

Team

Journy Of Data

Statistical Analysis of Emergency Cases by Region

Saudi Red Crescent 2023

Team Leader: Raghad Khalid



Team

Journy Of Data

Our Team

Writing & Report Presentation

Fatima Alrobiee

Bayan Majed

Nasaym Fahad

Data Cleaning in Python

Fatima Alrobiee

Nuha Murad

Statistical Analysis & Visualization in Python

Nuha Murad

Fatima Alrobiee

Raghad Khalid

Visualization & Dashboard in Power BI

Reyouf Faisal

Shahad Mahri

Raghad Khalid


Presentation & Delivery

Fatima Alrobiee

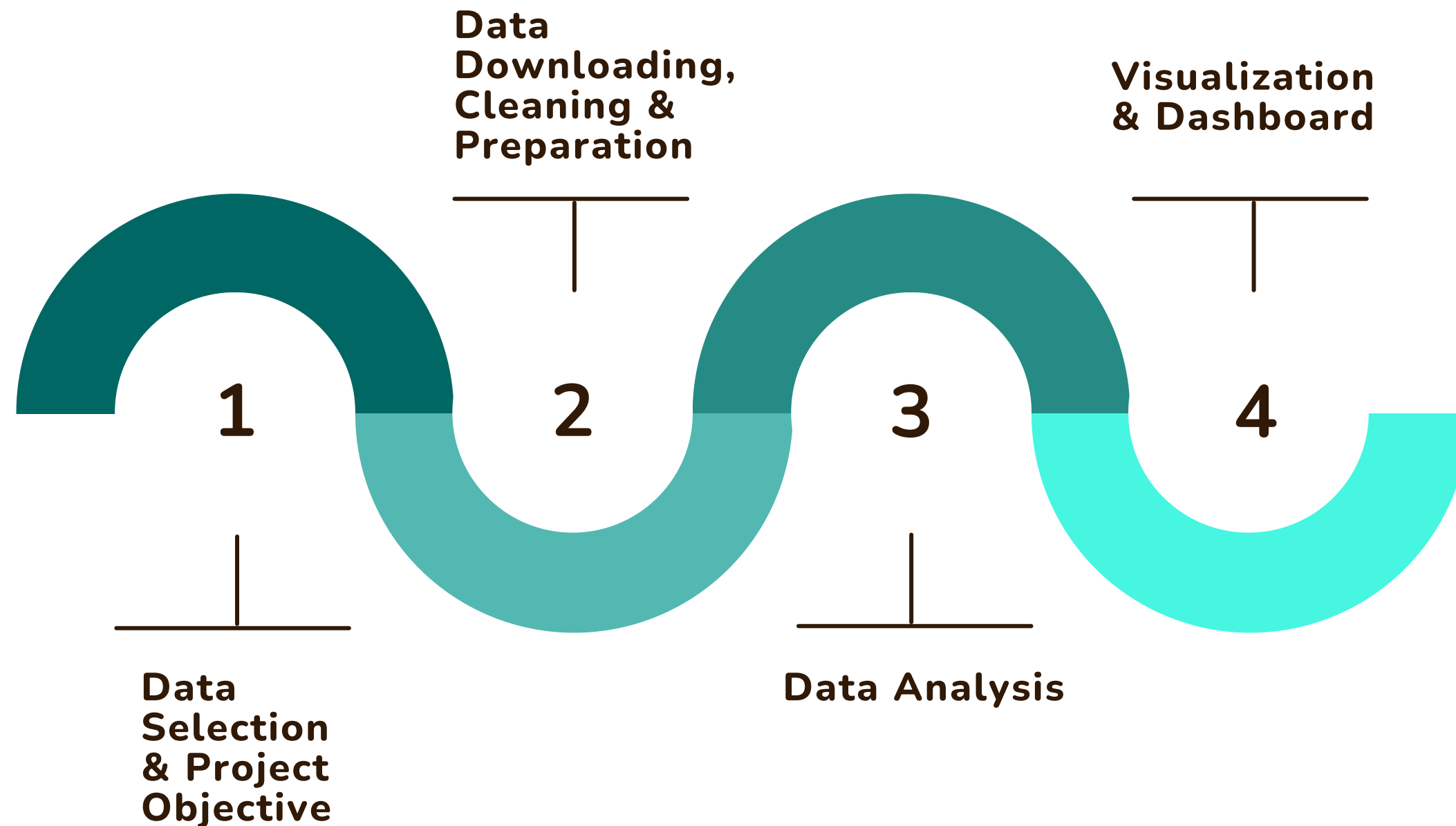
Raghad Khalid

Introductio

This project aims to analyze official Saudi data in order to extract key statistical indicators that support decision-making in healthcare and emergency services. **The data source** is the Saudi Red Crescent Authority – 2023, focusing on emergency patients and casualties transported to hospitals by case type and region. This analysis provides valuable insights to enhance resource allocation, identify high-risk areas, and improve overall emergency response strategies.



Project Workflow



Tools & Platforms Used



OPEN DATA PLATFORM
بوابة البيانات المفتوحة

colab



Power BI

Data Selection



Why did we choose this dataset?

We selected the dataset *"Patients and Casualties Transported by the Saudi Red Crescent Authority by Administrative Region and Case Type – 2023"*

from the **National Open Data Platform**. This dataset was chosen because it provides reliable and official information about emergency cases across Saudi Arabia. It enables us to analyze patterns by case type and administrative region, offering valuable insights into healthcare demand, emergency response distribution, and regional differences.

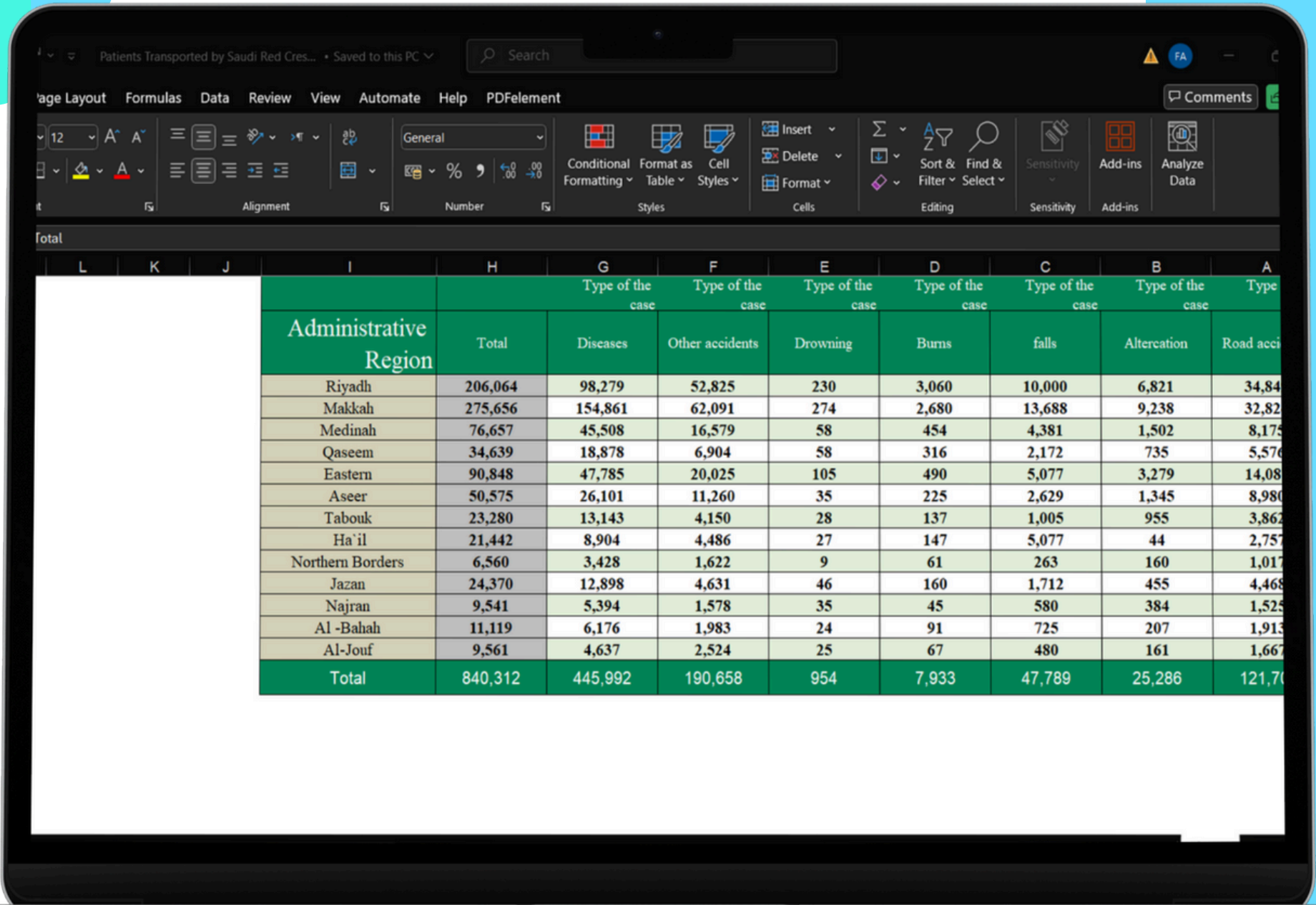
<https://open.data.gov.sa/ar/datasets/view/db3b6285-5ce6-4cd7-907d-5b8615d4ed0a>



What do we aim to achieve with this dataset ?

Project Objective

The objective of this project is to analyze the dataset of patients and casualties transported by the Saudi Red Crescent Authority to hospitals in 2023, categorized by administrative region and case type. This analysis aims to provide meaningful insights into the distribution of emergency cases across regions, identify common case types, highlight regional variations, and support better understanding of ambulance service demands.



Administrative Region	Total	Diseases	Other accidents	Drowning	Burns	falls	Altercation	Road acci
Riyadh	206,064	98,279	52,825	230	3,060	10,000	6,821	34,84
Makkah	275,656	154,861	62,091	274	2,680	13,688	9,238	32,82
Medinah	76,657	45,508	16,579	58	454	4,381	1,502	8,175
Qaseem	34,639	18,878	6,904	58	316	2,172	735	5,576
Eastern	90,848	47,785	20,025	105	490	5,077	3,279	14,08
Aseer	50,575	26,101	11,260	35	225	2,629	1,345	8,980
Tabouk	23,280	13,143	4,150	28	137	1,005	955	3,862
Ha'il	21,442	8,904	4,486	27	147	5,077	44	2,757
Northern Borders	6,560	3,428	1,622	9	61	263	160	1,017
Jazan	24,370	12,898	4,631	46	160	1,712	455	4,468
Najran	9,541	5,394	1,578	35	45	580	384	1,525
Al-Bahah	11,119	6,176	1,983	24	91	725	207	1,913
Al-Jouf	9,561	4,637	2,524	25	67	480	161	1,667
Total	840,312	445,992	190,658	954	7,933	47,789	25,286	121,70

Research Questions

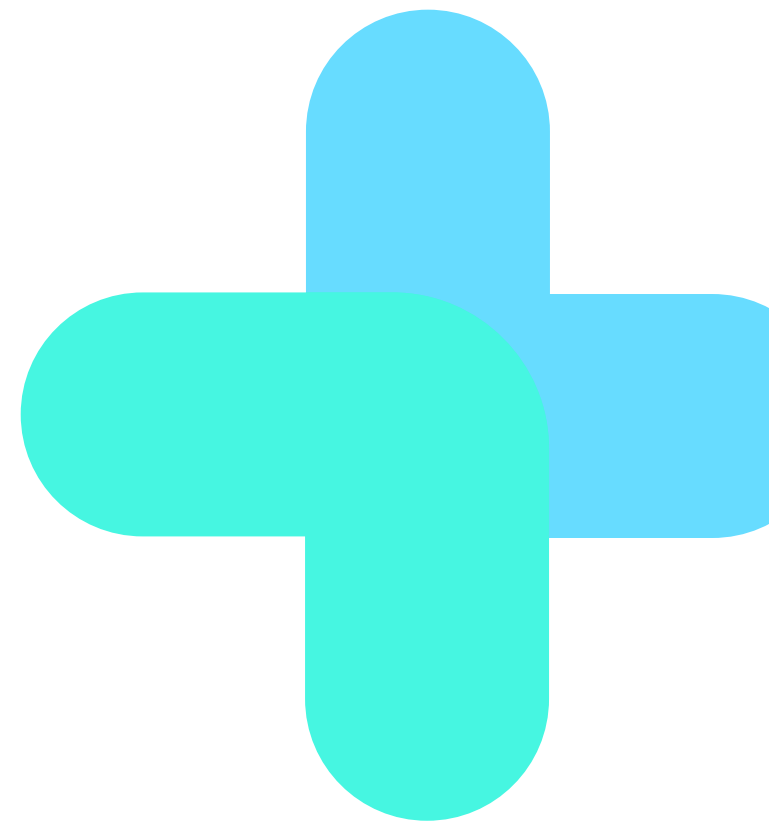
Key Question: What patterns, trends, and regional differences can be identified from the dataset to support better decision-making?

Descriptive Analysis

- Which administrative region recorded the highest number of transported cases in 2023?
- Which region recorded the lowest number of transported cases?
- What is the most common case type (e.g., medical, injury, accident)?
- What percentage of total transported cases does each region represent?

Comparative Analysis

- How does the distribution of case types vary across regions?
- Are there specific regions with higher injury-related cases compared to medical cases?
- What is the average number of transported cases per region?





Data Cleaning Steps:

1. Load CSV and use the second row as header
2. Fill missing values with 0
3. Strip extra spaces from column names
4. Rename columns with meaningful names
5. Convert numeric columns to integers (remove commas, ensure correct types)
6. Drop the last two rows (totals & empty rows)
7. Save cleaned dataset for future use
8. Reload the cleaned dataset
9. Generate summary statistics (describe)
10. Preview cleaned data

```
import pandas as pd

# بدل الأول Header قراءة الملف واستخدام الصف الثاني كـ
df = pd.read_csv('/content/Patients transported to hospitals by the SRCA.csv', header=1)

df.isnull().sum() # فحص عدد القيم المفقودة في كل عمود
df.fillna(0, inplace=True) # استبدال أي قيمة مفقودة بالعدد صفر

# تحديد الأعمدة الرقمية للتنظيف والتحويل
numeric_cols = ['Road accidents', 'Altercation', 'falls', 'Burns', 'Drowning', 'Other accidents', 'Diseases', 'Total']

# حذف الفواصل وتحويل القيم إلى أعداد صحيحة int
df[numeric_cols] = df[numeric_cols].replace(',', '', regex=True).astype(int)

df.columns = df.columns.str.strip() # تنظيف أسماء الأعمدة من أي مسافات زائدة

df = df.drop(df.index[-1]).reset_index(drop=True) # إعادة ترتيب الفهرسة (Total إجمالي) حذف الصف الأخير

df.to_csv('cleaned_red_crescent_data.csv', index=False) # حفظ نسخة نظيفة من الملف بعد التنظيف

df = pd.read_csv('/content/cleaned_red_crescent_data.csv') # قراءة النسخة النظيفة للتأكد

print(df.describe()) # عرض إحصائيات وصفية للأعمدة الرقمية بعد التنظيف
```




```
[ ] # أي منطقة إدارية سجلت أكبر/أقل عدد من الحالات المنقولة؟
max_area = df.loc[df['Total'] == df['Total'].max(), df.columns[-1]].values[0]
min_area = df.loc[df['Total'] == df['Total'].min(), df.columns[-1]].values[0]

print("أعلى عدد من الحالات في منطقة:", max_area)
print("أقل عدد من الحالات في منطقة:", min_area)
```

→ أعلى عدد من الحالات في منطقة: Makkah
 → أقل عدد من الحالات في منطقة: Northern Borders

```
[ ] # ما هو أكثر نوع حالة شيوعاً (مثل: طبية، إصابة، حادث)؟

# الأعمدة الرقمية بدون عمود التوتل
numeric_cols = df.select_dtypes(include='number').columns.drop('Total')

# مجموع كل نوع حالة على مستوى كل الداتا
type_sums = df[numeric_cols].sum()
most_common_type = type_sums.idxmax()
print("أكثر نوع حالة شيوعاً:", most_common_type)
```

→ أكثر نوع حالة شيوعاً: Diseases

```
[ ] # ما هي النسبة المئوية لكل منطقة من إجمالي الحالات المنقولة؟
total_cases = df['Total'].sum()
percentage_per_area = (df['Total'] / total_cases) * 100

result = pd.DataFrame({
    'المنطقة الإدارية': df[df.columns[-1]],
    'النسبة المئوية': percentage_per_area
})
print(result)
```

	المنطقة الإدارية	النسبة المئوية
0	Riyadh	24.522320
1	Makkah	32.804006
2	Medinah	9.122445
3	Qaseem	4.122159
4	Eastern	10.811222
5	Aseer	6.018598
6	Tabouk	2.770400
7	Ha`il	2.551671
8	Northern Borders	0.780662
9	Jazan	2.900113
10	Najran	1.135412
11	Al -Bahah	1.323199
12	Al-Jouf	1.137792





```
# لكل منطقة 'Most Common Case' إنشاء عمود
# العمود يحدد نوع الحالة الأكثر حدوثًا في كل منطقة
df2['Most Common Case'] = df2[numeric_cols].idxmax(axis=1)

# حساب النسبة المئوية لكل نوع حالة من إجمالي الحالات في نفس المنطقة
for case in numeric_cols:
    df2[case + ' %'] = (df2[case] / df2['Total']) * 100

# إنشاء جدول نهائي لعرض النتائج
result = df2[['Administrative Region', 'Most Common Case'] + [c + ' %' for c in numeric_cols]]
print(result)
```

	Administrative Region	Most Common Case	Road accidents %	Altercation %	\
0	Riyadh	Diseases	16.911736	3.310137	
1	Makkah	Diseases	11.907595	3.351278	
2	Medinah	Diseases	10.664388	1.959377	
3	Qaseem	Diseases	16.097462	2.121886	
4	Eastern	Diseases	15.506120	3.609325	
5	Aseer	Diseases	17.755808	2.659417	
6	Tabouk	Diseases	16.589347	4.102234	
7	Ha'il	Diseases	12.857942	0.205205	
8	Northern Borders	Diseases	15.503049	2.439024	
9	Jazan	Diseases	18.334017	1.867050	
10	Najran	Diseases	15.983650	4.024735	
11	Al -Bahah	Diseases	17.204785	1.861678	
12	Al-Jouf	Diseases	17.435415	1.683924	

	falls %	Burns %	Drowning %	Other accidents %	Diseases %
0	4.852861	1.484976	0.111616	25.635240	47.693435
1	4.965609	0.972226	0.099399	22.524814	56.179078
2	5.715068	0.592249	0.075662	21.627510	59.365746
3	6.270389	0.912267	0.167441	19.931291	54.499264
4	5.588455	0.539362	0.115578	22.042312	52.598846
5	5.198220	0.444884	0.069204	22.263964	51.608502
6	4.317010	0.588488	0.120275	17.826460	56.456186
7	23.677829	0.685570	0.125921	20.921556	41.525977
8	4.009146	0.929878	0.137195	24.725610	52.256098
9	7.025031	0.656545	0.188757	19.002872	52.925728
10	6.079027	0.471649	0.366838	16.539147	56.534954
11	6.520371	0.818419	0.215847	17.834338	55.544563
12	5.020395	0.700764	0.261479	26.398912	48.499111

Across all regions, the most common case type is Diseases.

```
# هل هناك مناطق معينة تسجل حالات إصابات أكثر مقارنة بالحالات الطبية؟

# تعريف الأعمدة التي تمثل الإصابات
injury_cols = ['Road accidents', 'Altercation', 'falls', 'Burns', 'Drowning', 'Other accidents']

# حساب مجموع الإصابات لكل منطقة
df2['Total Injuries'] = df2[injury_cols].sum(axis=1)

# مقارنة الإصابات مع الأمراض لكل منطقة
df2['Injuries > Diseases'] = df2['Total Injuries'] > df2['Diseases']

# عرض النتائج
result = df2[['Administrative Region', 'Total Injuries', 'Diseases', 'Injuries > Diseases']]
print(result)

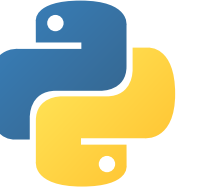
# تحديد المناطق التي تسجل إصابات أكثر من الحالات الطبية
high_injury_areas = df2[df2['Injuries > Diseases'] == True]['Administrative Region'].tolist()
print("\nالمناطق التي تسجل إصابات أكثر من الحالات الطبية:", high_injury_areas)
```

	Administrative Region	Total Injuries	Diseases	Injuries > Diseases
0	Riyadh	107785	98279	True
1	Makkah	120795	154861	False
2	Medinah	31149	45508	False
3	Qaseem	15761	18878	False
4	Eastern	43063	47785	False
5	Aseer	24474	26101	False
6	Tabouk	10137	13143	False
7	Ha'il	12538	8904	True
8	Northern Borders	3132	3428	False
9	Jazan	11472	12898	False
10	Najran	4147	5394	False
11	Al -Bahah	4943	6176	False
12	Al-Jouf	4924	4637	True

True → The number of injuries is greater than the number of diseases in that region.
False → The number of injuries is less than or equal to the number of diseases in that region.

المناطق التي تسجل إصابات أكثر من الحالات الطبية: ['Riyadh', 'Ha'il', 'Al-Jouf']





```
# مامتوسط عدد الحالات المنقولة لكل منطقة ؟  
  
# أول شي نحدد الأعمدة الي نحتاجها  
cols = ['Road accidents', 'Altercation', 'falls', 'Burns', 'Drowning', 'Other accidents', 'Diseases']  
  
# حساب المتوسط لكل منطقة  
df2['Average Cases'] = df2[cols].mean(axis=1)  
  
# عرض النتيجة  
df2[['Administrative Region', 'Average Cases']]
```

	Administrative Region	Average Cases
0	Riyadh	29437.714286
1	Makkah	39379.428571
2	Medinah	10951.000000
3	Qaseem	4948.428571
4	Eastern	12978.285714
5	Aseer	7225.000000
6	Tabouk	3325.714286
7	Ha'il	3063.142857
8	Northern Borders	937.142857
9	Jazan	3481.428571
10	Najran	1363.000000
11	Al -Bahah	1588.428571
12	Al-Jouf	1365.857143

Results highlight:

- Makkah: highest average (~39,379 cases)
- Riyadh: ~29,438 cases
- Northern Borders: lowest average (~937 cases)





Reorder the columns and save as an Excel file
to be ready for Power BI for visualization and dashboard creation.



```
ordered_cols = [  
    'Administrative Region', 'Total', 'Total Injuries',  
    'Road accidents', 'Altercation', 'falls', 'Burns', 'Drowning', 'Other accidents', 'Diseases',  
    'Road accidents %', 'Altercation %', 'falls %', 'Burns %', 'Drowning %', 'Other accidents %', 'Diseases %'  
]  
  
df2[ordered_cols]=df2[ordered_cols].round(2) # إعادة ترتيب الأعمدة  
  
df2.to_excel('SRCA_Analysis.xlsx', index=False)
```

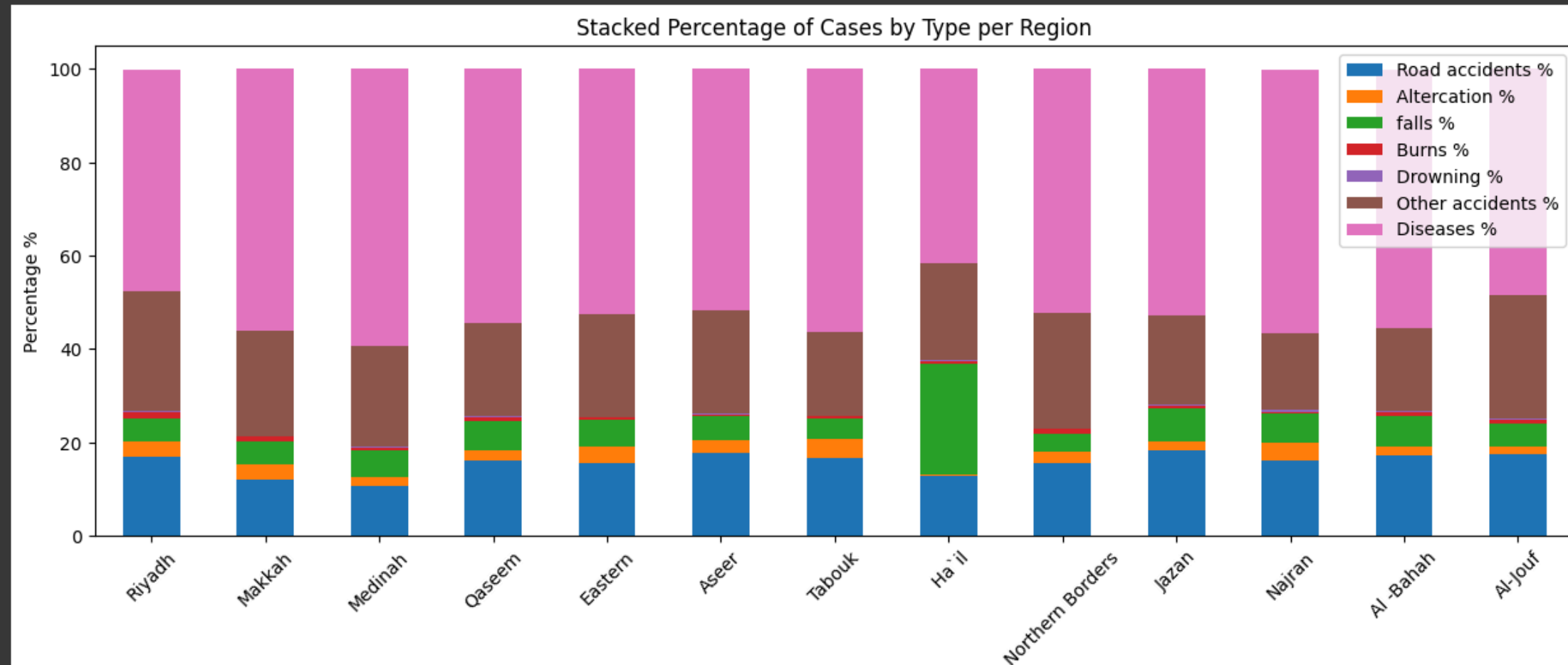
The file format after analysis



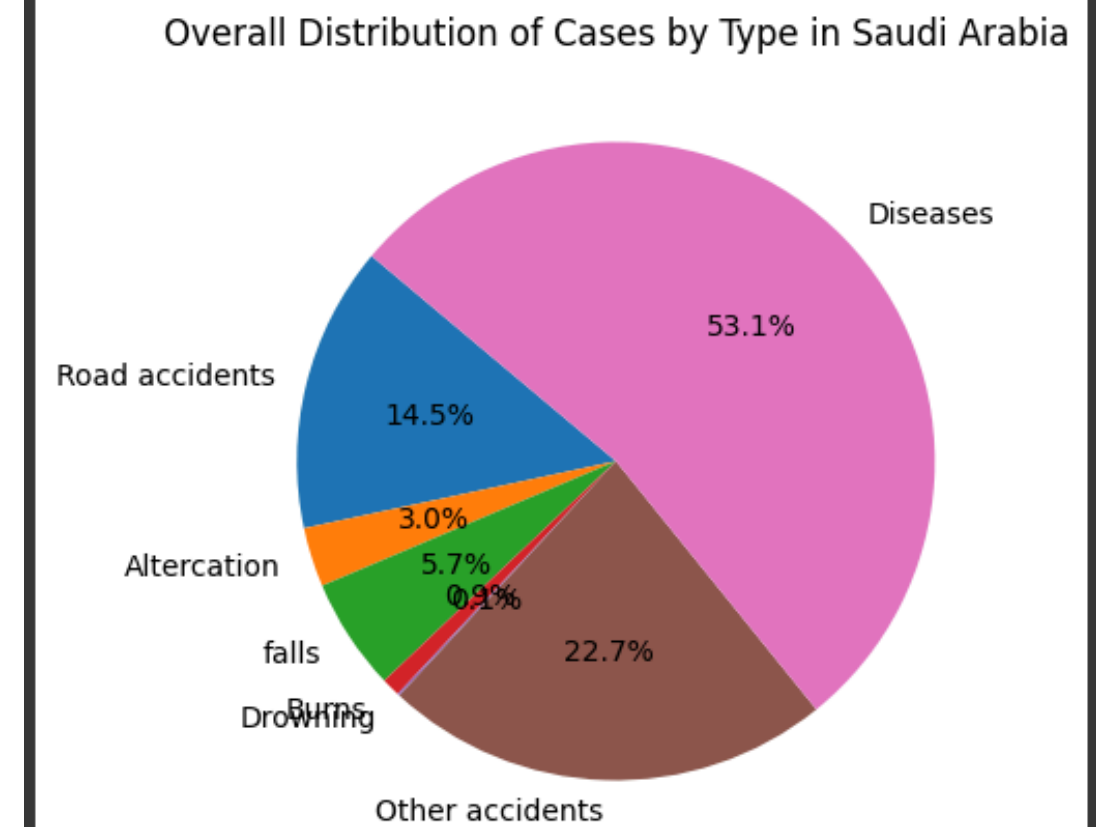
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
Road accidents	Altercation	falls	Burns	Drowning	Other accidents	Diseases	Total	Administrative Region	Most Common Case	Road accidents %	Altercation %	falls %	Burns %	Drowning %	Other accidents %	Diseases %	Total Injuries	Injuries > Diseases
34849	6821	10000	3060	230	52825	98279	206064	Riyadh	Diseases	16.91	3.31	4.85	1.48	0.11	25.64	47.69	107785	TRUE
32824	9238	13688	2680	274	62091	154861	275656	Makkah	Diseases	11.91	3.35	4.97	0.97	0.1	22.52	56.18	120795	FALSE
8175	1502	4381	454	58	16579	45508	76657	Medinah	Diseases	10.66	1.96	5.72	0.59	0.08	21.63	59.37	31149	FALSE
5576	735	2172	316	58	6904	18878	34639	Qaseem	Diseases	16.1	2.12	6.27	0.91	0.17	19.93	54.5	15761	FALSE
14087	3279	5077	490	105	20025	47785	90848	Eastern	Diseases	15.51	3.61	5.59	0.54	0.12	22.04	52.6	43063	FALSE
8980	1345	2629	225	35	11260	26101	50575	Aseer	Diseases	17.76	2.66	5.2	0.44	0.07	22.26	51.61	24474	FALSE
3862	955	1005	137	28	4150	13143	23280	Tabouk	Diseases	16.59	4.1	4.32	0.59	0.12	17.83	56.46	10137	FALSE
2757	44	5077	147	27	4486	8904	21442	Ha'il	Diseases	12.86	0.21	23.68	0.69	0.13	20.92	41.53	12538	TRUE
1017	160	263	61	9	1622	3428	6560	Northern Borders	Diseases	15.5	2.44	4.01	0.93	0.14	24.73	52.26	3132	FALSE
4468	455	1712	160	46	4631	12898	24370	Jazan	Diseases	18.33	1.87	7.03	0.66	0.19	19	52.93	11472	FALSE
1525	384	580	45	35	1578	5394	9541	Najran	Diseases	15.98	4.02	6.08	0.47	0.37	16.54	56.53	4147	FALSE
1913	207	725	91	24	1983	6176	11119	Al -Bahah	Diseases	17.2	1.86	6.52	0.82	0.22	17.83	55.54	4943	FALSE
1667	161	480	67	25	2524	4637	9561	Al-Jouf	Diseases	17.44	1.68	5.02	0.7	0.26	26.4	48.5	4924	TRUE



```
# توزيع نسب كل نوع حادث لكل منطقة
percent_cols = ['Road accidents %', 'Altercation %', 'falls %', 'Burns %', 'Drowning %', 'Other accidents %', 'Diseases %']
df_percent = df.set_index('Administrative Region')[percent_cols]
df_percent.plot(kind='bar', stacked=True, figsize=(15,5))
plt.title('Stacked Percentage of Cases by Type per Region')
plt.ylabel('Percentage %')
plt.xticks(rotation=45)
plt.show()
```

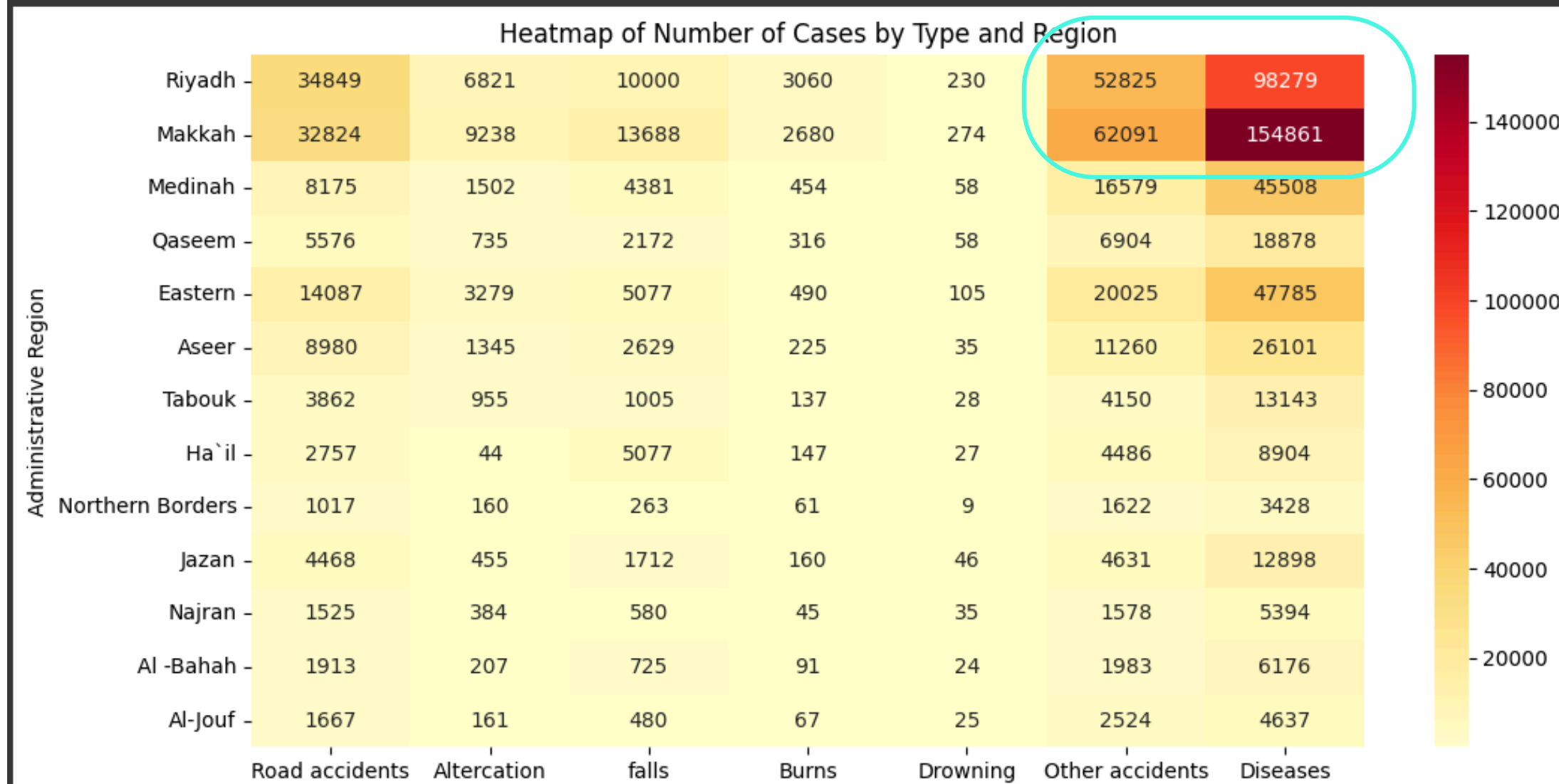


```
# التوزيع لكل نوع حادث على مستوى المملكة
total_cases = df[accident_columns].sum()
plt.figure(figsize=(5,5))
plt.pie(total_cases, labels=accident_columns, autopct='%1.1f%%', startangle=140)
plt.title('Overall Distribution of Cases by Type in Saudi Arabia')
plt.show()
```



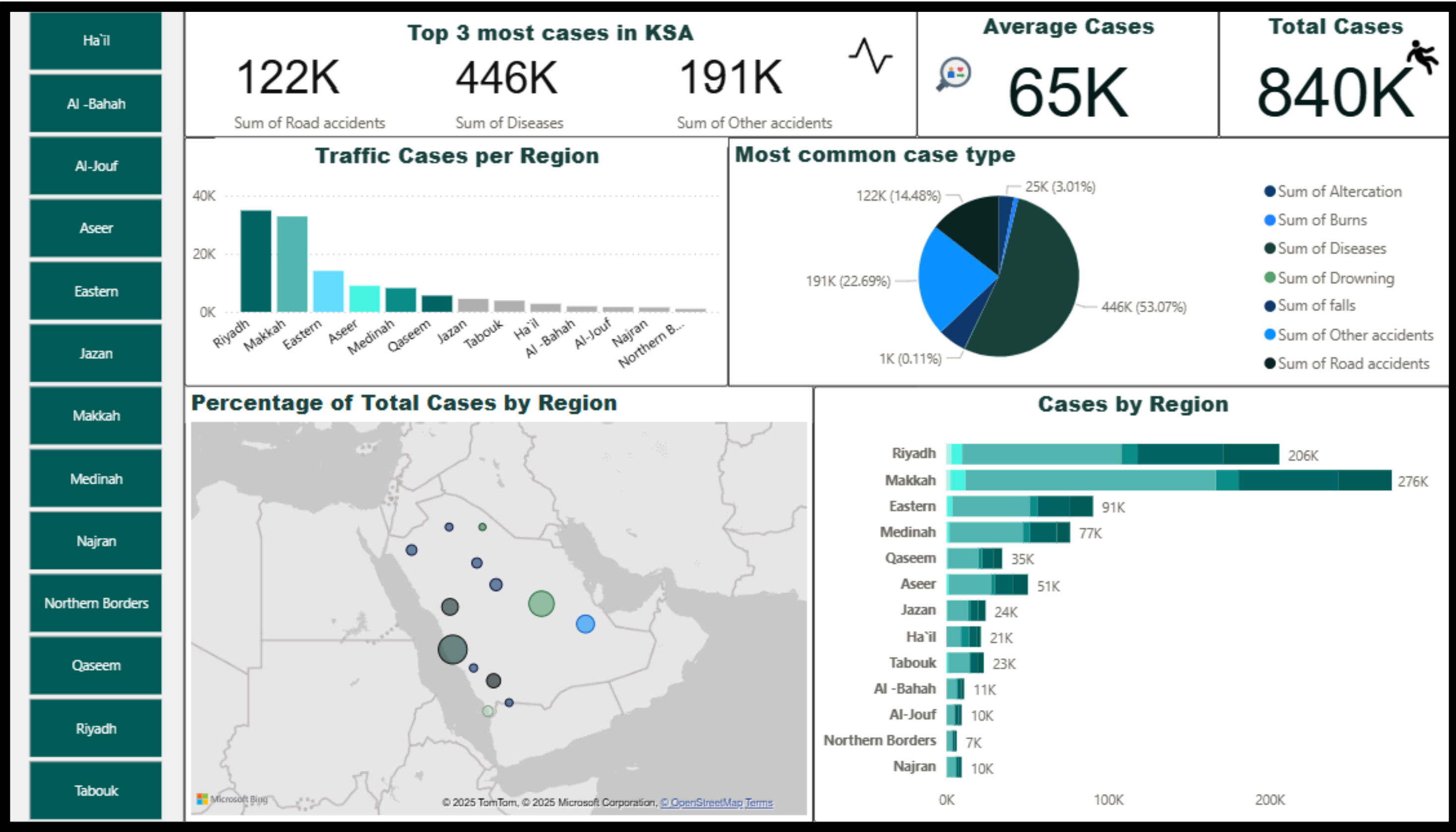


```
#أعلى المناطق لكل نوع حادث  
plt.figure(figsize=(12,6))  
sns.heatmap(df[accident_columns].set_index(df['Administrative Region']), annot=True, fmt='d', cmap='YlOrRd')  
plt.title('Heatmap of Number of Cases by Type and Region')  
plt.show()
```

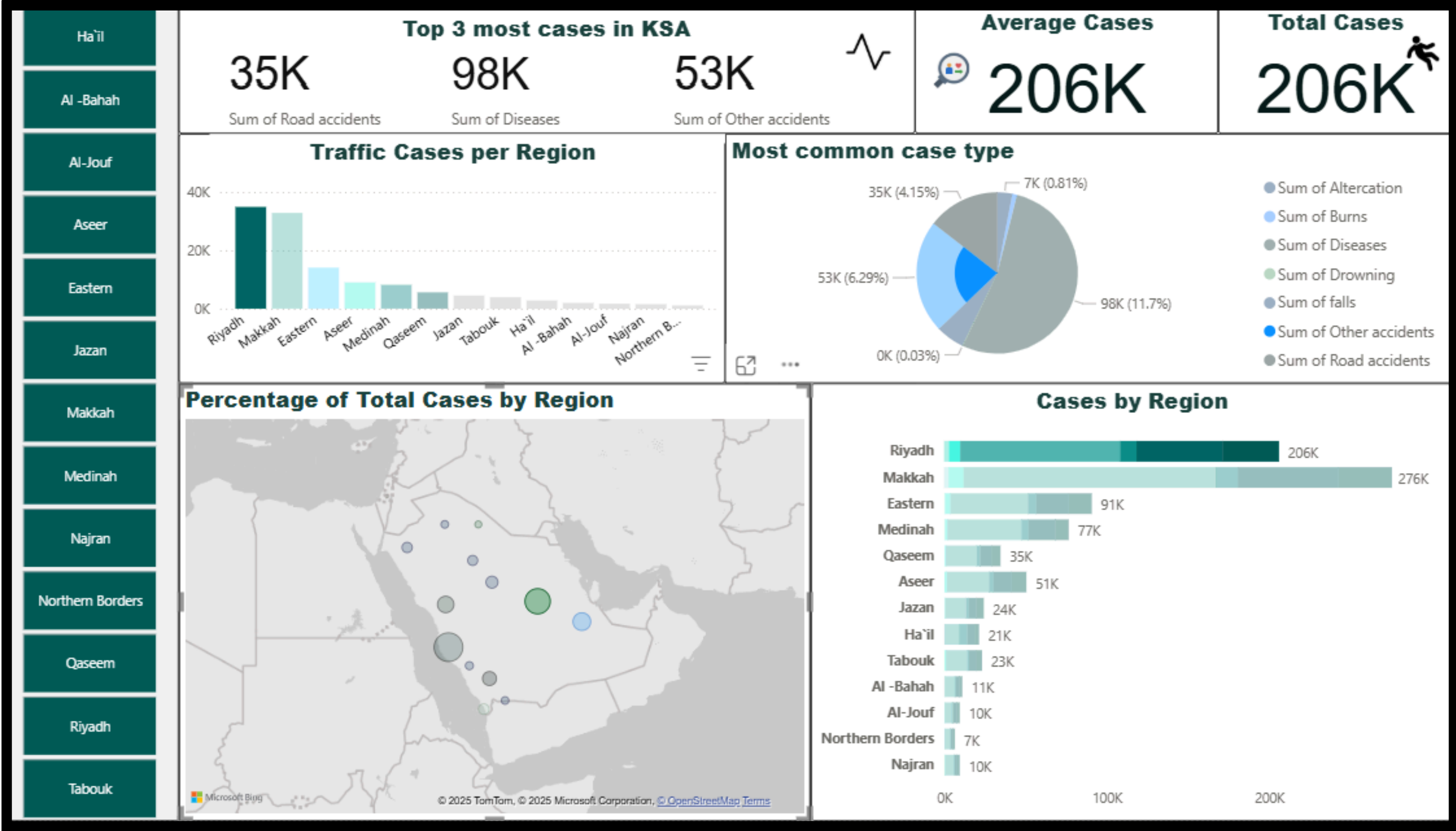




Data Visualizations in Power BI



Data Visualizations in Power BI



Future Work

- Integrate external data, such as the population of each administrative region in 2023, to calculate the proportion of emergency cases relative to population.
- Consider whether the data represents only residents or includes visitors/foreigners:
 - Option 1: Exclude visitors and calculate percentages based only on residents.
 - Option 2: Include an estimate of visitors in the population to get a more accurate per capita rate.
- Perform comparative analysis to identify regions with higher emergency rates per capita.
- Use these insights to improve resource allocation and emergency response planning.

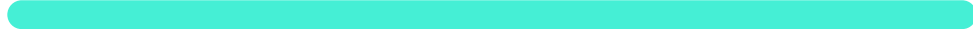


Resources

Statistical report



Conclusion

- Diseases are the most common emergency cases in Saudi Arabia.
 - Riyadh and Makkah have the highest number of transported cases; Northern Borders and Najran the lowest.
 - Regional differences highlight areas with higher emergency healthcare demand.
 - Cleaned data and visualizations provide insights for better resource allocation and decision-making.
- 

Team & Contact Information



Raghad Khalid

raghadkhalid797@gmail.com

Fatima Alrobiee

ftem41@gmail.com

Nuha Murad

moradnoha3@gmail.com

Reyouf Faisal

Reyouffaisal77@gmail.com

Shahad Mahri

mahrishahad51@gmail.com

Bayan Majed

Bayan.bakhsh1@gmail.com

Nasaym Fahad

nasaymfahad@gmail.com

Acknowledgements and Special Thanks

We would like to extend our sincere thanks to the **PREPX team** for providing this amazing training program. We are also grateful to all the trainers for their continuous guidance and support throughout the program. A special thank you goes to **Trainer Mohammad Sharif** for his ongoing assistance and invaluable recommendations, which have been truly priceless in helping us successfully complete this project.





تواصل معنا اليوم عبر

 prepxprograms.com

 info@prepxprograms.com

 [prepx2030](https://www.linkedin.com/company/prepx2030)

  [prepx_2030](https://www.instagram.com/prepx_2030)

Team

Journy Of Data

PREP

Statistical Analysis of Emergency Cases by Region

Saudi Red Crescent 2023

