

```
In [ ]: # Imports necesarios.
import matplotlib.pyplot as plt
import seaborn as sns
import pandas as pd
```

```
In [ ]: # Base de datos Local
data = sns.load_dataset("car_crashes")
data.head() # Cabecera de Los datos.
```

```
Out[ ]:
```

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	abbrev
0	18.8	7.332	5.640	18.048	15.040	784.55	145.08	AL
1	18.1	7.421	4.525	16.290	17.014	1053.48	133.93	AK
2	18.6	6.510	5.208	15.624	17.856	899.47	110.35	AZ
3	22.4	4.032	5.824	21.056	21.280	827.34	142.39	AR
4	12.0	4.200	3.360	10.920	10.680	878.41	165.63	CA

### Pregunta 1

Realice un **filtro** al conjunto de datos para incluir solo aquellos estados en donde la tasa total de accidentes (variable "total") es mayor que la media de todos los estados.

```
In [ ]: media_total_accidentes = data["total"].mean() # Calcular la media de la variable "T
```

```
In [ ]: filtro1 = data[data["total"] > media_total_accidentes] # Filtrar todos los valores
filtro1
```

Out[ ]:

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	abbrev
0	18.8	7.332	5.640	18.048	15.040	784.55	145.08	AL
1	18.1	7.421	4.525	16.290	17.014	1053.48	133.93	AK
2	18.6	6.510	5.208	15.624	17.856	899.47	110.35	AZ
3	22.4	4.032	5.824	21.056	21.280	827.34	142.39	AR
7	16.2	6.156	4.860	14.094	16.038	1137.87	151.48	DE
9	17.9	3.759	5.191	16.468	16.826	1160.13	144.18	FL
11	17.5	9.450	7.175	14.350	15.225	861.18	120.92	HI
16	17.8	4.806	4.272	13.706	15.130	780.45	133.80	KS
17	21.4	4.066	4.922	16.692	16.264	872.51	137.13	KY
18	20.5	7.175	6.765	14.965	20.090	1281.55	194.78	LA
24	17.6	2.640	5.456	1.760	17.600	896.07	155.77	MS
25	16.1	6.923	5.474	14.812	13.524	790.32	144.45	MO
26	21.4	8.346	9.416	17.976	18.190	816.21	85.15	MT
31	18.4	3.496	4.968	12.328	18.032	869.85	120.75	NM
33	16.8	6.552	5.208	15.792	13.608	708.24	127.82	NC
34	23.9	5.497	10.038	23.661	20.554	688.75	109.72	ND
36	19.9	6.368	5.771	18.308	18.706	881.51	178.86	OK
38	18.2	9.100	5.642	17.472	16.016	905.99	153.86	PA
40	23.9	9.082	9.799	22.944	19.359	858.97	116.29	SC
41	19.4	6.014	6.402	19.012	16.684	669.31	96.87	SD
42	19.5	4.095	5.655	15.990	15.795	767.91	155.57	TN
43	19.4	7.760	7.372	17.654	16.878	1004.75	156.83	TX
48	23.8	8.092	6.664	23.086	20.706	992.61	152.56	WV
50	17.4	7.308	5.568	14.094	15.660	791.14	122.04	WY



Una vez realizada dicha acción responda: ¿Cuál es la **media de accidentes** en donde el **alcohol** fue un factor para los estados filtrados?

```
In [ ]: preg1 = filtro1["alcohol"].mean()
preg1
```

Out[ ]: 6.1589583333333335

## Pregunta 2

Realice un filtro al conjunto de datos para seleccionar los **estados** en donde la tasa de accidentes por exceso de velocidad ("speeding") es menor que el primer cuartil de todas las tasas de "speeding".

```
In [ ]: # Cuartil 1 de la variable "speeding"
cuartil1 = data["speeding"].quantile(0.25)
cuartil1
```

```
Out[ ]: 3.7664999999999997
```

```
In [ ]: filtro2 = data[data["speeding"]<cuartil1]
filtro2
```

```
Out[ ]:
```

	total	speeding	alcohol	not_distracted	no_previous	ins_premium	ins_losses	abbrev
8	5.9	2.006	1.593	5.900	5.900	1273.89	136.05	DC
9	17.9	3.759	5.191	16.468	16.826	1160.13	144.18	FL
10	15.6	2.964	3.900	14.820	14.508	913.15	142.80	GA
14	14.5	3.625	4.205	13.775	13.775	710.46	108.92	IN
15	15.7	2.669	3.925	15.229	13.659	649.06	114.47	IA
21	8.2	1.886	2.870	7.134	6.560	1011.14	135.63	MA
22	14.1	3.384	3.948	13.395	10.857	1110.61	152.26	MI
23	9.6	2.208	2.784	8.448	8.448	777.18	133.35	MN
24	17.6	2.640	5.456	1.760	17.600	896.07	155.77	MS
27	14.9	1.937	5.215	13.857	13.410	732.28	114.82	NE
30	11.2	1.792	3.136	9.632	8.736	1301.52	159.85	NJ
31	18.4	3.496	4.968	12.328	18.032	869.85	120.75	NM
46	12.7	2.413	3.429	11.049	11.176	768.95	153.72	VA

**Pregunta 2** Utilizando el filtro anterior responda ¿Cuál es la mediana de las primas de seguro (ins\_premium) para estos estados.

```
In [ ]: preg2 = filtro2["ins_premium"].median()
preg2
```

```
Out[ ]: 896.07
```

**Pregunta 3** Crea una nueva variable que categorice los estados como "Alcohol Alto" o "Alcohol Bajo" según si la tasa de accidentes relacionados con **alcohol** está por encima o

por debajo de la mediana.


```
In [ ]: mediana_alcohol = data["alcohol"].median()
```

```
In [ ]: def asignar_rango(alcohol):  
        if alcohol > mediana_alcohol:  
            return "Alto_Alcohol"  
        else:  
            return "Bajo_Alcohol"
```

En base a esto responde ¿Cuál es la desviación estándar de las pérdidas aseguradas ("ins\_losses") en la categoría "Alcohol Alto"?

```
In [ ]: data["alcohol_rango"] = data["alcohol"].apply(asignar_rango)  
data.head()
```

```
Out[ ]:    total  speeding  alcohol  not_distracted  no_previous  ins_premium  ins_losses  abbrev  
0    18.8      7.332    5.640         18.048         15.040         784.55      145.08      AL  
1    18.1      7.421    4.525         16.290         17.014        1053.48      133.93      AK  
2    18.6      6.510    5.208         15.624         17.856         899.47      110.35      AZ  
3    22.4      4.032    5.824         21.056         21.280         827.34      142.39      AR  
4    12.0      4.200    3.360         10.920         10.680         878.41      165.63      CA
```



```
In [ ]: data.groupby("alcohol_rango")["ins_losses"].std() #std() es la desviación estandar.
```

```
Out[ ]: alcohol_rango  
Alto_Alcohol    24.689203  
Bajo_Alcohol    25.325266  
Name: ins_losses, dtype: float64
```

```
In [ ]: print(f"Respuesta: Alto_Alcohol = {round(24.689203,2)}")
```

Respuesta: Alto\_Alcohol = 24.69

**Pregunta 4** Cargar una archivo en formato png con el histograma de la variable "total".

```
In [ ]: sns.histplot(data = data, x="total")  
plt.title("Histograma de la variable Total")  
plt.savefig("histograma.png", dpi=300)  
plt.show()
```

Histograma de la variable Total

