

```
from google.colab import drive
drive.mount('/content/drive')
```

↗ Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).

▼ Introducción

Mediante esta libreta vamos a explorar los datos almacenados en estos tres ficheros y a responder preguntas haciendo operaciones y transformaciones con pandas. Además de completar las celdas de código para responder a los ejercicios, cuando así se indique se deberá añadir una celda de tipo *markdown* explicando con texto enriquecido la información o las conclusiones extraídas.

```
import pandas as pd
```

```
# Cargamos el conjunto de datos WorldCupMatches
df_partidos = pd.read_csv("/content/drive/MyDrive/ColabNotebooks/world_cup/WorldCupMatches.csv")
print(df_partidos)
```

```
↗ 4568      NaN      NaN      NaN      NaN
4569      NaN      NaN      NaN      NaN
4570      NaN      NaN      NaN      NaN
4571      NaN      NaN      NaN      NaN

      Win conditions  Attendance  Half-time Home Goals  Half-time Away Goals \
0                0          4444.0             3.0         0.0
1                1          18346.0             2.0         0.0
2                2          24059.0             2.0         0.0
3                3          2549.0              1.0         0.0
4                4          23409.0             0.0         0.0
...            ...            ...            ...            ...
4567            NaN            NaN            NaN            NaN
4568            NaN            NaN            NaN            NaN
4569            NaN            NaN            NaN            NaN
4570            NaN            NaN            NaN            NaN
4571            NaN            NaN            NaN            NaN

      Referee      Assistant 1 \
0  LOMBARDI Domingo (URU)  CRISTOPHE Henry (BEL)
1    MACIAS Jose (ARG)  MATEUCCI Francisco (URU)
2    TEJADA Anibal (URU)  VALLARINO Ricardo (URU)
3    WARNKEN Alberto (CHI)  LANGENUS Jean (BEL)
4    REGO Gilberto (BRA)  SAUCEDO Ulises (BOL)
...            ...            ...
4567            NaN            NaN
4568            NaN            NaN
4569            NaN            NaN
4570            NaN            NaN
4571            NaN            NaN

      Assistant 2  RoundID  MatchID  Home Team Initials \
0    REGO Gilberto (BRA)   201.0   1096.0             FRA
1    WARNKEN Alberto (CHI)   201.0   1090.0             USA
2    BALWAY Thomas (FRA)   201.0   1093.0             YUG
3    MATEUCCI Francisco (URU)   201.0   1098.0             ROU
4    RADULESCU Constantin (ROU)   201.0   1085.0             ARG
...            ...            ...            ...
4567            NaN            NaN            NaN
4568            NaN            NaN            NaN
4569            NaN            NaN            NaN
4570            NaN            NaN            NaN
4571            NaN            NaN            NaN

      Away Team Initials
0                MEX
1                BEL
2                BRA
3                PER
4                FRA
...            ...
4567            NaN
4568            NaN
4569            NaN
4570            NaN
4571            NaN
```

[4572 rows x 20 columns]

▼ 1. Exploración

Usar las funciones `head`, `info` y `describe` para explorar el dataframe `df_partidos`. **Crear una celda de tipo markdown** y comentar brevemente la información almacenada en cada dataframe (columnas y tipo de datos).

No es necesario hacer una documentación exhaustiva, tan solo comentar aspectos principales como qué tipo de información almacena el DataFrame, cuántas filas y columnas tiene, y comentar las columnas que se consideren más relevantes.

```
print(df_partidos.head())
```

```

4568      NaN      NaN      NaN      NaN
4569      NaN      NaN      NaN      NaN
4570      NaN      NaN      NaN      NaN
4571      NaN      NaN      NaN      NaN

Win conditions Attendance Half-time Home Goals Half-time Away Goals \
0      4444.0      3.0      0.0
1      18346.0      2.0      0.0
2      24059.0      2.0      0.0
3      2549.0      1.0      0.0
4      23409.0      0.0      0.0
...      ...      ...      ...
4567      NaN      NaN      NaN
4568      NaN      NaN      NaN
4569      NaN      NaN      NaN
4570      NaN      NaN      NaN
4571      NaN      NaN      NaN

Referee Assistant 1 \
0 LOMBARDI Domingo (URU) CRISTOPHE Henry (BEL)
1 MACIAS Jose (ARG) MATEUCCI Francisco (URU)
2 TEJADA Anibal (URU) VALLARINO Ricardo (URU)
3 WARNKEN Alberto (CHI) LANGENUS Jean (BEL)
4 REGO Gilberto (BRA) SAUCEDO Ulises (BOL)
...      ...      ...
4567      NaN      NaN
4568      NaN      NaN
4569      NaN      NaN
4570      NaN      NaN
4571      NaN      NaN

Assistant 2 RoundID MatchID Home Team Initials \
0 REGO Gilberto (BRA) 201.0 1096.0 FRA
1 WARNKEN Alberto (CHI) 201.0 1090.0 USA
2 BALWAY Thomas (FRA) 201.0 1093.0 YUG
3 MATEUCCI Francisco (URU) 201.0 1098.0 ROU
4 RADULESCU Constantin (ROU) 201.0 1085.0 ARG
...      ...      ...      ...
4567      NaN      NaN      NaN
4568      NaN      NaN      NaN
4569      NaN      NaN      NaN
4570      NaN      NaN      NaN
4571      NaN      NaN      NaN

Away Team Initials
0 MEX
1 BEL
2 BRA
3 PER
4 FRA
...      ...
4567      NaN
4568      NaN
4569      NaN
4570      NaN
4571      NaN

[4572 rows x 20 columns]>
```

```
df_partidos.info()
```

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4572 entries, 0 to 4571
Data columns (total 20 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Year                  852 non-null   float64
1   Datetime              852 non-null   object
2   Stage                 852 non-null   object
3   Stadium               852 non-null   object
4   City                  852 non-null   object
5   Home Team Name        852 non-null   object
6   Home Team Goals       852 non-null   float64
7   Away Team Goals       852 non-null   float64
8   Away Team Name        852 non-null   object
9   Win conditions        852 non-null   object
10  Attendance             850 non-null   float64
11  Half-time Home Goals   852 non-null   float64
12  Half-time Away Goals   852 non-null   float64
```

```
13 Referee      852 non-null object
14 Assistant 1  852 non-null object
15 Assistant 2  852 non-null object
16 RoundID      852 non-null float64
17 MatchID      852 non-null float64
18 Home Team Initials 852 non-null object
19 Away Team Initials 852 non-null object
dtypes: float64(8), object(12)
memory usage: 714.5+ KB
```

df_partidos.head(100)

	Year	Datetime	Stage	Stadium	City	Home Team Name	Home Team Goals	Away Team Goals	Away Team Name
0	1930.0	13 Jul 1930 - 15:00	Group 1	Pocitos	Montevideo	France	4.0	1.0	Mexico
1	1930.0	13 Jul 1930 - 15:00	Group 4	Parque Central	Montevideo	USA	3.0	0.0	Belgium
2	1930.0	14 Jul 1930 - 12:45	Group 2	Parque Central	Montevideo	Yugoslavia	2.0	1.0	Brazil
3	1930.0	14 Jul 1930 - 14:50	Group 3	Pocitos	Montevideo	Romania	3.0	1.0	Peru
4	1930.0	15 Jul 1930 - 16:00	Group 1	Parque Central	Montevideo	Argentina	1.0	0.0	France
...
95	1954.0	27 Jun 1954 - 17:00	Quarter-finals	Charmilles	Geneva	Germany FR	2.0	0.0	Yugoslavia
96	1954.0	27 Jun 1954 - 17:00	Quarter-finals	Wankdorf Stadium	Berne	Hungary	4.0	2.0	Brazil
97	1954.0	30 Jun 1954 - 18:00	Semi-finals	St. Jakob	Basel	Germany FR	6.0	1.0	Austria
98	1954.0	30 Jun 1954 - 18:00	Semi-finals	La Pontaise	Lausanne	Hungary	4.0	2.0	Uruguay
99	1954.0	03 Jul 1954 - 17:00	Match for third place	Hardturm	Zurich	Austria	3.0	1.0	Uruguay

100 rows × 20 columns

df_partidos.describe()

	Year	Home Team Goals	Away Team Goals	Attendance	Half-time Home Goals	Half-time Away Goals	
count	852.000000	852.000000	852.000000	850.000000	852.000000	852.000000	8.520
mean	1985.089202	1.811033	1.022300	45164.800000	0.708920	0.428404	1.066
std	22.448825	1.610255	1.087573	23485.249247	0.937414	0.691252	2.729
min	1930.000000	0.000000	0.000000	2000.000000	0.000000	0.000000	2.010
25%	1970.000000	1.000000	0.000000	30000.000000	0.000000	0.000000	2.620
50%	1990.000000	2.000000	1.000000	41579.500000	0.000000	0.000000	3.370

Con esta `df_partidos.head(100)`, mostramos los primeros 100 registros del data set,
 ✓ con el objetivo de explorar su contenido para interpretar su significado de cada columna.El dataset contiene [4572 rows x 20 columns]>

con el `df_partidos.info()`, visualizamos solamente los nombres de todas las columnas que tiene el dataset.

este dataset contiene 20 columnas diferentes, que son las que se muestran en la celda de código correspondiente.

`df_partidos.describe()` estamos mostrando toda la información de estadística descriptiva para cada una de las columnas que conforman el dataset. Esta información estadística corresponde a: Count (la cantidad de registros), mean (El valor medio), STD (desviación estándar), Los percentiles (25, 50, 75), el max (máximo valor (2014)), min (mínimo valor (1930)).

También me he dado cuenta de que hay varios valores nulos en los registros del dataset y que afectarán los cálculos.

Los tipos de datos correspondientes a las columnas del dataset varían entre los `float64`(8), `object`(12).

En mi opinión las columnas más relevantes son,

Year, Home Team Goals Away, Team Goals home. Attendance, city.

✓ 2. Limpieza de datos

Una de las cosas que llama la atención es la alta presencia de valores perdidos en el DataFrame `df_partidos`. Cuando trabajamos con datos obtenidos del mundo real siempre nos toparemos con problemas relacionados con la medición, captura o almacenamiento de dicha información.

Localiza las filas con valores perdidos. Analizar y **documentar en una nueva celda de tipo markdown** a qué se deben estos valores perdidos.

Finalmente usar la función [dropna](#) sobre el DataFrame `df_partidos` para eliminar los valores perdidos del DataFrame, y almacenar el resultado en `df_partidos` de nuevo.

```
new_var = df_partidos.isnull().any(axis=1)
print(new_var)
```

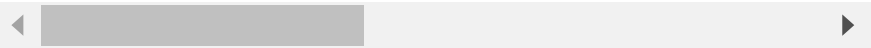
```
0      False
1      False
2      False
3      False
4      False
...
4567    True
4568    True
4569    True
4570    True
4571    True
Length: 4572, dtype: bool
```

```
df_partidos.head(100)
```



	Year	Datetime	Stage	Stadium	City	Home Team Name	Home Team Goals	Away Team Goals	Away Team Name
0	1930.0	13 Jul 1930 - 15:00	Group 1	Pocitos	Montevideo	France	4.0	1.0	Mexico
1	1930.0	13 Jul 1930 - 15:00	Group 4	Parque Central	Montevideo	USA	3.0	0.0	Belgium
2	1930.0	14 Jul 1930 - 12:45	Group 2	Parque Central	Montevideo	Yugoslavia	2.0	1.0	Brazil
3	1930.0	14 Jul 1930 - 14:50	Group 3	Pocitos	Montevideo	Romania	3.0	1.0	Peru
4	1930.0	15 Jul 1930 - 16:00	Group 1	Parque Central	Montevideo	Argentina	1.0	0.0	France
...
95	1954.0	27 Jun 1954 - 17:00	Quarter-finals	Charmilles	Geneva	Germany FR	2.0	0.0	Yugoslavia
96	1954.0	27 Jun 1954 - 17:00	Quarter-finals	Wankdorf Stadium	Berne	Hungary	4.0	2.0	Brazil
97	1954.0	30 Jun 1954 - 18:00	Semi-finals	St. Jakob	Basel	Germany FR	6.0	1.0	Austria
98	1954.0	30 Jun 1954 - 18:00	Semi-finals	La Pontaise	Lausanne	Hungary	4.0	2.0	Uruguay
99	1954.0	03 Jul 1954 - 17:00	Match for third place	Hardturm	Zurich	Austria	3.0	1.0	Uruguay

100 rows × 20 columns



Pasos siguientes: ☒ Ver gráficos recomendados

```
print (new_var)

0      False
1      False
2      False
3      False
4      False
...
4567   True
4568   True
4569   True
4570   True
4571   True
Length: 4572, dtype: bool

df_partidos.dropna()
```



	Year	Datetime	Stage	Stadium	City	Home Team Name	Home Team Goals	Away Team Goals	Away Team Name	Win conditions	Attendance	Half-time Home Goals	Half-time Away Goals	F
0	1930.0	13 Jul 1930 - 15:00	Group 1	Pocitos	Montevideo	France	4.0	1.0	Mexico		4444.0	3.0	0.0	LOM C
1	1930.0	13 Jul 1930 - 15:00	Group 4	Parque Central	Montevideo	USA	3.0	0.0	Belgium		18346.0	2.0	0.0	M Josi
2	1930.0	14 Jul 1930 - 12:45	Group 2	Parque Central	Montevideo	Yugoslavia	2.0	1.0	Brazil		24059.0	2.0	0.0	7 Anibe
3	1930.0	14 Jul 1930 - 14:50	Group 3	Pocitos	Montevideo	Romania	3.0	1.0	Peru		2549.0	1.0	0.0	WA Alber
4	1930.0	15 Jul 1930 - 16:00	Group 1	Parque Central	Montevideo	Argentina	1.0	0.0	France		23409.0	0.0	0.0	
...	
847	2014.0	05 Jul 2014 - 17:00	Quarter-finals	Arena Fonte Nova	Salvador	Netherlands	0.0	0.0	Costa Rica	Netherlands win on penalties (4 - 3)	51179.0	0.0	0.0	F IR
848	2014.0	08 Jul 2014 - 17:00	Semi-finals	Estadio Mineirao	Belo Horizonte	Brazil	1.0	7.0	Germany		58141.0	0.0	5.0	RODF Marcu
849	2014.0	09 Jul 2014 - 17:00	Semi-finals	Arena de Sao Paulo	Sao Paulo	Netherlands	0.0	0.0	Argentina	Argentina win on penalties (2 - 4)	63267.0	0.0	0.0	
850	2014.0	12 Jul 2014 - 17:00	Play-off for third place	Estadio Nacional	Brasilia	Brazil	0.0	3.0	Netherlands		68034.0	0.0	2.0	HA
851	2014.0	13 Jul 2014 - 16:00	Final	Estadio do Maracana	Rio De Janeiro	Germany	1.0	0.0	Argentina	Germany win after extra time	74738.0	0.0	0.0	F

850 rows × 20 columns

```
df_partidos.describe()
```



	Year	Home Team Goals	Away Team Goals	Attendance	Half-time Home Goals	Half-time Away Goals	
count	852.000000	852.000000	852.000000	850.000000	852.000000	852.000000	8.520
mean	1985.089202	1.811033	1.022300	45164.800000	0.708920	0.428404	1.066
std	22.448825	1.610255	1.087573	23485.249247	0.937414	0.691252	2.729
min	1930.000000	0.000000	0.000000	2000.000000	0.000000	0.000000	2.010
25%	1970.000000	1.000000	0.000000	30000.000000	0.000000	0.000000	2.620
50%	1990.000000	2.000000	1.000000	41579.500000	0.000000	0.000000	3.370

Primero localizamos con la funcion `df_partidos.isnull().any(axis=1)` todas las filas que tienen valores nulos, devolviendo un valor booleano False, en las filas donde el valor no es nulo. Y True donde el valor si es nulo. Por ejemplo las utimas filas del dataframe son True.

`df_partidos.dropna()` nos elimina todos los nulos que existen en el dataframe. En este caso podemos concluir que la gran mayoria de registros del dataframe en estudio tiene

valores nulos.

`df_partidos.describe()` volvemos a mirar la info estadística y nos damos cuenta que los cálculos no han tenido en cuenta los valores nulos.


3. Cargar datos

Al igual que hemos cargado los datos de los partidos en `df_partidos`, crear las variables `df_jugadores` y `df_copas` que contengan los dataframes correspondientes a la lectura de los csv `WorldCupPlayers` y `WorldCups` (usar `pd.read_csv`).

```
import pandas as pd
```


```
# Cargamos el conjunto de datos WorldCupPlayers
```

```
df_jugadores = pd.read_csv("/content/drive/MyDrive/ColabNotebooks/world_cup/WorldCupPlayers.csv")
df_jugadores
```



	RoundID	MatchID	Team Initials	Coach Name	Line-up	Shirt Number	Player Name	Position
0	201	1096	FRA	CAUDRON Raoul (FRA)	S	0	Alex THEPOT	GK
1	201	1096	MEX	LUQUE Juan (MEX)	S	0	Oscar BONFIGLIO	GK
2	201	1096	FRA	CAUDRON Raoul (FRA)	S	0	Marcel LANGILLER	NaN
3	201	1096	MEX	LUQUE Juan (MEX)	S	0	Juan CARRENO	NaN
4	201	1096	FRA	CAUDRON Raoul (FRA)	S	0	Ernest LIBERATI	NaN

```
print(df_jugadores.head())
```




	RoundID	MatchID	Team Initials	Coach Name	Line-up	
0	201	1096	FRA	CAUDRON Raoul (FRA)	S	
1	201	1096	MEX	LUQUE Juan (MEX)	S	
2	201	1096	FRA	CAUDRON Raoul (FRA)	S	
3	201	1096	MEX	LUQUE Juan (MEX)	S	
4	201	1096	FRA	CAUDRON Raoul (FRA)	S	
...
37779	255959	300186501	ARG	SABELLA Alejandro (ARG)	N	
37780	255959	300186501	GER	LOEW Joachim (GER)	N	
37781	255959	300186501	ARG	SABELLA Alejandro (ARG)	N	
37782	255959	300186501	GER	LOEW Joachim (GER)	N	
37783	255959	300186501	ARG	SABELLA Alejandro (ARG)	N	

	Shirt Number	Player Name	Position	Event
0	0	Alex THEPOT	GK	NaN
1	0	Oscar BONFIGLIO	GK	NaN
2	0	Marcel LANGILLER	NaN	G40'
3	0	Juan CARRENO	NaN	G70'
4	0	Ernest LIBERATI	NaN	NaN
...
37779	19	ALVAREZ	NaN	NaN
37780	6	KHEDIRA	NaN	NaN
37781	20	AGUERO	NaN	IH46' Y65'
37782	21	MUSTAFI	NaN	NaN
37783	23	BASANTA	NaN	NaN

```
[37784 rows x 9 columns]>
```

```
print(df_jugadores.info())
```



```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 37784 entries, 0 to 37783
Data columns (total 9 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   RoundID     37784 non-null  int64
1   MatchID     37784 non-null  int64
```

```

2 Team Initials 37784 non-null object
3 Coach Name    37784 non-null object
4 Line-up       37784 non-null object
5 Shirt Number  37784 non-null int64
6 Player Name   37784 non-null object
7 Position      4143 non-null object
8 Event         9069 non-null object
dtypes: int64(3), object(6)
memory usage: 2.6+ MB
None

```

```
print(df_jugadores.describe())
```

```

↗
count      RoundID      MatchID  Shirt Number
mean    3.778400e+04  3.778400e+04  10.726022
std      2.770144e+07  1.123916e+08   6.960138
min      2.010000e+02  2.500000e+01   0.000000
25%      2.630000e+02  1.199000e+03   5.000000
50%      3.370000e+02  2.216000e+03  11.000000
75%      2.559310e+05  9.741000e+07  17.000000
max      9.741060e+07  3.001865e+08  23.000000

```

cargamos el dataset de WorldCupPlayers.csv el cual contiene toda la informacion correspondiente a todos los jugadores que han participado en los distintos mundiales.

```
import pandas as pd
```

```
df_copas = pd.read_csv("/content/drive/MyDrive/ColabNotebooks/world_cup/WorldCups.csv")
df_copas
```

```

↗
   Year  Country  Winner  Runners-Up  Third  Fourth  GoalsScored  Qu
0  1930   Uruguay  Uruguay   Argentina    USA  Yugoslavia         70
1  1934     Italy    Italy Czechoslovakia  Germany    Austria         70
2  1938   France    Italy    Hungary    Brazil    Sweden         84
3  1950    Brazil  Uruguay     Brazil    Sweden    Spain         88
4  1954 Switzerland  Germany FR    Hungary    Austria    Uruguay        140
5  1958    Sweden    Brazil    Sweden    France  Germany FR        126
6  1962     Chile    Brazil Czechoslovakia    Chile  Yugoslavia         89
7  1966   England  England  Germany FR    Portugal    Soviet Union         89
8  1970    Mexico    Brazil     Italy  Germany FR    Uruguay         95
9  1974   Germany  Germany FR    Netherlands    Poland    Brazil         97
10 1978   Argentina  Argentina    Netherlands    Brazil    Italy        102
11 1982     Spain    Italy  Germany FR    Poland    France        146
12 1986    Mexico  Argentina  Germany FR    France    Belgium        132
13 1990     Italy  Germany FR    Argentina    Italy    England        115
14 1994     USA    Brazil     Italy    Sweden    Bulgaria        141
15 1998   France    France     Brazil    Croatia  Netherlands        171

```

```
print(df_copas.head())
```

```

↗
<bound method NDFrame.head of df_copas>
0  1930   Uruguay  Uruguay   Argentina    USA
1  1934     Italy    Italy Czechoslovakia  Germany
2  1938   France    Italy    Hungary    Brazil
3  1950    Brazil  Uruguay     Brazil    Sweden
4  1954 Switzerland  Germany FR    Hungary    Austria
5  1958    Sweden    Brazil    Sweden    France

```


6	1962	Chile	Brazil	Czechoslovakia	Chile
7	1966	England	England	Germany FR	Portugal
8	1970	Mexico	Brazil	Italy	Germany FR
9	1974	Germany	Germany FR	Netherlands	Poland
10	1978	Argentina	Argentina	Netherlands	Brazil
11	1982	Spain	Italy	Germany FR	Poland
12	1986	Mexico	Argentina	Germany FR	France
13	1990	Italy	Germany FR	Argentina	Italy
14	1994	USA	Brazil	Italy	Sweden
15	1998	France	France	Brazil	Croatia
16	2002	Korea/Japan	Brazil	Germany	Turkey
17	2006	Germany	Italy	France	Germany
18	2010	South Africa	Spain	Netherlands	Germany
19	2014	Brazil	Germany	Argentina	Netherlands

	Fourth	GoalsScored	QualifiedTeams	MatchesPlayed	Attendance
0	Yugoslavia	70	13	18	590.549
1	Austria	70	16	17	363.000
2	Sweden	84	15	18	375.700
3	Spain	88	13	22	1.045.246
4	Uruguay	140	16	26	768.607
5	Germany FR	126	16	35	819.810
6	Yugoslavia	89	16	32	893.172
7	Soviet Union	89	16	32	1.563.135
8	Uruguay	95	16	32	1.603.975
9	Brazil	97	16	38	1.865.753
10	Italy	102	16	38	1.545.791
11	France	146	24	52	2.109.723
12	Belgium	132	24	52	2.394.031
13	England	115	24	52	2.516.215
14	Bulgaria	141	24	52	3.587.538
15	Netherlands	171	32	64	2.785.100
16	Korea Republic	161	32	64	2.705.197
17	Portugal	147	32	64	3.359.439
18	Uruguay	145	32	64	3.178.856
19	Brazil	171	32	64	3.386.810

```
print(df_copas.info())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20 entries, 0 to 19
Data columns (total 10 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Year                  20 non-null    int64
1   Country               20 non-null    object
2   Winner                20 non-null    object
3   Runners-Up            20 non-null    object
4   Third                 20 non-null    object
5   Fourth                20 non-null    object
6   GoalsScored           20 non-null    int64
7   QualifiedTeams        20 non-null    int64
8   MatchesPlayed         20 non-null    int64
9   Attendance             20 non-null    object
dtypes: int64(4), object(6)
memory usage: 1.7+ KB
None
```

```
print(df_copas.describe())
```

```
count      Year  GoalsScored  QualifiedTeams  MatchesPlayed
mean    1974.800000    118.950000    21.250000    41.800000
std       25.582889     32.972836     7.268352    17.218717
min     1930.000000     70.000000    13.000000    17.000000
25%     1957.000000     89.000000    16.000000    30.500000
50%     1976.000000    120.500000    16.000000    38.000000
75%     1995.000000    145.250000    26.000000    55.000000
max      2014.000000    171.000000    32.000000    64.000000
```

✓ cargamos el dataset de WorldCups.csv el cual contiene toda la informacion correspondiente a todas las copas que han ganado en los distintos mundiales.

✓ 4. Rango temporal

El dataframe `df_copas` contiene datos de todos los mundiales disputados. ¿Cuál es el año del mundial más antiguo disputado? ¿Y el año del mundial más reciente?

Pista: recuerda que podemos usar funciones de agregación (`count` para contar el número de casos, `mean` para calcular el valor medio, etc.) directamente sobre columnas o Series de pandas. Por ejemplo, el siguiente código nos muestra el mayor número de goles marcado en un

mundial.

```
df_copas["GoalsScored"].max()
```

Opcional: Una vez localizados los años de interés (más antiguo y más reciente) visualizar las filas completas correspondientes a cada año usando la función `.loc`.

```
año_ultimo=df_copas["Year"].max()  
print("El ultimo fue:",año_ultimo)
```

```
↗ El ultimo fue: 2014
```

```
año_antiguo=df_copas["Year"].min()  
print("El el primer año fue:",año_antiguo,)
```

```
↗ El el primer año fue: 1930
```

```
fila_max_year = df_copas.loc[df_copas["Year"] ==año_antiguo, ["Year","Country", "Winner","Runners-Up","Third","Fourth"]]  
print(fila_max_year)
```

```
↗
```

Year	Country	Winner	Runners-Up	Third	Fourth
------	---------	--------	------------	-------	--------