Week 1: Data Cleaning and Feature Engineering

Dataset Overview --

The Excelerate Student Engagement Dataset captures detailed information about learners' participation, including signup timestamps, opportunity details, and demographic data such as gender and date of birth. It also includes categorical and status indicators to track the type and progress of each opportunity. This dataset provides a comprehensive foundation for analyzing student engagement patterns and behaviors.

The dataset includes the following key features --

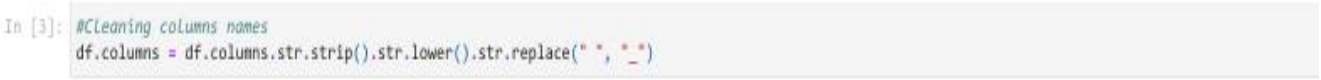
- **Learner SignUp DateTime:** Timestamp of when the user signed up.
- **Opportunity Id:** Unique identifier for each opportunity.
- **Opportunity Name:** Name of the opportunity.
- **Opportunity Category:** Category of the opportunity (e.g., Course, Event).
- **Opportunity End Date:** Date when the opportunity ends.
- **First Name:** User's first name.
- **Date of Birth:** User's date of birth.
- **Gender:** User's gender.
- **Country:** User's country of residence.
- **Institution Name:** Name of the institution the user is associated with.
- **Current/Intended Major:** User's current or intended major.
- **Entry created at:** Timestamp of when the entry was created.
- **Status Description:** Current status of the user.
- **Status Code:** Code representing the status.
- **Apply Date:** Date when the user applied for the opportunity.
- **Opportunity Start Date:** Date when the opportunity starts

Part 1: Data Cleaning

The first and foundational step in our Week 1 assignment involved cleaning the raw dataset provided. This process was crucial to ensure that our subsequent analysis and model training were based on consistent, reliable, and meaningful data. Here's a breakdown of the major cleaning procedures performed:

1. Standardizing Column Names --

To maintain consistency and improve code readability, all column names were reformatted. We removed whitespace, converted all names to lowercase, and replaced spaces with underscores. This uniform naming convention made the dataset easier to work with during processing.



2. Handling Date Formats --

A significant portion of the dataset involved time-based data such as sign-up dates, birthdates, and course start/end dates. These were originally stored as strings. We converted all relevant columns to proper datetime formats. This allowed for accurate computations such as age calculation and duration analysis.

In [4]:
```
#Converting date columns to datetime.
date_cols = [
    'learner_signup_datetime', 'opportunity_end_date', 'date_of_birth',
    'entry_created_at', 'apply_date', 'opportunity_start_date'
]
for col in date_cols:
    df[col] = pd.to_datetime(df[col], errors='coerce')
```

3. Cleaning Gender Values --

The gender column had inconsistent entries like "F", "fem", "Male ". These values were normalized to either male or female, ensuring consistency for demographic analysis.

4. Removing Duplicates --

We checked for and eliminated duplicate records to prevent redundant entries from skewing any analysis. This step helped us maintain the integrity of the dataset.

In [5]:
```
#Normalizing gender values
df['gender'] = df['gender'].str.lower()
df['gender'] = df['gender'].replace({
    'f': 'female', 'm': 'male', 'fem': 'female', 'mal': 'male',
    'female ': 'female', 'male ': 'male'
})
```

In [6]:
```
#Indentifying and Removi duplicates
duplicates = df.duplicated()
duplicates

df = df.drop_duplicates()
```

5. Deriving New Fields --

Several new columns were added to extract more insight from existing data:

- **Age** was computed based on date of birth.
- **Sign-up date** and **month** were extracted from the timestamp.

--- These transformations helped enrich the dataset with features more suitable for exploratory analysis and modeling.

In [7]:
```
#Adding age and date features
df['age'] = pd.Timestamp.today().year - df['date_of_birth'].dt.year
df['signup_date'] = df['learner_signup_datetime'].dt.date
df['signup_month'] = df['learner_signup_datetime'].dt.to_period("M")
```

6. Index Reset and Data Export --

After cleaning, we reset the dataframe index for a tidy structure and exported the cleaned version to a CSV file (cleaned_slu_data.csv) for future use.

In [8]:
```
#fixing types and reset index
df = df.reset_index(drop=True)
```

In [9]:
```
#saving data to csv
df.to_csv("cleaned_slu_data.csv", index=False)
```

Final Cleaned Dataset:
Rows: 8,558
Columns: 19

```
In [10]: print("Cleaned Data Preview:")
         print(df.head())
         print(f"Cleaned dataset shape: {df.shape}")
```

```
Cleaned Data Preview:
  learner_signup_datetime                opportunity_id  \
0     2023-06-14 12:30:35  00000000-0GN2-A0AY-7XK8-C5FZPP
1     2023-01-05 05:29:16  00000000-0GN2-A0AY-7XK8-C5FZPP
2     2023-09-04 20:35:08  00000000-0GN2-A0AY-7XK8-C5FZPP
3     2023-08-29 05:20:03  00000000-0GN2-A0AY-7XK8-C5FZPP
4     2023-06-01 15:26:36  00000000-0GN2-A0AY-7XK8-C5FZPP

                              opportunity_name opportunity_category  \
0  Career Essentials: Getting Started with Your P...               Course
1  Career Essentials: Getting Started with Your P...               Course
2  Career Essentials: Getting Started with Your P...               Course
3  Career Essentials: Getting Started with Your P...               Course
4  Career Essentials: Getting Started with Your P...               Course

  opportunity_end_date        first_name date_of_birth  gender        country  \
0  2024-06-29 18:52:39             Faria    2001-12-01  female       Pakistan
1  2024-06-29 18:52:39          Poojitha    2000-08-16  female          India
2  2024-06-29 18:52:39          Emmanuel    2002-01-27    male  United States
3  2024-06-29 18:52:39  Amrutha Varshini    1999-01-11  female  United States
4  2024-06-29 18:52:39    Vinay Varshith    2000-04-19    male  United States

                   institution_name current/intended_major  \
0                             Nwihs              Radiology
1                        SAINT LOUIS    Information Systems
2  Illinois Institute of Technology       Computer Science
3            Saint Louis University    Information Systems
4            Saint Louis University       Computer Science

      entry_created_at status_description  status_code          apply_date  \
0 2024-11-03 12:01:41            Started         1080 2023-06-14 12:36:09
1 2024-11-03 12:01:41            Started         1080 2023-01-05 06:08:21
2 2024-11-03 12:01:41            Started         1080                 NaT
3 2024-11-03 12:01:41     Team Allocated         1070 2023-09-10 22:02:42
4 2024-11-03 12:01:41            Started         1080 2023-06-01 15:40:10

  opportunity_start_date  age signup_date signup_month
0    2022-03-11 18:30:39   24  2023-06-14      2023-06
1    2022-03-11 18:30:39   25  2023-01-05      2023-01
2    2022-03-11 18:30:39   23  2023-09-04      2023-09
3    2022-03-11 18:30:39   26  2023-08-29      2023-08
4    2022-03-11 18:30:39   25  2023-06-01      2023-06
Cleaned dataset shape: (8558, 19)
```

Part 2: Feature Engineering

Once the dataset was cleaned, the next phase involved engineering additional features that would provide deeper insights and assist in model performance.
This was a crucial step to enhance the predictive power of the data.

1. Temporal Features --

From the signup timestamp, we derived:

- **Signup hour** – to analyze the time of day users registered.
- **Signup weekday** – to detect weekly patterns.
- **Signup quarter** – to capture seasonal trends.

These features help in identifying behavioral patterns around user engagement.

```
In [11]: # 1. Signup Time Features
         df['signup_hour'] = df['learner_signup_datetime'].dt.hour
         df['signup_weekday'] = df['learner_signup_datetime'].dt.day_name()
         df['signup_quarter'] = df['learner_signup_datetime'].dt.quarter
```

2. Age Bucketing --

We categorized users into age brackets such as '18-24', '25-34', and more, using logical bins. This grouping helps in demographic segmentation and analysis.

```
In [12]: # 2. Age Group Bucketing
         bins = [0, 17, 24, 34, 44, 54, 64, 100]
         labels = ['<18', '18-24', '25-34', '35-44', '45-54', '55-64', '65+']
         df['age_group'] = pd.cut(df['age'], bins=bins, labels=labels)
```

3. Opportunity Metrics --

- **Completion time (in days)**: Calculated as the difference between signup date and opportunity end date.
- **Opportunity duration**: Measured from the start to end date of the opportunity.
- **Completion status**: A new boolean feature, is_completed, was created by checking whether the user's status included the word "completed".

```
In [13]: # 3. Completion Time (days between signup and opportunity end)
         df['completion_time_days'] = (df['opportunity_end_date'] - df['learner_signup_datetime']).dt.days

In [14]: # 4. Opportunity Duration
         df['opportunity_duration'] = (df['opportunity_end_date'] - df['opportunity_start_date']).dt.days

In [15]: # 5. Completion Status (boolean)
         df['is_completed'] = df['status_description'].str.lower().str.contains('completed')
```

These engineered variables are particularly useful for evaluating course engagement and performance trends.

4. Handling Missing Values --

We addressed missing data in the new features by using placeholders like -1, especially for metrics like opportunity duration where data might be incomplete. This prevented the loss of valuable entries during model training.

```
In [16]: # 6. Fill missing values
         df['completion_time_days'] = df['completion_time_days'].fillna(-1)
         df['opportunity_duration'] = df['opportunity_duration'].fillna(-1)
```

5. Preview of Engineered Features --

A subset of columns was examined to verify the correctness of the feature engineering process. Key columns included:

- signup_hour
- signup_weekday
- age_group
- completion_time_days
- opportunity_duration
- is_completed

```
In [17]: #Previewing the  engineered features
         print(df[['signup_hour', 'signup_weekday', 'signup_quarter', 'age', 'age_group',
                   'completion_time_days', 'opportunity_duration', 'is_completed']].head())
```

```
   signup_hour signup_weekday  signup_quarter  age age_group  \
0         12.0      Wednesday             2.0   24     18-24
1          5.0       Thursday             1.0   25     25-34
2         20.0         Monday             3.0   23     18-24
3          5.0        Tuesday             3.0   26     25-34
4         15.0       Thursday             2.0   25     25-34

   completion_time_days  opportunity_duration  is_completed
0                 381.0                 841.0         False
1                 541.0                 841.0         False
2                 298.0                 841.0         False
3                 305.0                 841.0         False
4                 394.0                 841.0         False
```

a) Summary of Outcomes

By the end of Week 1:

- We transformed a messy dataset into a structured and meaningful form.
- Key time-based and categorical fields were normalized and enriched.
- New features were engineered to capture user behavior, engagement timing, and course dynamics.
- The refined dataset was now fully prepared for exploratory data analysis (EDA) and predictive modeling in the upcoming weeks.

b) Why This Matters

Clean, structured data is the cornerstone of any successful data science project. Without accurate preprocessing:

- Insights from EDA may be misleading.
- Machine learning models can perform poorly or exhibit bias.
- Business decisions may be based on faulty assumptions.