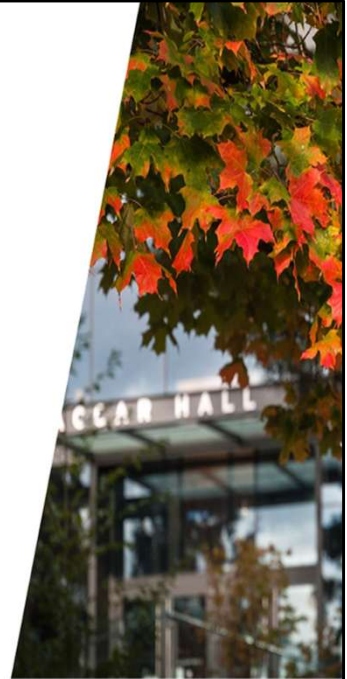


INNOVATIVE DATA SOURCES AND TOOLS: GENERATIVE AI

Data Analytics for Professional Accountants
(ACCTG 522)

Class 7 | MPAcc Class of 2026



Welcome!

Class Agenda

- **Review**
- Brief Overview on Generative AI
- Labs:
 - Pure Oils Part 1 (Chat GPT and Python)
- Conclusion and look ahead
 - Pure Oils Part 2 (Alteryx)

The Analytical Mindset

The Analytical Mindset is defined as the ability to work with data to apply appropriate analytics and interpret and share insight with stakeholders.

Why do we care about Extract, Transform & Load (ETL)?

The Extract, Transform & Load or ETL Process is the **first task** required to complete the analytical process.

- In most cases, data required for the analytical process is stored in a database and must be **extracted** from that database and **loaded** into analytics software.
- More challenging **extraction** (typically **unstructured**) require more **advanced** tools.

Regular Expressions

More advanced ETL Processes include using **Regular Expressions** (RegEx) to extract patterns from any set of characters (numbers, string, punctuation).

- Regular Expressions allowed us to split the **email header** text into useable fields
- And allowed us to undertake analysis of **unstructured** text (i.e., the subject & email body).

Welcome!

Class Agenda

- Review
- **Brief Overview on Generative AI**
- Labs:
 - Pure Oils Part 1 (Chat GPT and Python)
- Conclusion and look ahead
 - Pure Oils Part 2 (Alteryx)

Generative AI capabilities

Generative AI can produce a wide range of outputs depending on the specific application and type of data that is needed. Here are some common output types that are applicable to business

Text

Prompt: Explain the business impact of Generative AI in 50 words

Generative AI is an opportunity to improve business efficiency by automating repetitive tasks, generating new ideas, and enhancing decision-making through data analysis. It can also provide personalized recommendations for customers and streamline new product development.

Video

Prompt: Create a video of a teddy bear painting a portrait



Image

Prompt: A bowl of soup that is a portal to another dimension as digital art



Code

Prompt: In python, code a program that predicts the likelihood of customer conversion



Audio

Generative AI-powered customer service agents

- Phone / Voice
- Multi-Lingual
- Multiple Tasks
- Empathy / Humor
- Ensures Resolution
- Systems Integration

3D

Prompt: A beautiful dress made out of garbage bags, on a mannequin. Studio lighting, high quality, high resolution.



ChatGPT 5 ▾

Explain the business impact of Generative AI in 50 words

Generative AI transforms business by automating content creation, enhancing decision-making, and enabling personalized customer experiences. It accelerates innovation, reduces costs, and improves productivity across functions such as marketing, finance, and operations. However, it introduces ethical, legal, and data governance challenges that require careful oversight to sustain trust and competitive advantage.

📄 👍 💬 ↗️ ↺ ⋮

Write me a short python def that randomly selects string data from lists and numerical data from ranges, and a way to produce a field that concatenates string and numbers, the def should include which lists to draw from, able to accommodate additional lists and be able to vary the number of records. Start the def as def csv_generator (num_obs=, ranges_and_lists=[]). num_obs is the numbers of records to be randomly produced. ranges_and_lists=[] is a list variable that lists in order the fields to be produced. After producing the data this output should be saved to a csv file.

```
python Copy code

import csv
import random
from itertools import product

def csv_generator(num_obs=10, ranges_and_lists=[]):
    """
    Generates a CSV file with randomly selected data from lists and numeric ranges.

    Parameters:
        num_obs (int): Number of observations (rows) to generate.
        ranges_and_lists (list): List of tuples specifying field name and source.
            Each element should be in one of the following forms:
                ('field_name', [list_of_strings])
                ('field_name', range(start, end))
                ('field_name', ('concat', ['field1', 'field2'])) # combines existing fields

    Example:
        csv_generator(5, [
            ('name', ['Alice', 'Bob', 'Charlie']),
            ('city', ['Seattle', 'Portland', 'Austin']),
            ('age', range(20, 41)),
            ('id', ('concat', ['name', 'age']))
        ])
    """
```

```
# Initialize output data
data = []

for _ in range(num_obs):
    record = {}

    # Generate basic fields
    for field, source in ranges_and_lists:
        if isinstance(source, range): # numeric range
            record[field] = random.choice(list(source))
        elif isinstance(source, list): # list of strings
            record[field] = random.choice(source)
        elif isinstance(source, tuple) and source[0] == 'concat':
            # concatenation field (will process later)
            record[field] = source
        else:
            raise ValueError(f"Unsupported data type for field: {field}")

    # Handle concatenation fields after all base fields exist
    for field, source in ranges_and_lists:
        if isinstance(source, tuple) and source[0] == 'concat':
            concat_fields = source[1]
            record[field] = ''.join(str(record[f]) for f in concat_fields)

    data.append(record)

# Write to CSV file
filename = "generated_data.csv"
with open(filename, "w", newline='', encoding='utf-8') as f:
    writer = csv.DictWriter(f, fieldnames=[f[0] for f in ranges_and_lists])
    writer.writeheader()
    writer.writerows(data)

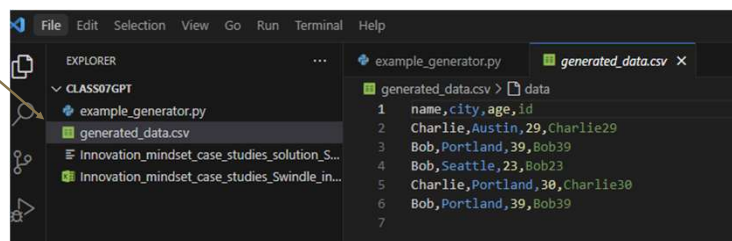
print(f"{num_obs} records written to {filename}")
```

On the remote labs we can use visual studio code, import a few libraries and make an update at the end of the file to call the def



```
csv_generator(5, [  
    ('name', ['Alice', 'Bob', 'Charlie']),  
    ('city', ['Seattle', 'Portland', 'Austin']),  
    ('age', range(20, 41)),  
    ('id', ('concat', ['name', 'age']))  
])
```

And our data is generated:



```
File Edit Selection View Go Run Terminal Help  
EXPLORER  
CLASS07GPT  
example_generator.py  
generated_data.csv  
Innovation_mindset_case_studies_solution_S...  
Innovation_mindset_case_studies_Swindle_in...  
example_generator.py generated_data.csv x  
generated_data.csv > data  
1 name,city,age,id  
2 Charlie,Austin,29,Charlie29  
3 Bob,Portland,39,Bob39  
4 Bob,Seattle,23,Bob23  
5 Charlie,Portland,30,Charlie30  
6 Bob,Portland,39,Bob39  
7
```

Core Capabilities: Doc & Data Intelligence

Doc Intelligence

- Automated understanding and summarization of documents
- Quick extraction of important points and data from any text
- Comparison of document contents to identify key differences
- Accurate language translation for diverse document types
- Trend and pattern detection within large volumes of text

Data Intelligence

- Connect to any data source, integrating seamlessly with AI.
- Utilize AI to uncover deep insights from complex datasets.
- Empower finance to do real-time decision-making with dynamic, AI-driven data analysis.
- Leverage predictive analytics to forecast trends and guide strategic planning.



AI Engagement Models: Prompt vs. Pre-Prompt

Prompt-Response

- Use an AI assistant for direct, real-time interaction to receive immediate feedback and assistance.
- Adaptive AI responses evolve from each user interaction, enhancing accuracy and relevance.
- Enable swift problem-solving and decision support, improving overall efficiency and user experience.

Pre-Prompt

- AI operates in the background based on predefined prompts and conditions.
- Conducts continuous background processing for data analysis, info generation, and monitoring.
- Automates tasks to streamline operations and prepare actionable insights without user intervention.

Welcome!

Class Agenda

- Review
- Brief Overview on Generative AI
- **Labs:**
 - **Pure Oils Part 1 (Chat GPT and Python)**
- Conclusion and look ahead
 - Pure Oils Part 2 (Alteryx)

Pure Oils Case Overview

Founder: Skylar Swindler

Founded: 2021

Headquarters: Recife, Brazil

Business Claim: Produces 100% natural essential oils from fallen rainforest leaves — marketed as eco-friendly and possessing “near magical” properties.

Reality: Entire operation is fraudulent — bottles filled with water and other liquids.

Employees: 110 total

- Majority hired in the past year
- Promoted as native Brazilians “lifted out of poverty”
- Example employee story: *Fernanda Silva Pazos* featured in marketing
- Skylar claims large-scale employment of local workers

Lab Part 1

Help Skylar Swindler at Pure Oils provide evidence of his company's *legitimate* existence:

- Single Prompt ("One Shot" approach)
 - Assessment of Red-flags
- Secondary prompt for refinement
- Use as a Coding Assistant
- ***Generate data for 110 current employees in csv format***



The Class Prompt

Generate me a dataset of 110 employees for a fictitious company called Pure Oils and return it as a csv. The fields are [UniqueID,LastName,FirstName,JobTitle,Department,HireDate,LastPromotionDate,TerminationDate,AnnualSalary]. The first employee is Called Skylar Swindler, his job title is CEO. He was hired on 1/1/2021.

Single prompt / Class Prompt

Generate me a dataset of 110 employees for a fictitious company called Pure Oils and return it as a csv. The fields are [UniqueID, LastName, FirstName, JobTitle, Department, HireDate, LastPromotionDate, TerminationDate, AnnualSalary]. The first employee is Called Skylar Swindler, his job title is CEO. He was hired on 1/1/2021.

Here's your generated dataset for **Pure Oils** — 110 employees including CEO Skylar Swindler.

You can download the CSV file here:

📄 [pure_oils_employees.csv](#)

The dataset includes fields for UniqueID, LastName, FirstName, JobTitle, Department, HireDate, LastPromotionDate, TerminationDate, and AnnualSalary.

I am highly confident (≈99%) the data generation meets your specifications. ↻

📄 📄 📄 📄 📄 ...

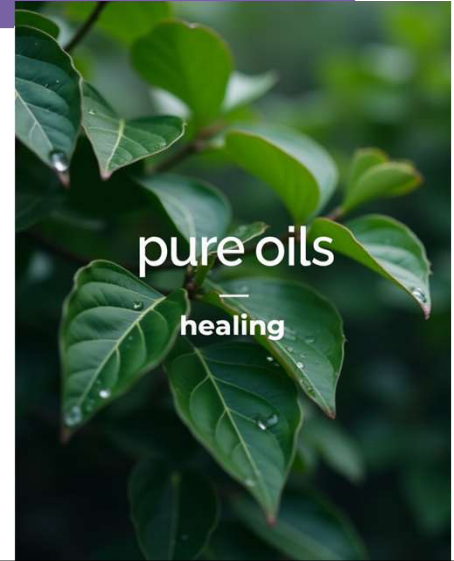
Example output

UniqueID	LastName	FirstName	JobTitle	Department	HireDate	LastPromotionDate	TerminationDate	AnnualSalary
PO-100001	Swindler	Skylar	CEO	Executive		1/1/2021		250000
PO-100002	Patel	Peyton	Assistant	Marketing		12/8/2024	7/23/2025	147817
PO-100003	Young	Alex	Technician	IT		8/21/2023		76073
PO-100004	Baker	Sky	Specialist	Sales		2/25/2023		86402
PO-100005	Allen	Jordan	Technician	Operations		5/22/2024	1/31/2025	70100
PO-100006	Garcia	Jesse	Engineer	Sales		6/28/2022	7/21/2025	62712
PO-100007	Baker	Drew	Technician	IT		10/23/2022	2/13/2024	98189
PO-100008	Wright	Dakota	Supervisor	Finance		8/26/2024	9/9/2024	145092
PO-100009	Rivera	Avery	Specialist	Quality		2/6/2022	2/13/2025	113581
PO-100010	Baker	Morgan	Engineer	Logistics		6/13/2021	12/30/2021	79331

Follow-up Prompt

Keep your prompt open and prompt fixes of the data.

Look to see how much improvement you achieve.



Coding assistant



Have Chat GPT help build you customizable python code to generate the fake dataset.

Tips: use it's knowledge of various features of the demographics you may not be aware of.

To: Asher-Curtis⁴
From: {Student-Name}, Generative-AI Analyst⁴
Subject: Assessment of Chat-GPT as a Data-Generator⁴
Date: {Insert-date}⁴

Summary-of-Findings (1 paragraph)⁴

Summarize the key progression from the single prompt to coding assistant.⁴

First-Prompt-Analysis⁴

Using the following class-prompt:⁴

*Generate me a dataset of 110 employees for a fictitious company called Pure Oils and return it as a csv. The fields are: [UniqueID, LastName, FirstName, Job Title, Department, Hire Date, Last Promotion Date, Termination Date, Annual Salary]. The first employee is Called Skylar Swindler, his job title is CEO. He was hired on 1/1/2021.*⁴

I found the following red-flags:⁴

- → Use a list with bullets here.⁴

Follow-up-Prompt-Improvement⁴

After using the class-prompt, I prompted the LLM with the following:⁴

*Update the dataset to fix the following....*⁴

⁴

Summarize here the improvement in "reality" of the fake data output and what red flags remained, or any new indicators of AI generation of the data.⁴

Assessment of code generating ability⁴

Write a short paragraph here to document potential improvement in the flexibility and potential output from using the GPT as a coding assistant. Only make minor modifications to the code and highlight these in bold.⁴

Attachments⁴

Attachment 1 (First-Prompt csv output uploaded to Canvas)⁴

Attachment 2 (Follow-up Prompt uploaded to Canvas)⁴

Attachment 3: GPT-Assisted Python Code (paste into this document).⁴

Welcome!

Class Agenda

- Review
- Brief Overview on Generative AI
- Labs:
 - Pure Oils Part 1 (Chat GPT and Python)
- **Conclusion and look ahead**
 - **Pure Oils Part 2 (Alteryx)**

Lab Part 2 – In future ACCTG521

Assess the output files from classmates:

- Alteryx workflows:
 - Generate a diagnostic for a fake/AI generated csv file
 - Look for red flags, set up tests, use Alteryx to help find and document the red flags



Thank you!

