**أولاً: المشكلات التي وُجدت في البيانات الخام**

واجهت البيانات الخام عدداً من المشكلات الجوهرية التي استلزمت معالجتها قبل البدء في أي تحليل أو عرض رسومي. فيما يلي عرض تفصيلي لهذه المشكلات مع توضيح طرق التعامل معها:

**١- القيم المفقودة (Missing Values)**

كانت بعض الأعمدة تحتوي على قيم مفقودة، مما قد يؤدي إلى تحليلات غير دقيقة. تمثلت هذه القيم المفقودة في:

* غياب بيانات "سبب التأخير" (Reason for Delay) في العديد من السجلات، خاصةً للرحلات التي وصلت في موعدها.
* غياب بيانات "وقت الوصول الفعلي" (Actual Arrival Time) للرحلات التي تم إلغاؤها.
* غياب أو ظهور قيمة "None" في عمود "بطاقة التخفيض" (Railcard).

**طريقة المعالجة:**

* تم ملء القيم المفقودة في عمود "سبب التأخير" بقيمة توضيحية "لا يوجد تأخير" (No Delay) في حالة الرحلات التي لم تتأخر.
* تم ملء القيم المفقودة في عمود "وقت الوصول الفعلي" بكلمة "ملغاة" (Cancelled) للرحلات التي تم إلغاؤها.
* تم استبدال قيمة "None" في عمود "بطاقة التخفيض" بعبارة "لا توجد بطاقة تخفيض" (No Railcard) لتعزيز وضوح البيانات.

**٢- تنسيقات البيانات غير الصحيحة (Incorrect Data Formats)**

لوحظ أن بعض الأعمدة الزمنية والتاريخية كانت مخزنة كنصوص عادية (Strings)، مما يصعّب إجراء الحسابات أو التحليلات الزمنية.

**طريقة المعالجة:**

* تم تحويل أعمدة "تاريخ الرحلة" (Date of Journey) و"تاريخ الشراء" (Date of Purchase) إلى نوع بيانات التاريخ (Datetime).
* تم التأكد من أن الأوقات (مثل "وقت الشراء"، و"وقت المغادرة"، و"وقت الوصول المجدول") موحدة التنسيق بصيغة (ساعة:دقيقة:ثانية) (HH:MM:SS).
* تم التأكد من أن عمود "سعر التذكرة" (Ticket Price) مخزن كبيانات رقمية مناسبة للتحليل الإحصائي.

**٣- وجود مسافات زائدة في النصوص (Extra Whitespace)**

كان هناك احتمال لوجود مسافات زائدة في أسماء الأعمدة أو القيم النصية، مما قد يؤدي إلى أخطاء أثناء التصنيف أو التجميع.

**طريقة المعالجة:**

* تم إزالة جميع المسافات الزائدة من أسماء الأعمدة باستخدام دوال تنقية النصوص.
* تم تنظيف جميع القيم النصية داخل الأعمدة، بحيث لا تحتوي على مسافات غير مرغوبة في البداية أو النهاية.

**٤- عدم اتساق تسمية الفئات (Inconsistent Category Labels)**

لوحظ أن بعض الفئات النصية التي تمثل نفس المعنى كانت مكتوبة بأشكال مختلفة، مثل:

* ظهور "Staffing" و"Staff Shortage" للإشارة إلى نفس المشكلة (نقص الموظفين).
* ظهور "Weather" و"Weather Conditions" للإشارة إلى نفس الظاهرة الجوية.
* وجود اختلاف في حالة الأحرف مثل "Signal failure" بدلاً من "Signal Failure".

**طريقة المعالجة:**

* تم توحيد جميع هذه القيم إلى صيغة واحدة واضحة:
  + توحيد "Staffing" إلى "Staff Shortage".
  + توحيد "Weather Conditions" إلى "Weather".
  + تصحيح حالة الأحرف لتصبح "Signal Failure" بشكل موحد عبر جميع السجلات.

**٥- التكرارات في السجلات (Duplicate Records)**

عند مراجعة البيانات الخام، تم رصد وجود بعض السجلات المكررة التي تحمل نفس تفاصيل الرحلة مع اختلاف طفيف أو بدون اختلاف، مما قد يؤدي إلى تحليلات مضللة.

**طريقة المعالجة:**

* تم فحص البيانات واكتشاف التكرارات بناءً على تطابق الأعمدة.
* تم حذف السجلات المكررة، والإبقاء فقط على سجل واحد لكل رحلة لضمان دقة النتائج.

بعد تنفيذ عمليات التنظيف السابقة، أصبحت البيانات خالية من القيم المفقودة، موحدة الصيغ، خالية من المسافات الزائدة، خالية من التكرارات، ومنظمة بشكل يسمح بإجراء تحليلات دقيقة وموثوقة.  
هذه المعالجة الدقيقة مهدت الطريق لإنشاء داشبوردات تحليلية دقيقة وعالية الجودة لاحقاً في المشروع.

**ثانياً: شرح كود التنظيف والتحليل**

# استيراد المكتبات

import pandas as pd

import numpy as np

import matplotlib.pyplot as plt

import seaborn as sns

تم استيراد مكتبات إدارة البيانات (pandas)، العمليات الحسابية (numpy)، والرسم البياني (matplotlib و seaborn)، استعداداً للمعالجة والتحليل البصري.

# تحميل البيانات

df = pd.read\_csv(r"D:\\railway 2.csv")

تم تحميل ملف البيانات الخام بصيغة CSV إلى إطار بيانات (DataFrame) يسمى df.

# نظافة البيانات

print(df.head())

print(df.tail())

print(df.describe())

print(df.info())

print(df.isnull().sum())

print(df.dtypes)

print(df.duplicated().sum())

استعراض أول وآخر خمس صفوف، وإظهار ملخص إحصائي، ومراجعة أنواع البيانات والقيم المفقودة والتكرارات بهدف تقييم جودة البيانات قبل بدء التنظيف.

# إزالة المسافات الزائدة من أسماء الأعمدة والقيم النصية

df.columns = df.columns.str.strip()

df = df.applymap(lambda x: x.strip() if isinstance(x, str) else x)

تمت إزالة أي مسافات إضافية في أسماء الأعمدة وفي جميع القيم النصية الموجودة داخل الجدول، لضمان تنسيق موحد ودقيق.

# تنظيف النصوص بشكل أدق للأعمدة النصية فقط

str\_cols = df.select\_dtypes(include=['object']).columns

df[str\_cols] = df[str\_cols].apply(lambda x: x.str.strip())

تم تحديد الأعمدة النصية فقط، ثم تنظيفها من أي مسافات زائدة بشكل خاص دون التأثير على الأعمدة العددية.

# تحويل التواريخ إلى النوع التاريخي

df['Date of Journey'] = pd.to\_datetime(df['Date of Journey'], errors='coerce')

df['Date of Purchase'] = pd.to\_datetime(df['Date of Purchase'], errors='coerce')

تم تحويل أعمدة "تاريخ الرحلة" و"تاريخ الشراء" إلى صيغة تاريخية (Datetime)، لتسهيل العمليات الزمنية مثل الفرز والتحليل عبر الزمن.

# معالجة القيم المفقودة

df.loc[(df['Journey Status'] == 'On Time') & (df['Reason for Delay'].isna()), 'Reason for Delay'] = 'No Delay'

df.loc[(df['Journey Status'] == 'Delayed') & (df['Reason for Delay'].isna()), 'Reason for Delay'] = 'Unknown'

df['Actual Arrival Time'] = df['Actual Arrival Time'].fillna('Cancelled')

df['Railcard'] = df['Railcard'].fillna('No Railcard')

تم معالجة القيم المفقودة:

* تعيين "No Delay" للرحلات في موعدها بدون سبب تأخير،
* تعيين "Unknown" للرحلات المتأخرة بدون سبب،
* إدخال "Cancelled" كوقت وصول فعلي للرحلات الملغاة،
* إدخال "No Railcard" للركاب الذين لم يستخدموا بطاقة تخفيض.

# تعديل بعض القيم الموحدة

df['Reason for Delay'] = df['Reason for Delay'].replace({

'Staffing': 'Staff Shortage',

'Weather Conditions': 'Weather'

})

تم توحيد القيم المتعددة التي تعبر عن نفس المعنى، مثل استبدال "Staffing" بـ"Staff Shortage"، و"Weather Conditions" بـ"Weather"، لضمان الاتساق في التحليل.

# مراجعة النظافة بعد المعالجة

print(df.isnull().sum())

print(df.dtypes)

تم التحقق مجدداً من خلو الأعمدة من القيم المفقودة، ومن أن جميع الأعمدة تحمل النوع الصحيح للبيانات.

# حفظ نسخة من الداتا بعد التنظيف

df.to\_csv(r"D:\\railway 2\_cleaned.csv", index=False)

تم حفظ نسخة جديدة من البيانات المنظفة تحت اسم railway 2\_cleaned.csv للعمل عليها في الخطوات التالية دون الحاجة إلى إعادة التنظيف.

# تهيئة الرسوم

sns.set\_style("whitegrid")

plt.style.use('default')

تم تحديد نمط الرسوم البيانية ليكون بسيطًا ومنظماً بخلفية بيضاء مع شبكة خفيفة، مما يسهل قراءة البيانات المرئية.

# ألوان متدرجة مخصصة للرسوم

custom\_palette = ['#004c6d', '#2176b7', '#57a0d3', '#a9c9eb', '#dceefb']

تم إعداد لوحة ألوان مخصصة تستخدم في جميع الرسوم البيانية لخلق تناسق لوني مميز وجذاب.

# Dashboard 1

fig1, axs1 = plt.subplots(2, 2, figsize=(18, 12))

axs1 = axs1.flatten()

تم إنشاء لوحة رسم بياني مكونة من أربعة مخططات فرعية (2×2) بمساحة واسعة مناسبة لعرض الرسوم بشكل واضح.

# رسم التوزيعات المختلفة (Dashboard 1)

# 1- توزيع حالة الرحلات

sns.countplot(x='Journey Status', data=df, palette=custom\_palette, ax=axs1[0])

تم إنشاء رسم عمودي يظهر توزيع حالات الرحلات بين الوصول في الموعد، التأخير، والإلغاء.

# 2- أسباب التأخير

delay\_reasons = df[df['Reason for Delay'] != 'No Delay']['Reason for Delay'].value\_counts()

sns.barplot(x=delay\_reasons.values, y=delay\_reasons.index, palette=custom\_palette, ax=axs1[1])

تم إنشاء رسم عمودي يُظهر أكثر الأسباب شيوعاً لحدوث التأخير في الرحلات.

# 3- أكثر 5 محطات مغادرة

top\_departure = df['Departure Station'].value\_counts().head(5)

sns.barplot(x=top\_departure.values, y=top\_departure.index, palette=custom\_palette, ax=axs1[2])

تم رسم أكثر خمس محطات مغادرة نشطة وفقاً لعدد الرحلات التي تنطلق منها.

# 4- أكثر 5 وجهات وصول

top\_arrival = df['Arrival Destination'].value\_counts().head(5)

sns.barplot(x=top\_arrival.values, y=top\_arrival.index, palette=custom\_palette, ax=axs1[3])

تم رسم أكثر خمس وجهات وصول شعبية للرحلات.

# إعداد عنوان الداشبورد الأول

plt.suptitle('Railway Dashboard - Part 1', fontsize=20, color='#004c6d', y=1.02)

plt.tight\_layout(rect=[0, 0, 1, 0.96])

fig1.savefig('railway\_dashboard\_part1.png', dpi=300, bbox\_inches='tight')

plt.show()

تم إضافة عنوان موحد للداشبورد الأول، وضبط المسافات بين المخططات، ثم حفظ الداشبورد كصورة PNG عالية الجودة.

# Dashboard 2

fig2, axs2 = plt.subplots(2, 2, figsize=(18, 12))

axs2 = axs2.flatten()

تم إنشاء لوحة ثانية تحتوي على أربعة مخططات فرعية لعرض تحليلات إضافية.

# 1- توزيع وسائل الدفع

payment\_methods = df['Payment Method'].value\_counts()

axs2[0].pie(payment\_methods.values, labels=payment\_methods.index, autopct='%1.1f%%', startangle=140,

colors=custom\_palette, textprops={'color':"black"})

تم رسم مخطط دائري يوضح نسب استخدام كل وسيلة دفع من وسائل الدفع المختلفة.

# 2- توزيع استخدام بطاقات التخفيض

railcard\_usage = df['Railcard'].value\_counts()

sns.barplot(x=railcard\_usage.values, y=railcard\_usage.index, palette=custom\_palette, ax=axs2[1])

تم رسم توزيع استخدام أنواع بطاقات التخفيض المختلفة.

# 3- التذاكر المباعة عبر الزمن

sales\_over\_time = df['Date of Purchase'].value\_counts().sort\_index()

axs2[2].plot(sales\_over\_time.index, sales\_over\_time.values, marker='o', color='#00BFFF')

تم رسم مخطط خطي يوضح كيف تغير عدد التذاكر المباعة مع مرور الوقت.

# 4- أكثر محطات المغادرة التي بها تأخيرات

delayed\_stations = df[df['Journey Status'] == 'Delayed']['Departure Station'].value\_counts().head(5)

sns.barplot(x=delayed\_stations.values, y=delayed\_stations.index, palette=custom\_palette, ax=axs2[3])

تم عرض أكثر محطات المغادرة التي تتكرر بها التأخيرات باستخدام رسم بياني عمودي.

# إعداد عنوان الداشبورد الثاني

plt.suptitle('Railway Dashboard - Part 2', fontsize=20, color='#004c6d', y=1.02)

plt.tight\_layout(rect=[0, 0, 1, 0.96])

fig2.savefig('railway\_dashboard\_part2.png', dpi=300, bbox\_inches='tight')

plt.show()

تم إضافة عنوان موحد للداشبورد الثاني، وضبط المسافات بين المخططات، ثم حفظ الداشبورد الثاني كصورة PNG عالية الجودة.

# توحيد كتابة 'Signal failure' إلى 'Signal Failure'

df['Reason for Delay'] = df['Reason for Delay'].replace('Signal failure', 'Signal Failure')

يبحث الكود في عمود Reason for Delay.

كل قيمة مكتوبة حرف صغير في كلمة Signal Failure يتم استبدالها بالقيمة الصحيحة.

# إزالة التكرارات

df = df.drop\_duplicates()

الكود يبحث عن أي صفوف مكررة تمامًا داخل الداتا.و يقوم بحذفها تلقائيًا مع الاحتفاظ بأول نسخة فقط من الصف المتكرر.

بعد الكود ده الداتا (df) تصبح خالية ١٠٠٪ من التكرارات.

#dashboard 3

sns.set\_style("whitegrid")

plt.style.use('default')

يتم هنا ضبط شكل الرسوم البيانية بحيث تكون الخلفية شبكية (whitegrid)، والأسلوب العام للرسم هو الافتراضي البسيط لضمان وضوح البيانات.

fig3, axs3 = plt.subplots(2, 2, figsize=(18, 12))

axs3 = axs3.flatten()

يتم إنشاء شكل (Figure) يحتوي على شبكة من 2 صف × 2 عمود، أي 4 أماكن (Subplots) للرسم.  
 يتم تحويلهم إلى مصفوفة مسطحة (flatten()) لتسهيل التعامل مع كل رسم بياني على حدة.

df['Day\_of\_Week'] = df['Date of Journey'].dt.day\_name()

delay\_by\_day = df.groupby('Day\_of\_Week')['Journey Status'].apply(lambda x: (x == 'Delayed').mean() \* 100)

delay\_by\_day = delay\_by\_day.reindex([

'Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday'

])

delay\_by\_day.plot(kind='barh', color='#2176b7', ax=axs3[0])

يتم هنا إنشاء عمود جديد Day\_of\_Week يحول تاريخ الرحلة إلى اسم اليوم.  
 بعد ذلك يتم حساب نسبة الرحلات المتأخرة في كل يوم باستخدام groupby و lambda.  
 يتم ترتيب الأيام ترتيبًا طبيعيًا بدايةً من الاثنين.  
أخيرًا يتم رسم النتائج على هيئة رسم بياني أفقي (barh) باللون الأزرق.

sns.histplot(df['Price'], bins=30, kde=True, color='#004c6d', ax=axs3[1])

يتم رسم توزيع الأسعار باستخدام histplot مع 30 فئة (bins)،  
 بالإضافة إلى رسم منحنى الكثافة الاحتمالية (KDE) لتوضيح شكل توزيع الأسعار،  
 الرسم يتم باللون الأزرق الداكن (#004c6d).

df['Departure\_Hour'] = pd.to\_datetime(df['Departure Time'], errors='coerce').dt.hour

delay\_by\_hour = df.groupby('Departure\_Hour')['Journey Status'].apply(lambda x: (x == 'Delayed').mean() \* 100)

delay\_by\_hour.plot(kind='line', marker='o', color='#00BFFF', ax=axs3[2])

يتم تحويل وقت المغادرة إلى ساعة (رقم من 0 إلى 23) وحفظه في عمود جديد Departure\_Hour.  
 يتم حساب نسبة التأخير في كل ساعة.  
 يتم رسم خط بياني (Line Plot) يوضح كيف تتغير نسبة التأخير خلال ساعات اليوم، مع نقاط بارزة (Marker 'o').

sns.boxplot(x='Journey Status', y='Price', data=df, palette='Blues', ax=axs3[3])

يتم رسم مخطط صندوقي (Boxplot) لمقارنة توزيع أسعار التذاكر حسب حالة الرحلة (On Time, Delayed, Cancelled).  
 يوضح هذا التحليل الفروقات المحتملة في الأسعار حسب حالة الرحلة باستخدام تدرج لوني أزرق (Blues palette).

plt.suptitle('Railway Dashboard - Part 3, fontsize=20, color='#004c6d', y=1.02)

plt.tight\_layout(rect=[0, 0, 1, 0.96])

يتم إضافة عنوان رئيسي للداشبورد (Railway Dashboard - Part 3: Advanced Analysis)  
 ويتم ضبط المسافات حول الرسومات باستخدام tight\_layout بحيث لا تتداخل العناوين مع الرسومات.

fig3.savefig('railway\_dashboard\_part3.png', dpi=300, bbox\_inches='tight')

plt.show()

يتم حفظ الداشبورد كصورة عالية الجودة باسم railway\_dashboard\_part3.png.  
 وأخيرًا يتم عرض الشكل باستخدام plt.show().

**شرح الداشبورد الأول: Railway Dashboard - Part 1**

**تحليل أداء الرحلات وأسباب التأخير والمحطات الرئيسية**

في هذا الجزء من التحليل البصري للبيانات، تم إعداد لوحة معلومات بعنوان "Railway Dashboard - Part 1"، والتي تهدف إلى استكشاف الأداء العام للرحلات من حيث الالتزام بالمواعيد، وتحليل أسباب التأخير، وتحديد المحطات الأكثر نشاطًا سواءً من حيث الانطلاق أو الوصول. وقد تمثلت مكونات هذه اللوحة في أربعة رسوم بيانية رئيسية كما يلي:

**1- توزيع حالة الرحلات (Journey Status Distribution)**

في هذا الرسم البياني، تم عرض توزيع الرحلات إلى ثلاث فئات رئيسية:

* **رحلات وصلت في الموعد المحدد** (On Time).
* **رحلات تعرضت للتأخير** (Delayed).
* **رحلات تم إلغاؤها** (Cancelled).

وقد أظهرت النتائج أن الغالبية العظمى من الرحلات وصلت في موعدها دون تأخير، بينما كانت نسبة الرحلات المتأخرة أو الملغاة أقل بكثير، مما يعكس مؤشرات إيجابية حول التزام القطارات بمواعيدها.

**2- أسباب التأخير (Reasons for Delay)**

يقدم هذا الرسم البياني تحليلاً لأكثر الأسباب شيوعًا التي أدت إلى تأخير الرحلات.  
وقد تبين أن:

* **الأحوال الجوية (Weather)** كانت السبب الرئيسي للتأخير،
* تلتها مشكلات مثل **فشل الإشارة (Signal Failure)** و**نقص الموظفين (Staff Shortage)** و**المشكلات التقنية (Technical Issue)**.

ويُعد هذا التحليل ضروريًا لدعم قرارات تحسين الأداء التشغيلي ومعالجة الأسباب المتكررة للتأخير.

**3- المحطات الخمسة الأعلى مغادرة (Top 5 Departure Stations)**

يعرض هذا الرسم البياني أكثر خمس محطات شهدت أكبر عدد من الرحلات المغادرة.  
وقد جاءت محطة **Manchester Piccadilly** في صدارة القائمة، تلتها محطات أخرى بارزة مثل **London Euston** و**Liverpool Lime Street**.  
هذا التحليل يساعد في تحديد المحطات الرئيسية التي تتطلب إدارة تشغيلية فعالة ودعمًا إضافيًا في أوقات الذروة.

**4- المحطات الخمسة الأعلى وصولًا (Top 5 Arrival Destinations)**

يسلط هذا الرسم الضوء على أكثر خمس محطات استقبلت الرحلات.  
وقد احتلت محطة **Birmingham New Street** المركز الأول كأكثر محطة وصول ازدحامًا، متبوعة بمحطات أخرى مثل **Liverpool Lime Street** و**York**.

يعد هذا النوع من التحليل مهمًا للتخطيط المستقبلي وتحسين الخدمات المقدمة في المحطات الحيوية.

**شرح الداشبورد الثاني: Railway Dashboard - Part 2**

**تحليل أنماط الدفع، استخدام البطاقات، مبيعات التذاكر، وتأخيرات المحطات**

يتناول هذا الجزء من التحليل البصري مجموعة من المؤشرات المهمة التي تعكس سلوك الركاب من حيث طرق الدفع المستخدمة، استخدام البطاقات المخفضة، تطور مبيعات التذاكر عبر الزمن، بالإضافة إلى تحليل المحطات الأكثر تعرضاً للتأخير. وقد تم تجميع هذه التحليلات في أربعة رسوم بيانية رئيسية كما يلي:

**1- توزيع وسائل الدفع (Payment Method Distribution)**

يعرض هذا الرسم البياني نسب استخدام وسائل الدفع المختلفة عند شراء التذاكر، وتشمل:

* **بطاقات الائتمان (Credit Card)**،
* **الدفع اللاتلامسي (Contactless)**،
* **بطاقات الخصم (Debit Card)**.

وقد أظهرت النتائج أن بطاقات الائتمان كانت الوسيلة الأكثر شيوعًا بين الركاب، تليها طريقة الدفع اللاتلامسي، بينما جاءت بطاقات الخصم بنسبة أقل بكثير.  
هذا التحليل يساعد في فهم تفضيلات الركاب ومن ثم تطوير الخدمات المالية المقدمة لهم.

**2- استخدام البطاقات المخفضة (Railcard Usage)**

يوضح هذا الرسم البياني توزيع استخدام أنواع البطاقات المخفضة المختلفة مثل:

* **No Railcard** (عدم استخدام بطاقة تخفيض)،
* **Adult Railcard** (بطاقة البالغين)،
* **Disabled Railcard** (بطاقة ذوي الاحتياجات الخاصة)،
* **Senior Railcard** (بطاقة كبار السن).

وقد تبين أن النسبة الأكبر من الركاب لا يستخدمون بطاقات تخفيض، في حين جاءت بطاقة البالغين في المرتبة الثانية من حيث الاستخدام.  
يوفر هذا التحليل معلومات مهمة عند تصميم الحملات الترويجية والعروض الخاصة بالركاب.

**3- مبيعات التذاكر عبر الزمن (Tickets Sold Over Time)**

يستعرض هذا الرسم البياني تطور عدد التذاكر المباعة يوميًا خلال الفترة الزمنية المشمولة في الدراسة.

وقد لاحظنا:

* وجود ارتفاع ملحوظ في مبيعات التذاكر في بداية العام،
* تلاه انخفاض تدريجي قد يكون مرتبطًا بالعوامل الموسمية أو فترات العطلات.

هذا التحليل مفيد جدًا في التخطيط لتوزيع الموارد البشرية واللوجستية خلال فترات الذروة والانخفاض.

**4- المحطات الخمسة الأعلى في التأخيرات (Top Departure Stations with Delays)**

يعرض هذا الرسم البياني المحطات التي شهدت أكبر عدد من الرحلات المتأخرة.  
وقد جاءت محطة **Liverpool Lime Street** في المرتبة الأولى، تليها محطات أخرى مثل **Manchester Piccadilly** و**London Euston**.

يمكّن هذا التحليل إدارات المحطات من دراسة الأسباب الخاصة بالتأخيرات وتحسين إجراءات التشغيل لتقليل معدلات التأخير مستقبلاً.

**الملاحظات الخاصة بـ Dashboard 1:**

**1- ملاحظة على أسباب التأخير (Reasons for Delay):**

لوحظ ضمن تصنيف أسباب التأخير وجود فئة عامة تحت اسم **Other**.  
تشير هذه الفئة إلى مجموعة الأسباب التي لم يتم تصنيفها ضمن الأسباب الرئيسية مثل الطقس أو فشل الإشارة أو المشكلات التقنية.  
يُوصى عند عرض هذه النتائج بذكر أن فئة "Other" تشمل أسبابًا متنوعة وغير محددة بدقة في البيانات الأصلية.

**الملاحظات الخاصة بـ Dashboard 2:**

**1- ملاحظة على استخدام بطاقات التخفيض (Railcard Usage):**

أظهرت البيانات أن نسبة كبيرة من الركاب لا يستخدمون بطاقات تخفيض مصنفين تحت "No Railcard").)  
وهذا يعكس واقعًا قد يرتبط بطبيعة الفئات المستهدفة أو أن بعض المسافرين يفضلون الحجز دون استخدام بطاقات خصم.  
عند عرض هذا التحليل، من المفيد الإشارة إلى هذه الملاحظة لتوضيح التركيبة السكانية للركاب.

**2- ملاحظة على مبيعات التذاكر عبر الزمن (Tickets Sold Over Time):**

تبين من خلال الرسم البياني وجود تذبذب ملحوظ في أعداد التذاكر المباعة على مدار الفترة الزمنية، مع وجود فجوات في بعض الأيام.  
من المحتمل أن تكون هذه الفجوات ناتجة عن:

* فترات الإجازات الرسمية،
* أو انخفاض الطلب الموسمي،
* أو نقص في تسجيل بعض البيانات.

ينبغي عند تفسير هذا الرسم الإشارة إلى هذا التذبذب باعتباره نمطًا طبيعياً في حركة مبيعات التذاكر على مدار السنة.

**شرح "Railway Dashboard - Part 3 "**

تتكوّن هذه الداشبورد من أربع تحليلات تفصيلية تهدف إلى فهم العوامل المؤثرة على تأخير الرحلات وأسعار التذاكر في بيانات السكك الحديدية. فيما يلي شرح مفصل لكل جزء:

**١. تحليل نسبة التأخير حسب يوم الأسبوع**

**Delay Percentage by Day of the Week**

* يوضح هذا الرسم البياني الأفقي نسبة الرحلات المتأخرة في كل يوم من أيام الأسبوع.
* نلاحظ أن يوم **الخميس** يظهر فيه أعلى نسبة تأخير مقارنة ببقية الأيام، في حين تكون نسبة التأخير أقل في أيام أخرى مثل **الأحد** و**الجمعة**.
* يساعد هذا التحليل في التعرف على الأيام التي تتطلب تعزيزات إضافية لتحسين الخدمة.

**٢. توزيع أسعار التذاكر**

**Ticket Price Distribution**

* يمثل هذا الرسم التوزيع الإحصائي لأسعار التذاكر.
* نلاحظ أن معظم التذاكر تقع ضمن نطاقات سعرية منخفضة (0–50 تقريبًا)، مع وجود عدد قليل جدًا من التذاكر ذات الأسعار المرتفعة.
* يظهر منحنى الكثافة (KDE) فوق المدرج البياني مما يوضح اتجاه توزيع الأسعار بشكل أكثر سلاسة.
* يساعد هذا التحليل في فهم طبيعة التسعير ومدى تنوع الأسعار المقدمة للعملاء.

**٣. تحليل نسبة التأخير حسب ساعة المغادرة**

**Delay Percentage by Departure Hour**

* يعرض هذا الرسم العلاقة بين ساعة انطلاق الرحلة ونسبة حدوث التأخير.
* نلاحظ وجود بعض الساعات، مثل **الساعة 9 صباحًا** و**الساعة 12 ظهرًا**، ترتفع فيها نسبة التأخير مقارنة بساعات أخرى.
* هذا النوع من التحليل يفيد في تحسين جدولة القطارات عبر زيادة عدد القطارات أو تحسين أوقات الانطلاق في ساعات الذروة.

**٤. مقارنة سعر التذكرة حسب حالة الرحلة**

**Ticket Price vs Journey Status**

* باستخدام مخطط الصندوق (Boxplot)، يتم مقارنة توزيع أسعار التذاكر بناءً على حالة الرحلة:  
  **On Time** (في الموعد)، **Delayed** (متأخرة)، **Cancelled** (ملغاة).
* يوضح المخطط أن الرحلات المتأخرة عمومًا لها تذاكر بسعر متوسط أعلى قليلًا مقارنة بالرحلات في موعدها.
* كما يظهر وجود بعض القيم الشاذة (outliers) بأسعار مرتفعة في جميع الفئات.
* يساهم هذا التحليل في فحص ما إذا كان هناك علاقة بين الأسعار وجودة الخدمة.