Python Script: tips.py

```python
import pandas as pd
import matplotlib.pyplot as plt
from reportlab.platypus import SimpleDocTemplate, Paragraph, Spacer, Table,
TableStyle
from reportlab.lib.styles import getSampleStyleSheet
from reportlab.lib.pagesizes import letter
from reportlab.lib import colors
import os


# =========================
# Build correct absolute paths
# =========================

BASE_DIR = os.path.dirname(os.path.abspath(__file__)) # .../Tips/src
DATA_PATH = os.path.join(BASE_DIR, "..", "data",
"tips.csv") # .../Tips/data/tips.csv
# Fix path on Windows
DATA_PATH = os.path.normpath(DATA_PATH)

# =========================
# Load dataset
# =========================
df = pd.read_csv(DATA_PATH)
df['tip_percentage'] = df['tip'] / df['total_bill'] * 100


# =========================
# Terminal print helper
# =========================
analyses_print_text = []


def add_print(title, df_obj):
analyses_print_text.append("\n" + "=" * 46)
analyses_print_text.append(f" {title}")
analyses_print_text.append("=" * 46)
analyses_print_text.append(df_obj.to_string(index=False))
analyses_print_text.append("\n")


# =========================
# ANALYSES (1-14)
# =========================

# 1) Gender Count by Time
gender_time = df.groupby(['time', 'sex']).size().unstack(
fill_value=0).reset_index()
```

```python
add_print("ANALYSIS 1: Gender Count by Time (Dinner vs Lunch)", gender_time)

# 2) Gender Count by Day
gender_day = df.groupby(['day', 'sex']).size().unstack(
fill_value=0).reset_index()
add_print("ANALYSIS 2: Gender Count by Day of Week", gender_day)

# 3) Smoker Count by Gender
smoker_gender = df.groupby(['sex', 'smoker']).size().unstack(
fill_value=0).reset_index()
add_print("ANALYSIS 3: Smoker Count by Gender", smoker_gender)

# 4) Total Tip Amount by Gender
tip_by_gender = df.groupby('sex')['tip'].sum().reset_index()
tip_by_gender.columns = ['Sex', 'Total_Tip_Amount']
add_print("ANALYSIS 4: Total Tip Amount by Gender", tip_by_gender)

# 5) Average Tip by Gender
avg_tip_by_gender = df.groupby('sex')['tip'].mean().reset_index()
avg_tip_by_gender.columns = ['Sex', 'Average_Tip']
add_print("ANALYSIS 5: Average Tip by Gender", avg_tip_by_gender)

# 6) Average Tip Percentage by Gender
avg_tip_percent_gender = df.groupby(
'sex')['tip_percentage'].mean().reset_index()
avg_tip_percent_gender.columns = ['Sex', 'Average_Tip_Percentage']
add_print("ANALYSIS 6: Average Tip Percentage by Gender",
avg_tip_percent_gender)

# 7) Average Tip Percentage by Day
avg_tip_percent_day =
df.groupby('day')['tip_percentage'].mean().reset_index()
avg_tip_percent_day.columns = ['Day', 'Average_Tip_Percentage']
add_print("ANALYSIS 7: Average Tip Percentage by Day", avg_tip_percent_day)

# 8) Average Tip Percentage by Time
avg_tip_percent_time = df.groupby(
'time')['tip_percentage'].mean().reset_index()
avg_tip_percent_time.columns = ['Time', 'Average_Tip_Percentage']
add_print("ANALYSIS 8: Average Tip Percentage by Time (Lunch vs Dinner)",
avg_tip_percent_time)

# 9) Total Customers per Day
cust_per_day = df['day'].value_counts().reset_index()
cust_per_day.columns = ['Day', 'Total_Customers']
cust_per_day = cust_per_day.sort_values('Day')
add_print("ANALYSIS 9: Total Customers per Day", cust_per_day)

# 10) Average Tip by Group Size
avg_tip_by_size = df.groupby('size')['tip'].mean().reset_index()
avg_tip_by_size.columns = ['Group_Size', 'Average_Tip']
add_print("ANALYSIS 10: Average Tip by Group Size", avg_tip_by_size)
```

```python
# 11) Average Tip Percentage by Smoker Status
avg_tip_percent_smoker = df.groupby(
'smoker')['tip_percentage'].mean().reset_index()
avg_tip_percent_smoker.columns = ['Smoker', 'Average_Tip_Percentage']
add_print("ANALYSIS 11: Average Tip Percentage by Smoker Status",
avg_tip_percent_smoker)


# 12) Correlation Matrix
corr_matrix = df[['total_bill', 'tip',
'tip_percentage', 'size']].corr().reset_index()
add_print("ANALYSIS 12: Correlation Matrix", corr_matrix)


# 13) Total Revenue per Day
revenue_per_day = df.groupby('day')['total_bill'].sum().reset_index()
revenue_per_day.columns = ['Day', 'Total_Revenue']
add_print("ANALYSIS 13: Total Revenue (Total Bill) per Day",
revenue_per_day)


# 14) Top 10 Highest Tip Percentage Bills
top_tippers = df.sort_values('tip_percentage', ascending=False).head(
10).reset_index(drop=True)
top_tippers = top_tippers[['total_bill', 'tip',
'tip_percentage', 'sex', 'smoker', 'day', 'time', 'size']]
add_print("ANALYSIS 14: Top 10 Highest Tip Percentage Bills", top_tippers)


# ========================
# Print all analyses
# ========================
terminal_output = "\n".join(analyses_print_text)
print(terminal_output)


# ========================
# Build PDF Report
# ========================
styles = getSampleStyleSheet()

REPORT_PATH = os.path.join(BASE_DIR, "..", "reports",
"tips_advanced_report.pdf")
REPORT_PATH = os.path.normpath(REPORT_PATH)

doc = SimpleDocTemplate(REPORT_PATH, pagesize=letter)
story = []


def df_to_table(df_obj, title):
data = [list(df_obj.columns)]
for _, row in df_obj.iterrows():
data.append(list(row))
```

```python
tbl = Table(data, hAlign='LEFT')

tbl_style = TableStyle([
('BACKGROUND', (0, 0), (-1, 0), colors.HexColor('#4F81BD')),
('TEXTCOLOR', (0, 0), (-1, 0), colors.white),
('FONTNAME', (0, 0), (-1, 0), 'Helvetica-Bold'),
('FONTSIZE', (0, 0), (-1, -1), 8),
('BACKGROUND', (0, 1), (-1, -1), colors.whitesmoke),
('GRID', (0, 0), (-1, -1), 0.25, colors.grey),
('ALIGN', (0, 0), (-1, -1), 'LEFT'),
])

tbl.setStyle(tbl_style)

story.append(Paragraph(title, styles['Heading3']))
story.append(tbl)
story.append(Spacer(1, 12))


# Add tables
df_to_table(gender_time, "Analysis 1: Gender Count by Time (Dinner vs
Lunch)")
df_to_table(gender_day, "Analysis 2: Gender Count by Day of Week")
df_to_table(smoker_gender, "Analysis 3: Smoker Count by Gender")
df_to_table(tip_by_gender, "Analysis 4: Total Tip Amount by Gender")
df_to_table(avg_tip_by_gender, "Analysis 5: Average Tip by Gender")
df_to_table(avg_tip_percent_gender,
"Analysis 6: Average Tip Percentage by Gender")
df_to_table(avg_tip_percent_day, "Analysis 7: Average Tip Percentage by
Day")
df_to_table(avg_tip_percent_time, "Analysis 8: Average Tip Percentage by
Time")
df_to_table(cust_per_day, "Analysis 9: Total Customers per Day")
df_to_table(avg_tip_by_size, "Analysis 10: Average Tip by Group Size")
df_to_table(avg_tip_percent_smoker,
"Analysis 11: Average Tip Percentage by Smoker Status")
df_to_table(corr_matrix, "Analysis 12: Correlation Matrix")
df_to_table(revenue_per_day, "Analysis 13: Total Revenue by Day")
df_to_table(top_tippers, "Analysis 14: Top 10 Highest Tip Percentage Bills")

doc.build(story)

print(f"\nPDF Saved as: {REPORT_PATH}")
```