Task 8

```
#!/usr/bin/env python3
"""
build_task8_package.py
Generates:
- sales_data.csv  (reads Superstore_Sales.csv if present else creates synthetic)
- README.md
- insights.txt

Usage:
   pip install pandas matplotlib reportlab
   python build_task8_package.py
"""
import os
from pathlib import Path
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from datetime import datetime
from reportlab.lib.pagesizes import A4
from reportlab.platypus import SimpleDocTemplate, Paragraph, Spacer, Image, Table, TableStyle, PageBreak
from reportlab.lib.styles import getSampleStyleSheet, ParagraphStyle
from reportlab.pdfbase.cidfonts import UnicodeCIDFont
from reportlab.pdfbase import pdfmetrics
from reportlab.lib import colors

OUT = Path("task8-dashboard-package")
```

```
OUT.mkdir(exist_ok=True)
IMG = OUT / "images"
IMG.mkdir(exist_ok=True)

# 1) Load dataset (if Superstore_Sales.csv present) else create synthetic sample
src = Path("Superstore_Sales.csv")
if src.exists():
    df = pd.read_csv(src, parse_dates=['Order Date'], dayfirst=False)
    df.rename(columns=lambda c: c.strip(), inplace=True)
else:
    # synthetic sample (monthly, regions, categories)
    rng = pd.date_range("2023-01-01", periods=180, freq='D')
    regions = ["East","West","Central","South"]
    categories = ["Furniture","Office Supplies","Technology"]
    rows = []
    np.random.seed(42)
    for d in rng:
        for _ in range(np.random.randint(5, 12)):
            rows.append({
                "Order Date": d,
                "Region": np.random.choice(regions, p=[0.3,0.25,0.2,0.25]),
                "Category": np.random.choice(categories, p=[0.25,0.45,0.30]),
                "Sales": round(np.random.uniform(10, 1200), 2),
                "Profit": round(np.random.uniform(-100, 400), 2)
            })
    df = pd.DataFrame(rows)

# normalize column names
```

```
if 'Order Date' not in df.columns and 'OrderDate' in df.columns:
   df['Order Date'] = pd.to_datetime(df['OrderDate'])
if not pd.api.types.is_datetime64_any_dtype(df['Order Date']):
   df['Order Date'] = pd.to_datetime(df['Order Date'], errors='coerce')

# add month-year
df['Month'] = df['Order Date'].dt.to_period('M').dt.to_timestamp()
# save working CSV
df.to_csv(OUT / "sales_data.csv", index=False)

# 2) Aggregations for visuals
monthly = df.groupby('Month', observed=True).agg(Sales=('Sales','sum')).reset_index()
by_region = df.groupby('Region', observed=True).agg(Sales=('Sales','sum')).reset_index()
by_category = df.groupby('Category', observed=True).agg(Sales=('Sales','sum')).reset_index()

# 3) Create charts
plt.style.use('seaborn-darkgrid')

# KPI banner
total_sales = df['Sales'].sum()
total_profit = df['Profit'].sum() if 'Profit' in df.columns else 0.0
orders_count = df.shape[0]
avg_order = total_sales / orders_count if orders_count else 0
plt.figure(figsize=(10,1.6))
plt.axis('off')
kpi_text = f"Total Sales: ${total_sales:,.2f}   |   Total Profit: ${total_profit:,.2f}   |   Orders: {orders_count:,}   |   Avg Order: ${avg_order:,.2f}"
plt.text(0.01, 0.5, kpi_text, fontsize=10)
plt.savefig(IMG / "kpi_banner.png", bbox_inches='tight', dpi=200)

plt.close()


# Monthly sales line

plt.figure(figsize=(10,4))

plt.plot(monthly['Month'], monthly['Sales'], marker='o', linewidth=1)

plt.title("Sales over Months")

plt.xlabel("Month")

plt.ylabel("Sales")

plt.tight_layout()

plt.savefig(IMG / "monthly_sales.png", dpi=200)

plt.close()


# Sales by region bar

plt.figure(figsize=(7,4))

plt.bar(by_region['Region'], by_region['Sales'])

plt.title("Sales by Region")

plt.xlabel("Region")

plt.ylabel("Sales")

plt.tight_layout()

plt.savefig(IMG / "sales_by_region.png", dpi=200)

plt.close()


# Donut chart for category share

plt.figure(figsize=(6,6))

vals = by_category['Sales']

labels = by_category['Category']
```

```
wedges, texts = plt.pie(vals, labels=labels, startangle=140, wedgeprops=dict(width=0.4))
plt.title("Sales by Category (Donut)")
plt.tight_layout()
plt.savefig(IMG / "sales_by_category_donut.png", dpi=200)
plt.close()

# 4) Create insights.txt
insights = [
  "Insight 1: The region with the highest total sales is {}.".format(by_region.sort_values('Sales', ascending=False).iloc[0]['Region']),
  "Insight 2: Category with largest revenue share is {}.".format(by_category.sort_values('Sales', ascending=False).iloc[0]['Category']),
  "Insight 3: Monthly sales show peaks and troughs — examine marketing or promotions around peak months.",
  "Insight 4: Average order value suggests focusing on upsell strategies if AOV is lower than target."
]
with open(OUT / "insights.txt", "w") as f:
  for line in insights:
    f.write(line + "\n")

# Executive summary
story.append(Paragraph("<b>Executive summary</b>", styles['Heading2']))
exec_txt = ("This report presents a simple dashboard design showing sales performance by month, region and category. "
      "It includes clear KPIs, a time-series chart, regional breakdown, and category share visualization. "
      "Use the included images and README to re-create the dashboard in Power BI or Tableau.")
```

```
story.append(Paragraph(exec_txt, styles['Justify']))
story.append(PageBreak())

# README content
story.append(Paragraph("<b>README</b>", styles['Heading2']))
readme_lines = [
    "Project: Task 8 — Simple Sales Dashboard Design",
    "Files included: sales_data.csv, insights.txt, README.md",

]
for l in readme_lines:
    story.append(Paragraph(l, styles['Normal']))
story.append(PageBreak())

# Dataset sample
story.append(Paragraph("<b>Dataset sample (first 10 rows)</b>", styles['Heading2']))
sample = df.head(10).astype(str)
tbl = [sample.columns.tolist()] + sample.values.tolist()
t = Table(tbl, repeatRows=1)
t.setStyle(TableStyle([("BACKGROUND",(0,0),(-1,0),colors.lightgrey),("GRID",(0,0),(-1,-1),0.25,colors.grey),("FONTSIZE",(0,0),(-1,-1),8)]))
story.append(t)
story.append(PageBreak())

# Visuals section
story.append(Paragraph("<b>Visualizations</b>", styles['Heading2']))
for img in ["kpi_banner.png","monthly_sales.png","sales_by_region.png","sales_by_category_donut.png"]:
    path = IMG / img
    if path.exists():
        story.append(Paragraph(img.replace('_',' ').replace('.png','').title(), styles['Heading3']))
        story.append(Image(str(path), width=450, height=250))
        story.append(Spacer(1,8))
story.append(PageBreak())

# Insights
story.append(Paragraph("<b>Insights</b>", styles['Heading2']))
for line in insights:
    story.append(Paragraph("- " + line, styles['Normal']))

doc.build(story)

# 6) README.md
readme_md = OUT / "README.md"
readme_md_text = f"""# Task 8 — Simple Sales Dashboard Design

This repo contains deliverables for Task 8 (Dashboard Design).

Files:
- sales_data.csv (source dataset)
- insights.txt
- README.md (this file)

Recreate in Power BI / Tableau:
1. Import sales_data.csv
2. Convert Order Date to Month-Year
```

```
3. Create KPI cards: Total Sales, Total Profit, Orders, AOV
4. Create visuals:
  - Line chart: Sales over Months
  - Bar chart: Sales by Region
  - Donut chart: Sales by Category
5. Add slicer: Region or Category
"""
readme_md.write_text(readme_md_text)


print("Package ready at:", OUT.resolve())
print("Open", pdf_path.resolve())
```