roster
- Interesting_Data_Sources.html

STUDENT SCORING: WHY?

As described in the class's syllabus:

- Group Project Grade:
  - a very small portion of your group project grade is based on peer assessment
- Individual Grade:
  - 10% of your overall grade is based on peer assessment of your contributions to the group project

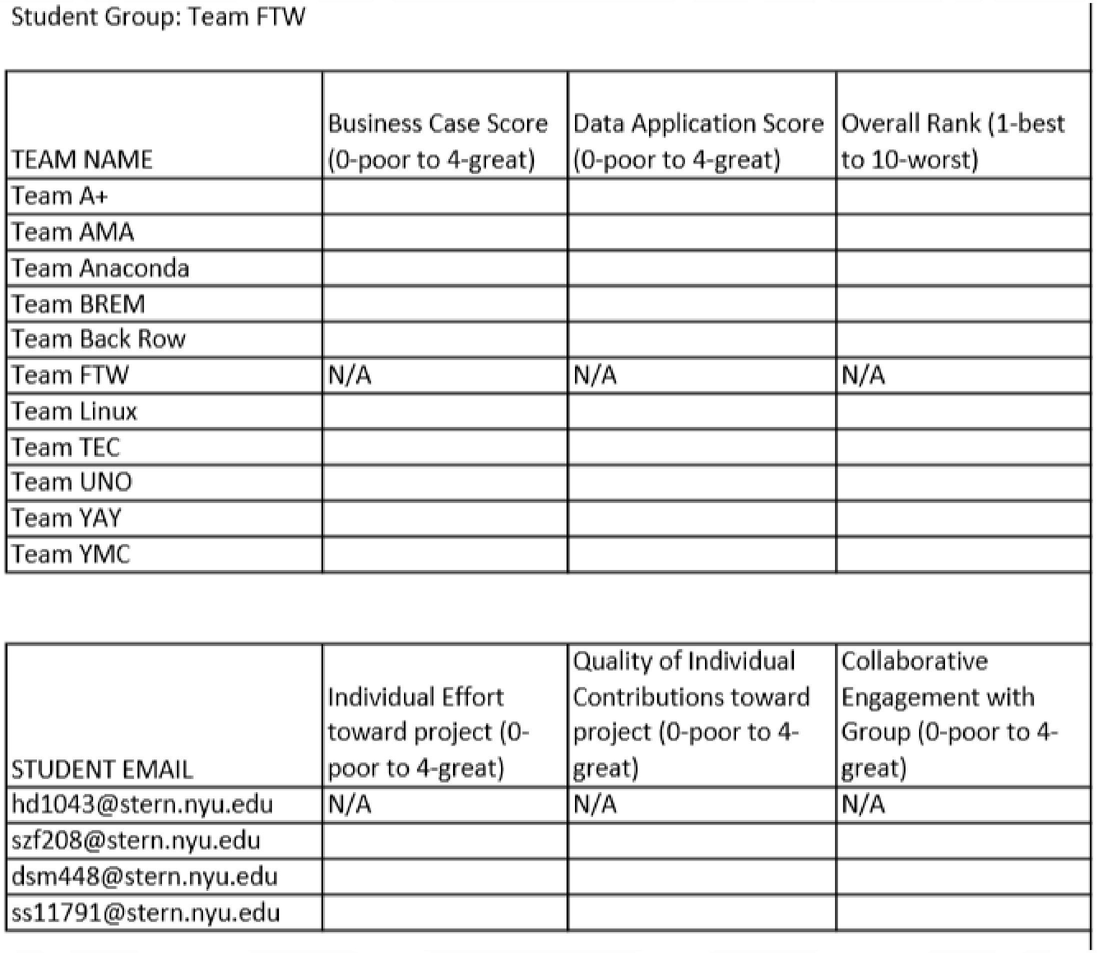
STUDENT SCORING: WHAT?

On May 9 – each of you will be handed an individualized worksheet to:

1. Record your scoring of each group presentation, and rank them all overall
2. Record your scoring of each team member’s contributions to the project

You will complete this sheet during class and hand back to Prof. Collin and Ajinkya at end of the class.

STUDENT SCORING: EXAMPLE?



Student Group: Team FTW

TEAM NAME	Business Case Score (0-poor to 4-great)	Data Application Score (0-poor to 4-great)	Overall Rank (1-best to 10-worst)
Team A+			
Team AMA			
Team Anaconda			
Team BREM			
Team Back Row			
Team FTW	N/A	N/A	N/A
Team Linux			
Team TEC			
Team UNO			
Team YAY			
Team YMC			

STUDENT EMAIL	Individual Effort toward project (0-poor to 4-great)	Quality of Individual Contributions toward project (0-poor to 4-great)	Collaborative Engagement with Group (0-poor to 4-great)
hd1043@stern.nyu.edu	N/A	N/A	N/A
szf208@stern.nyu.edu			
dsm448@stern.nyu.edu			
ss11791@stern.nyu.edu			

STUDENT SCORING: RANKING OTHER GROUPS

Scoring Sheet asks you to rank other groups on the quality of their project and presentation.

Here are some key areas to consider:

1. Business Case
   1. Would you sponsor/fund this project if you were the target user?
   2. Do you understand the user and their need for the data app?
2. Data Application
   1. Is the end result clear?
   2. Does it satisfy the stated user's needs?
   3. Do you understand the data source and data pipeline processing stages used to create the application?

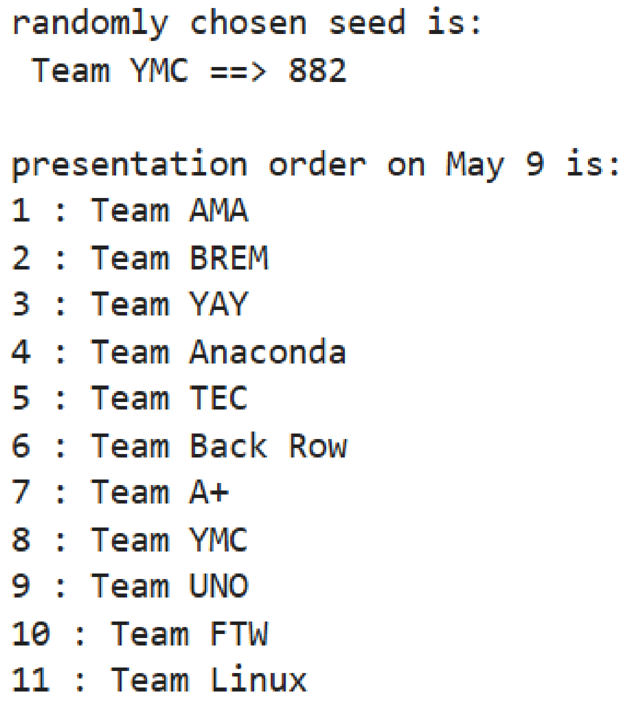
STUDENT SCORING:
RATING YOUR TEAMMATES

Scoring Sheet asks you to rate your peers on their effort, quality and collaboration with others.

Here are some key work areas of the project's delivery to consider in rating your teammates' contributions:

1. Project Research & Analysis
2. Producing Code
3. Producing Presentation
4. Coordinating Group Activities

PRESENTATION ORDER ON MAY 9: PYTHON RANDOM SELECTION



```
randomly chosen seed is:
 Team YMC ==> 882

presentation order on May 9 is:
1 : Team AMA
2 : Team BREM
3 : Team YAY
4 : Team Anaconda
5 : Team TEC
6 : Team Back Row
7 : Team A+
8 : Team YMC
9 : Team UNO
10 : Team FTW
11 : Team Linux
```