In [1]:
```
!python --version
```

```
Python 3.13.5
```

In [3]:
```
!pip install pandas numpy matplotlib seaborn plotly scikit-learn openpyxl
```

```
Requirement already satisfied: pandas in c:\users\shari\anaconda3\lib\site-packages
(2.2.3)
Requirement already satisfied: numpy in c:\users\shari\anaconda3\lib\site-packages
(2.1.3)
Requirement already satisfied: matplotlib in c:\users\shari\anaconda3\lib\site-packa
ges (3.10.0)
Requirement already satisfied: seaborn in c:\users\shari\anaconda3\lib\site-packages
(0.13.2)
Requirement already satisfied: plotly in c:\users\shari\anaconda3\lib\site-packages
(5.24.1)
Requirement already satisfied: scikit-learn in c:\users\shari\anaconda3\lib\site-pac
kages (1.6.1)
Requirement already satisfied: openpyxl in c:\users\shari\anaconda3\lib\site-package
s (3.1.5)
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\shari\anaconda3\li
b\site-packages (from pandas) (2.9.0.post0)
Requirement already satisfied: pytz>=2020.1 in c:\users\shari\anaconda3\lib\site-pac
kages (from pandas) (2024.1)
Requirement already satisfied: tzdata>=2022.7 in c:\users\shari\anaconda3\lib\site-p
ackages (from pandas) (2025.2)
Requirement already satisfied: contourpy>=1.0.1 in c:\users\shari\anaconda3\lib\site
-packages (from matplotlib) (1.3.1)
Requirement already satisfied: cycler>=0.10 in c:\users\shari\anaconda3\lib\site-pac
kages (from matplotlib) (0.11.0)
Requirement already satisfied: fonttools>=4.22.0 in c:\users\shari\anaconda3\lib\sit
e-packages (from matplotlib) (4.55.3)
Requirement already satisfied: kiwisolver>=1.3.1 in c:\users\shari\anaconda3\lib\sit
e-packages (from matplotlib) (1.4.8)
Requirement already satisfied: packaging>=20.0 in c:\users\shari\anaconda3\lib\site-
packages (from matplotlib) (24.2)
Requirement already satisfied: pillow>=8 in c:\users\shari\anaconda3\lib\site-packag
es (from matplotlib) (11.1.0)
Requirement already satisfied: pyparsing>=2.3.1 in c:\users\shari\anaconda3\lib\site
-packages (from matplotlib) (3.2.0)
Requirement already satisfied: tenacity>=6.2.0 in c:\users\shari\anaconda3\lib\site-
packages (from plotly) (9.0.0)
Requirement already satisfied: scipy>=1.6.0 in c:\users\shari\anaconda3\lib\site-pac
kages (from scikit-learn) (1.15.3)
Requirement already satisfied: joblib>=1.2.0 in c:\users\shari\anaconda3\lib\site-pa
ckages (from scikit-learn) (1.4.2)
Requirement already satisfied: threadpoolctl>=3.1.0 in c:\users\shari\anaconda3\lib
\site-packages (from scikit-learn) (3.5.0)
Requirement already satisfied: et-xmlfile in c:\users\shari\anaconda3\lib\site-packa
ges (from openpyxl) (1.1.0)
Requirement already satisfied: six>=1.5 in c:\users\shari\anaconda3\lib\site-package
s (from python-dateutil>=2.8.2->pandas) (1.17.0)
```

In [5]:
```
#Core libraries for data analysis and visualization
import pandas as pd
```

```
import numpy as np
import matplotlib.pyplot as pit
import seaborn as sns
import plotly.express as px
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LinearRegression
from sklearn.metrics import mean_squared_error, r2_score
```

In [7]:
```
#Load dataset
file_path = r"C:/Users/SHARI/OneDrive/Pictures/Documents/Neka/Uncommon Project Port
```

In [10]:
```
# Check column names, data types, and memory usage
data.info()

# Get descriptive statistics for numeric columns (min, max, mean, etc.)
data.describe()

# Check for missing values across all columns
data.isnull().sum()
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[10], line 2
      1 # Check column names, data types, and memory usage
----> 2 data.info()
      4 # Get descriptive statistics for numeric columns (min, max, mean, etc.)
      5 data.describe()

NameError: name 'data' is not defined
```

In [11]:
```
whos
```

```
Variable             Type        Data/Info
-----------------------------------------
LinearRegression     ABCMeta     <class 'sklearn.linear_mo<...>._base.LinearRegressi
on'>
file_path            str         C:/Users/SHARI/OneDrive/P<...>xercise_2022_Cleaned.
xlsx
mean_squared_error   function    <function mean_squared_er<...>or at 0x00000222853C7
4C0>
np                   module      <module 'numpy' from 'C:\<...>ges\\numpy\\__init__.
py'>
pd                   module      <module 'pandas' from 'C:<...>es\\pandas\\__init__.
py'>
pit                  module      <module 'matplotlib.pyplo<...>\\matplotlib\\pyplot.
py'>
px                   module      <module 'plotly.express' <...>y\\express\\__init__.
py'>
r2_score             function    <function r2_score at 0x00000222853C7CE0>
sns                  module      <module 'seaborn' from 'C<...>s\\seaborn\\__init__.
py'>
train_test_split     function    <function train_test_split at 0x0000022285409760>
```

In [12]:
```
import pandas as pd

# Load your Excel file into a pandas DataFrame
```

```
data = pd.read_excel(r"C:\Users\SHARI\OneDrive\Documents\Neka\Uncommon Project Port

#Display the first few rows to confirm successful loading
data.head()
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
Cell In[12], line 4
      1 import pandas as pd
      3 # Load your Excel file into a pandas DataFrame
----> 4 data = pd.read_excel(r"C:\Users\SHARI\OneDrive\Documents\Neka\Uncommon Proje
ct Portfolio\Data_Preparation_Exercise_2022_Cleaned.xlsx")
      6 #Display the first few rows to confirm successful loading
      7 data.head()

File ~\anaconda3\Lib\site-packages\pandas\io\excel\_base.py:495, in read_excel(io, s
heet_name, header, names, index_col, usecols, dtype, engine, converters, true_value
s, false_values, skiprows, nrows, na_values, keep_default_na, na_filter, verbose, pa
rse_dates, date_parser, date_format, thousands, decimal, comment, skipfooter, storag
e_options, dtype_backend, engine_kwargs)
    493 if not isinstance(io, ExcelFile):
    494     should_close = True
--> 495     io = ExcelFile(
    496         io,
    497         storage_options=storage_options,
    498         engine=engine,
    499         engine_kwargs=engine_kwargs,
    500     )
    501 elif engine and engine != io.engine:
    502     raise ValueError(
    503         "Engine should not be specified when passing "
    504         "an ExcelFile - ExcelFile already has the engine set"
    505     )

File ~\anaconda3\Lib\site-packages\pandas\io\excel\_base.py:1550, in ExcelFile.__ini
t__(self, path_or_buffer, engine, storage_options, engine_kwargs)
   1548     ext = "xls"
   1549 else:
-> 1550     ext = inspect_excel_format(
   1551         content_or_path=path_or_buffer, storage_options=storage_options
   1552     )
   1553     if ext is None:
   1554         raise ValueError(
   1555             "Excel file format cannot be determined, you must specify "
   1556             "an engine manually."
   1557         )

File ~\anaconda3\Lib\site-packages\pandas\io\excel\_base.py:1402, in inspect_excel_f
ormat(content_or_path, storage_options)
   1399 if isinstance(content_or_path, bytes):
   1400     content_or_path = BytesIO(content_or_path)
-> 1402 with get_handle(
   1403     content_or_path, "rb", storage_options=storage_options, is_text=False
   1404 ) as handle:
   1405     stream = handle.handle
   1406     stream.seek(0)

File ~\anaconda3\Lib\site-packages\pandas\io\common.py:882, in get_handle(path_or_bu
f, mode, encoding, compression, memory_map, is_text, errors, storage_options)
    873         handle = open(
    874             handle,
```

```
    875             ioargs.mode,
   (...)
    878             newline="",
    879         )
    880     else:
    881         # Binary mode
--> 882         handle = open(handle, ioargs.mode)
    883     handles.append(handle)
    885 # Convert BytesIO or file objects passed with an encoding

FileNotFoundError: [Errno 2] No such file or directory: 'C:\\Users\\SHARI\\OneDrive
\\Documents\\Neka\\Uncommon Project Portfolio\\Data_Preparation_Exercise_2022_Cleane
d.xlsx'
```

In [13]:
```
import pandas as pd

data = pd.read_excel("C:\Users\SHARI\OneDrive\Pictures\Documents\Neka\Uncommon Proj

data.head()
```

```
  Cell In[13], line 3
    data = pd.read_excel("C:\Users\SHARI\OneDrive\Pictures\Documents\Neka\Uncommon P
roject Portfolio\Uncommon_AP_Data_Preparation\Data_Preparation_Exercise_2022.xlsx")
                         ^
SyntaxError: (unicode error) 'unicodeescape' codec can't decode bytes in position 2-
3: truncated \UXXXXXXXX escape
```

In [14]:
```
import pandas as pd

data = pd.read_excel("C:\\Users\SHARI\\OneDrive\\Pictures\\Documents\\Neka\\Uncommo

data.head()
```

Out[14]:

	Student Number	Score	Subject
0	303000105	4	Biology
1	301001086	4	Biology
2	306001140	2	Biology
3	303000119	2	Biology
4	303000023	4	Biology

In [16]:
```
# Check column names, data types, and memory usage
data.info()

# Get descriptive statistics for numeric columns (min, max, mean, etc.)
data.describe()

# Check for missing values across all columns
data.isnull().sum()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 720 entries, 0 to 719
Data columns (total 3 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   Student Number  720 non-null    int64 
 1   Score           720 non-null    int64 
 2   Subject         720 non-null    object
dtypes: int64(2), object(1)
memory usage: 17.0+ KB
```

Out[16]:
```
Student Number    0
Score             0
Subject           0
dtype: int64
```

In [17]:
```
# Create a new column indicating whether the student passe the exam
data["Passed"] = data["Score"].apply(lambda x: "Yes" if x >= 3 else "No")

# Display the first few rows to confirm
data.head()
```

Out[17]:

	Student Number	Score	Subject	Passed
0	303000105	4	Biology	Yes
1	301001086	4	Biology	Yes
2	306001140	2	Biology	No
3	303000119	2	Biology	No
4	303000023	4	Biology	Yes

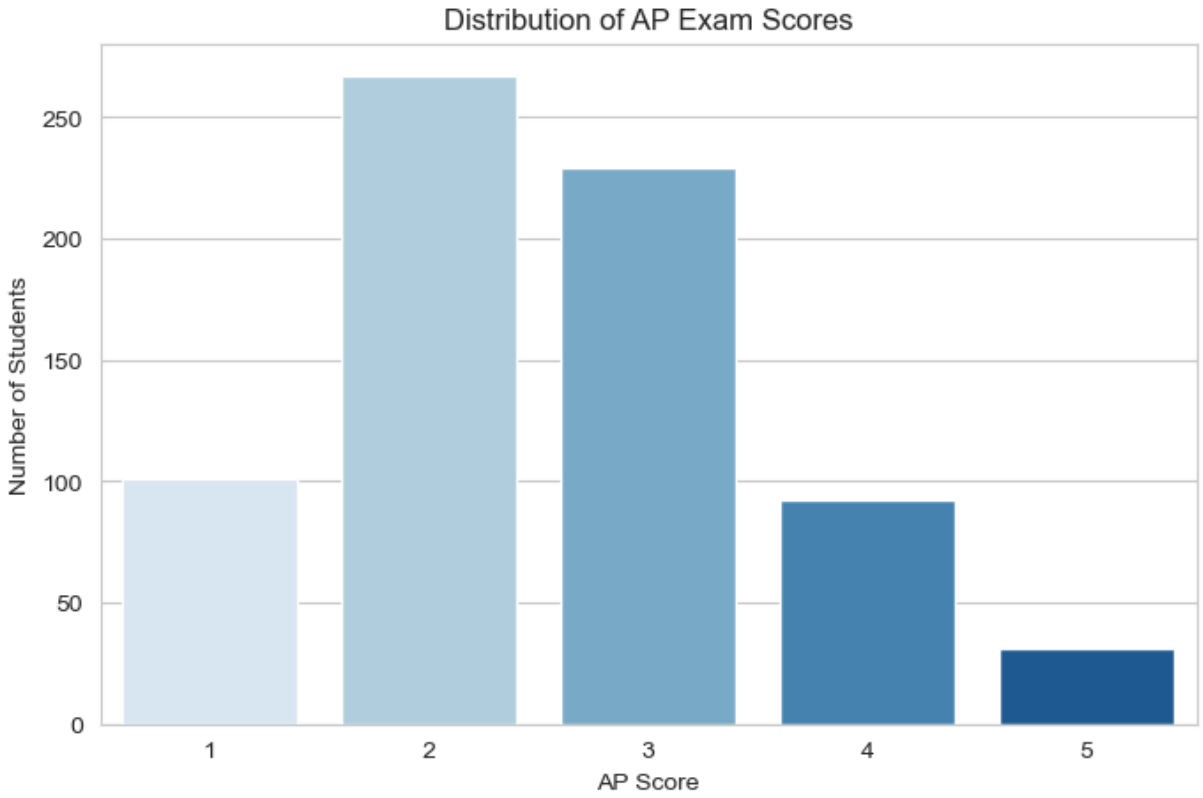
In [19]:
```
# Export the cleaned dataset to CSV format
data.to_csv("C:\\Users\SHARI\\OneDrive\\Pictures\\Documents\\Neka\\Uncommon Project

print("File exported successfully!")
```

```
File exported successfully!
```

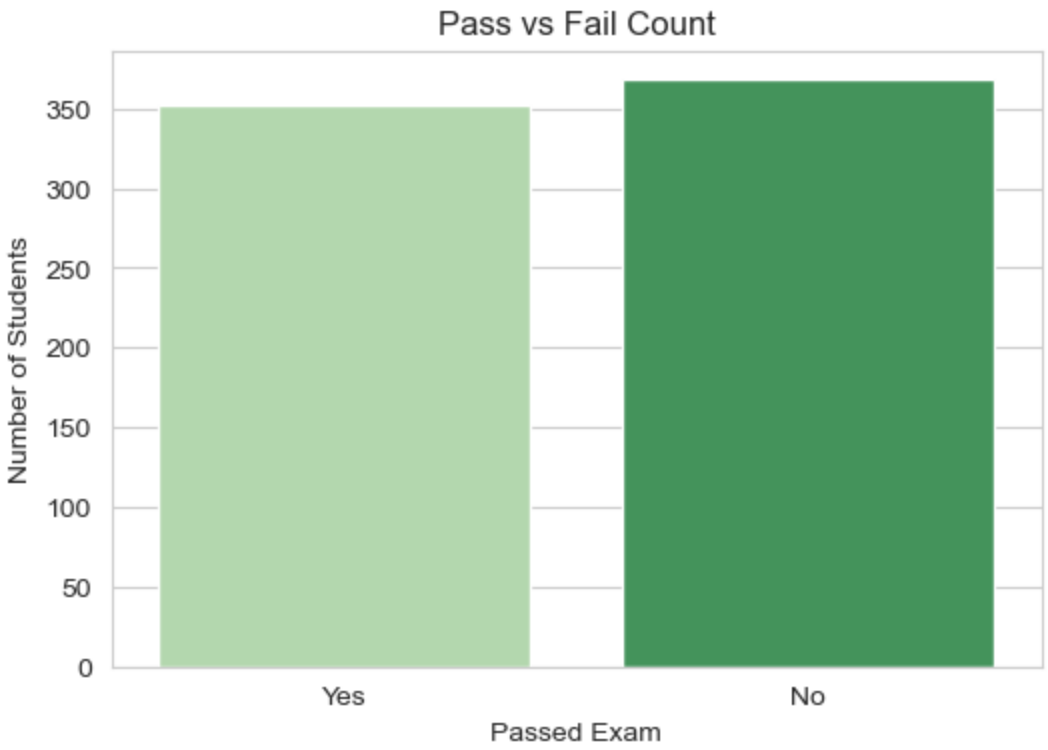
In [26]:
```
import matplotlib.pyplot as plt
import seaborn as sns

#Set a clean style
sns.set_style("whitegrid")

# 1 Distribution of Scores
plt.figure(figsize=(8,5))
sns.countplot(x='Score', data=data, palette='Blues')
plt.title("Distribution of AP Exam Scores")
plt.xlabel("AP Score")
plt.ylabel("Number of Students")
plt.show()

#2 Pass vs Fail
plt.figure(figsize=(6,4))
```

```
sns.countplot(x='Passed', data=data, palette='Greens')
plt.title("Pass vs Fail Count")
plt.xlabel("Passed Exam")
plt.ylabel("Number of Students")
plt.show()
```

```
C:\Users\SHARI\AppData\Local\Temp\ipykernel_27244\3528960652.py:9: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

  sns.countplot(x='Score', data=data, palette='Blues')
```




```
C:\Users\SHARI\AppData\Local\Temp\ipykernel_27244\3528960652.py:17: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.1
4.0. Assign the `x` variable to `hue` and set `legend=False` for the same effect.

  sns.countplot(x='Passed', data=data, palette='Greens')
```



import matplotlib.pyplot as plt import seaborn as sns

Set a clean style

sns.set_style("whitegrid")

1 Distribution of Scores

plt.figure(figsize=(8,5)) sns.countplot(x="Score", data=data, palette="Blues") plt.title("Distribution of AP Exam Scores") plt.xlabel("AP Score") plt.ylabel("Number of Students") plt.show()

2 Pass vs Fail

plt.figure(figsize=(6,4)) sns.countplot(x="Passed", data=data, palette="Greens") plt.title("Pass vs Fail Count") plt.xlabel("Passed Exam") plt.ylabel("Number of Students") plt.show()

In [ ]: