

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

```
In [ ]: # Base de datos CSV
data = pd.read_csv("base_datos.csv")
data.head() # Cabecera de Los datos.
```

```
Out[ ]: Profundidad_Perforacion  Tiempo_Perforacion  Costo_Operacion  Concentracion_Cobre  Num_Trabajadores  Tipo_Maquinaria  Grupo_Turno
0          97.72          63.44          5298.69          34.23          22.0      Maquinaria_A  Turno_Noch
1          117.64          64.76          5697.33          33.90          25.0      Maquinaria_B  Turno_Noch
2           97.86          56.05          5225.78          27.79          24.0      Maquinaria_A  Turno_Noch
3          119.94          56.35          5198.99          33.67          28.0      Maquinaria_B  Turno_Noch
4          119.49          63.92          5866.86          38.16          24.0      Maquinaria_B  Turno_Mañan
```

```
In [ ]: #media tiempo perforacion2 decimales.
media_tp = data["Tiempo_Perforacion"].mean().round(2)
media_tp
```

```
Out[ ]: 52.58
```

```
In [ ]: #Media numero trabajadores2 decimales.
media_nt = data["Num_Trabajadores"].mean().round(2)
media_nt
```

```
Out[ ]: 22.73
```

```
In [ ]: #desviación estandar
desviacion_tp = data["Tiempo_Perforacion"].std().round(2)
desviacion_tp
```

```
Out[ ]: 6.45
```

```
In [ ]: #desviación estandar numero trabajadores
desviacion_nt = data["Num_Trabajadores"].std().round(2)
desviacion_nt
```

```
Out[ ]: 3.93
```

```
In [ ]: #cuartil 1 tiempo perforacion
cuartil1_tp = data["Tiempo_Perforacion"].quantile(0.25).round(2)
cuartil1_tp
```

```
Out[ ]: 47.88
```

```
In [ ]: #cuartil 3 numero trabajadores
cuartil3_nt = data["Num_Trabajadores"].quantile(0.75)
cuartil3_nt
```

```
Out[ ]: 26.0
```

Problema 1

Filtra los datos para incluir solo los registros del 'Turno_Noche' y describe las estadísticas de 'Tiempo_Perforacion' y 'Num_Trabajadores' para este turno.

```
In [ ]: # Datos filtrados solo turno noche.
lista_turno_noche = data[data["Grupo_Turno"] == "Turno_Noche"]
lista_turno_noche
```

Out[]:	Profundidad_Perforacion	Tiempo_Perforacion	Costo_Operacion	Concentracion_Cobre	Num_Trabajadores	Tipo_Maquinaria	Grupo_Tur
	0	97.72	63.44	5298.69	34.23	22.0	Maquinaria_A Turno_Noc
	1	117.64	64.76	5697.33	33.90	25.0	Maquinaria_B Turno_Noc
	2	97.86	56.05	5225.78	27.79	24.0	Maquinaria_A Turno_Noc
	3	119.94	56.35	5198.99	33.67	28.0	Maquinaria_B Turno_Noc
	5	110.20	52.04	4800.02	34.47	21.0	Maquinaria_C Turno_Noc

	993	110.07	49.99	5156.25	31.79	18.0	Maquinaria_C Turno_Noc
	994	101.74	46.81	4852.90	37.20	18.0	Maquinaria_C Turno_Noc
	997	85.71	59.66	4569.57	29.96	20.0	Maquinaria_A Turno_Noc
	998	91.63	42.16	5113.77	30.29	25.0	Maquinaria_C Turno_Noc
	999	94.25	42.12	5137.53	33.90	18.0	Maquinaria_C Turno_Noc

519 rows × 7 columns

```
In [ ]: # Descripción de estadísticas de Tiempo_Perforacion para lista_turno_noche.
lista_turno_noche["Tiempo_Perforacion"].describe().round(2)
```

```
Out[ ]: count    519.00
mean      52.72
std        6.58
min        35.59
25%       48.12
50%       53.32
75%       57.18
max        69.85
Name: Tiempo_Perforacion, dtype: float64
```

```
In [ ]: # Descripción de estadísticas de Num_Trabajadores para lista_turno_noche.
lista_turno_noche["Num_Trabajadores"].describe().round(2)
```

```
Out[ ]: count    519.00
mean     22.58
std       4.07
min       12.42
25%       20.00
50%       23.00
75%       25.00
max       32.45
Name: Num_Trabajadores, dtype: float64
```

Problema 2

Crea una variable categórica 'Profundidad_Categorizada' que clasifique la 'Profundidad_Perforacion' en 'Baja', 'Media' y 'Alta' considerando que la clasificación media acumula el **40%** de los datos centrales. Después, describe la variable 'Tiempo_Perforacion' para cada una de estas nuevas categorías.

```
In [ ]: #Creación de La variable 'Profundidad_Perforacion'.
q30, q70 = data['Profundidad_Perforacion'].quantile([0.3, 0.7])

def categorizarProfundidad(valor):
    if valor < q30:
        return "Baja"
    elif valor > q70:
        return "Alta"
    else:
        return "Media"

data["Profundidad_Categorizada"] = data["Profundidad_Perforacion"].apply(categorizarProfundidad)
data.head()
```

```
Out[ ]:
```

	Profundidad_Perforacion	Tiempo_Perforacion	Costo_Operacion	Concentracion_Cobre	Num_Trabajadores	Tipo_Maquinaria	Grupo_Turn
0	97.72	63.44	5298.69	34.23	22.0	Maquinaria_A	Turno_Noch
1	117.64	64.76	5697.33	33.90	25.0	Maquinaria_B	Turno_Noch
2	97.86	56.05	5225.78	27.79	24.0	Maquinaria_A	Turno_Noch
3	119.94	56.35	5198.99	33.67	28.0	Maquinaria_B	Turno_Noch
4	119.49	63.92	5866.86	38.16	24.0	Maquinaria_B	Turno_Mañan

```
In [ ]: # Descripción de la variable Tiempo_Perforacion para cada una de las categorías de Profundidad_Categorizada.
data.groupby("Profundidad_Categorizada")["Tiempo_Perforacion"].describe().round(2)
```

```
Out[ ]:
```

	count	mean	std	min	25%	50%	75%	max
Profundidad_Categorizada								
Alta	300.0	54.35	6.77	35.78	50.52	54.76	60.02	68.33
Baja	300.0	51.56	5.39	35.59	47.49	52.00	55.72	69.85
Media	400.0	52.03	6.69	34.98	47.04	52.40	56.28	70.95

```
In [ ]: # Tiempo de perforación medio para profundidades bajas.
data[data["Profundidad_Categorizada"] == "Baja"]["Tiempo_Perforacion"].mean().round(2)
```

```
Out[ ]: 51.56
```

```
In [ ]: # Mediana de tiempo de perforación para profundidades bajas.
data[data["Profundidad_Categorizada"] == "Baja"]["Tiempo_Perforacion"].median().round(2)
```

```
Out[ ]: 52.0
```

```
In [ ]: # Tiempo de perforación medio para profundidades medias.
data[data["Profundidad_Categorizada"] == "Media"]["Tiempo_Perforacion"].mean().round(2)
```

```
Out[ ]: 52.03
```

```
In [ ]: # Mediana de tiempo de perforación para profundidades medias.
data[data["Profundidad_Categorizada"] == "Media"]["Tiempo_Perforacion"].median().round(2)
```

```
Out[ ]: 52.4
```

```
In [ ]: # Desviación estandar de tiempo de perforación para profundidades media.
data[data["Profundidad_Categorizada"] == "Media"]["Tiempo_Perforacion"].std().round(2)
```

```
Out[ ]: 6.69
```

```
In [ ]: # Máximo de tiempo de perforación para profundidades alta.
data[data["Profundidad_Categorizada"] == "Alta"]["Tiempo_Perforacion"].max().round(2)
```

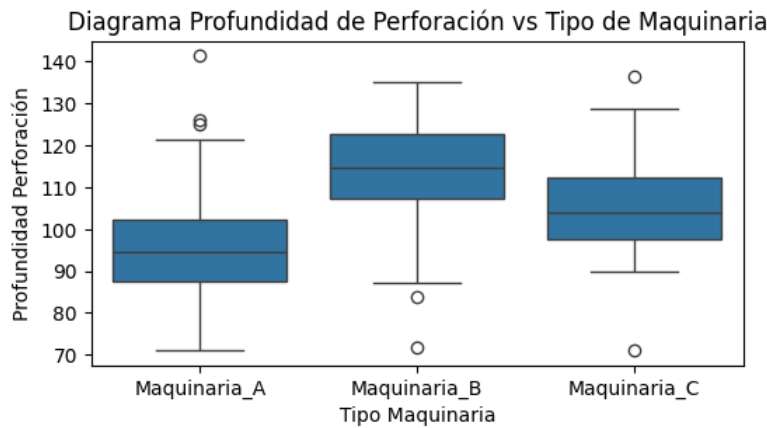
```
Out[ ]: 68.33
```

Problema 3

Genera un diagrama de cajas para comparar la 'Profundidad_Perforacion' en cada tipo de 'Tipo_Maquinaria' (Maquinaria_A, Maquinaria_B, Maquinaria_C).

- ¿Qué tipo de maquinaria tiende a ser utilizada para perforaciones más profundas?.
- Identifique la máquina que cuenta con el mayor número de valores atípicos.
- Genere un archivo con la gráfica diagrama de cajas.

```
In [ ]: plt.figure(figsize=(6,3))
sns.boxplot(data=data, y="Profundidad_Perforacion", x="Tipo_Maquinaria")
plt.title("Diagrama Profundidad de Perforación vs Tipo de Maquinaria")
plt.ylabel("Profundidad Perforación")
plt.xlabel("Tipo Maquinaria")
plt.savefig("diagrama-profundidad-tipo-maquinaria.png")
plt.show()
```



¿Qué tipo de maquinaria tiende a ser utilizada para perforaciones más profundas?

```
In [ ]: # Tipo de maquinaria para perforaciones más profundas.
media_tm = data.groupby('Tipo_Maquinaria')['Profundidad_Perforacion'].mean()
nameTipo = media_tm.idxmax()
print("El tipo de maquinaria que tiende a ser utilizada para perforaciones más profundas es :", nameTipo)
```

El tipo de maquinaria que tiende a ser utilizada para perforaciones más profundas es : Maquinaria_B

Identifique la máquina que cuenta con el mayor número de valores atípicos.

```
In [ ]: q1 = data["Profundidad_Perforacion"].quantile(0.25)
q3 = data["Profundidad_Perforacion"].quantile(0.75)

Lim_inf = q1-1.5*(q3-q1)
Lim_sup = q3+1.5*(q3-q1)

Lim_inf, Lim_sup
```

Out[]: (68.6875000000003, 140.48749999999998)

```
In [ ]: atipicos = data[(data["Profundidad_Perforacion"] < Lim_inf) | (data["Profundidad_Perforacion"] > Lim_sup)]
atipicos
```

```
Out[ ]:   Profundidad_Perforacion  Tiempo_Perforacion  Costo_Operacion  Concentracion_Cobre  Num_Trabajadores  Tipo_Maquinaria  Grupo_Tu
986                141.48                48.78        4829.53                31.99                22.0        Maquinaria_A  Turno_Mañ
```

```
In [ ]: def identificar_atipicos(data):
    nameProfundidad = "Profundidad_Perforacion"
    nameTipo = "Tipo_Maquinaria"
    atipicos = []
    for tipo in data[nameTipo].unique():
        data_filtrada = data[data[nameTipo] == tipo]
        q1 = data_filtrada[nameProfundidad].quantile(0.25)
        q3 = data_filtrada[nameProfundidad].quantile(0.75)
        iqr = 1.5 * (q3 - q1)
        lim_inf = q1 - iqr
        lim_sup = q3 + iqr

        atipicos_valores = data_filtrada[(data_filtrada[nameProfundidad] < lim_inf) | (data_filtrada[nameProfundidad] > lim_sup)]
        atipicos.append(atipicos_valores)
    return pd.concat(atipicos, ignore_index = True)

identificar_atipicos(data)
```

Out[]:

	Profundidad_Perforacion	Tiempo_Perforacion	Costo_Operacion	Concentracion_Cobre	Num_Trabajadores	Tipo_Maquinaria	Grupo_Turn
0	126.17	55.57	5422.91	43.12	18.0	Maquinaria_A	Turno_Noch
1	124.94	55.02	5008.77	30.19	19.5	Maquinaria_A	Turno_Noch
2	141.48	48.78	4829.53	31.99	22.0	Maquinaria_A	Turno_Mañan
3	83.94	50.03	5829.92	36.47	26.0	Maquinaria_B	Turno_Mañan
4	71.86	54.37	5639.40	33.68	21.0	Maquinaria_B	Turno_Noch
5	71.12	40.53	4955.27	33.56	24.0	Maquinaria_C	Turno_Mañan
6	136.63	45.90	4651.69	29.10	26.0	Maquinaria_C	Turno_Mañan