```
# -*- coding: utf-8 -*-
"""python_project.ipynb

Automatically generated by Colab.

Original file is located at

https://colab.research.google.com/drive/1cSboArQVxAuTHk0KhIFpNHzcpKsk94BP
"""

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
# URL of the dataset
url =
"https://docs.google.com/spreadsheets/d/1VP9BE_eI2yl6uUHSm4mGiiwjRdoqCqnk
cIjsv5Q2ex4/export?format=csv"

# Load the dataset directly from the URL
df = pd.read_csv(url)

# Display the first few rows
print(df.head())

# --- Preprocessing ---
# Replace the "height" column with random numbers between 150 and 180
if "height" in df.columns:
    df["height"] = np.random.randint(150, 181, size=len(df))
    print("Height column replaced with random values between 150 and
180.")
else:
    print("Error: 'height' column not found.")

# --- Analysis Tasks ---

# Task 1: Distribution of employees across each team and percentage split
team_distribution = df["Team"].value_counts()
total_employees = len(df)
team_percentage = (team_distribution / total_employees * 100).round(2)

# Visualization for Task 1
plt.figure(figsize=(8, 6))
team_distribution.plot(kind="bar", color="skyblue")
plt.title("Distribution of Employees Across Teams")
plt.ylabel("Number of Employees")
plt.xlabel("Teams")
plt.grid(axis="y")
plt.show()

# Task 2: Segregate employees based on positions
position_distribution = df["Position"].value_counts()

# Visualization for Task 2
plt.figure(figsize=(8, 6))
position_distribution.plot(kind="bar", color="orange")
plt.title("Distribution of Employees by Position")
plt.ylabel("Number of Employees")
```

```
plt.xlabel("Positions")
plt.grid(axis="y")
plt.show()

# Task 3: Identify the predominant age group
df["Age_Group"] = pd.cut(df["Age"], bins=[20, 30, 40, 50, 100],
labels=["20-30", "31-40", "41-50", "51+"])
age_group_distribution = df["Age_Group"].value_counts()

# Visualization for Task 3
plt.figure(figsize=(8, 6))
age_group_distribution.plot(kind="bar", color="green")
plt.title("Distribution of Employees by Age Group")
plt.ylabel("Number of Employees")
plt.xlabel("Age Groups")
plt.grid(axis="y")
plt.show()

# Task 4: Team and position with the highest salary expenditure
salary_expenditure = df.groupby(["Team",
"Position"])["Salary"].sum().reset_index()
highest_salary_expenditure =
salary_expenditure.loc[salary_expenditure["Salary"].idxmax()]

# Visualization for Task 4
plt.figure(figsize=(10, 6))
sns.barplot(x="Team", y="Salary", data=salary_expenditure,
hue="Position", palette="viridis")
plt.title("Salary Expenditure by Team and Position")
plt.ylabel("Total Salary")
plt.xlabel("Teams")
plt.grid(axis="y")
plt.xticks(rotation=45)
plt.show()

# Task 5: Correlation between age and salary
correlation = df["Age"].corr(df["Salary"])
print(f"Correlation between Age and Salary: {correlation:.2f}")

# Scatter plot for Task 5
plt.figure(figsize=(8, 6))
sns.scatterplot(x="Age", y="Salary", data=df, color="purple")
plt.title("Correlation Between Age and Salary")
plt.xlabel("Age")
plt.ylabel("Salary")
plt.grid(True)
plt.show()

# --- Data Story ---
data_story = """
Insights:
Team distribution shows that Team A has the most employees.
Position distribution reveals Analyst is the most common role.
Employees are mostly in the 20-29 age group, indicating a mid-career
workforce.
Salary expenditure is highest for Managers in Team A, suggesting
leadership roles are compensated well in this team.
```

```
The strong positive correlation between age and salary implies that
experience (age) plays a role in determining compensation.

Overall, the analysis reveals significant trends in team distribution,
salary allocation, and age-related patterns.
"""

print(data_story)
```