

NIVEL 4

PANDAS - DataFrames



¿QUÉ ES UN DATAFRAME?

- ✓ Muchos registros, ordenados según algún criterio
- ✓ Varias columnas con nombre
- ✓ Diferentes tipos de datos por columnas

Tiempo	Atleta	País	Fecha	Ciudad
9.58	Usain Bolt	Jamaica	16/08/09	Berlin
9.69	Usain Bolt	Jamaica	16/09/08	Beijing
9.72	Usain Bolt	Jamaica	31/05/08	New York
9.74	Asafa Powell	Jamaica	9/09/07	Rieti
9.77	Asafa Powell	Jamaica	18/08/06	Zurich

CONSTRUIR UN DATAFRAME DESDE UNA LISTA DE DICCIONARIOS

A partir de una lista de diccionarios

```
a1 = {"tiempo": 9.58, "atleta": "Usain Bolt", "país": "Jamaica", "fecha": "16/08/2009", "ciudad": "Berlin"}
a2 = {"tiempo": 9.69, "atleta": "Usain Bolt", "país": "Jamaica", "fecha": "16/09/2008", "ciudad": "Beijing"}
a3 = {"tiempo": 9.72, "atleta": "Usain Bolt", "país": "Jamaica", "fecha": "31/05/2008", "ciudad": "New York"}
a4 = {"tiempo": 9.74, "atleta": "Asafa Powell", "país": "Jamaica", "fecha": "9/09/2007", "ciudad": "Rieti"}
a5 = {"tiempo": 9.77, "atleta": "Asafa Powell", "país": "Jamaica", "fecha": "18/08/2006", "ciudad": "Zurich"}
records = [a1, a2, a3, a4, a5]
```

```
df_records1 = pd.DataFrame(records)
```

- Un diccionario por registro
- Una lista de diccionarios

In [580]: df_records1

Out[580]:

	tiempo	atleta	país	fecha	ciudad
0	9.58	Usain Bolt	Jamaica	16/08/2009	Berlin
1	9.69	Usain Bolt	Jamaica	16/09/2008	Beijing
2	9.72	Usain Bolt	Jamaica	31/05/2008	New York
3	9.74	Asafa Powell	Jamaica	9/09/2007	Rieti
4	9.77	Asafa Powell	Jamaica	18/08/2006	Zurich

CONSTRUIR UN DATAFRAME DESDE UN DICCIONARIO DE LISTAS

- Una lista de valores por columna
- Un diccionario para organizar las columnas y darles nombre

```
# A partir de un diccionario de listas
tiempos = [9.58, 9.69, 9.72, 9.74, 9.77]
atletas = ["Usain Bolt", "Usain Bolt", "Usain Bolt", "Asafa Powell", "Asafa Powell"]
países = ["Jamaica", "Jamaica", "Jamaica", "Jamaica", "Jamaica"]
fechas = ["16/08/2009", "16/09/2008", "31/05/2008", "9/09/2007", "18/08/2006"]
ciudades = ["Berlin", "Beijing", "New York", "Rieti", "Zurich"]

datos = {"tiempo": tiempos, "atleta": atletas, "país": países, "fecha": fechas, "ciudad": ciudades}

df_records2 = pd.DataFrame(datos)
```

CONSTRUIR UN DATAFRAME DESDE UN DICCIONARIO DE SERIES

- Una serie por columna
- Un diccionario para organizar las columnas y darles nombre

A partir de series

```
tiempos = pd.Series([9.58, 9.69, 9.72, 9.74, 9.77])
atletas = pd.Series(["Usain Bolt", "Usain Bolt", "Usain Bolt", "Asafa Powell", "Asafa Powell"])
países = pd.Series(["Jamaica", "Jamaica", "Jamaica", "Jamaica", "Jamaica"])
fechas = pd.Series(["16/08/2009", "16/09/2008", "31/05/2008", "9/09/2007", "18/08/2006"])
ciudades = pd.Series(["Berlin", "Beijing", "New York", "Rieti", "Zurich"])

datos_series = {"tiempo": tiempos, "atleta": atletas, "país": países, "fecha": fechas, "ciudad": ciudades}
df_records3 = pd.DataFrame(datos_series)
```

CONSTRUIR UN DATAFRAME A PARTIR DE UN ARCHIVO CSV

- CSV: Comma Separated Values (Valores separados por coma)



	tiempo,atleta,país,fecha,ciudad
1	9.58,Usain Bolt,Jamaica,16/08/2009,Berlin
2	9.69,Usain Bolt,Jamaica,16/09/2008,Beijing
3	9.72,Usain Bolt,Jamaica,31/05/2008,New York
4	9.74,Asafa Powell,Jamaica,9/09/2007,Rieti
5	9.77,Asafa Powell,Jamaica,18/08/2006,Zurich
6	
7	

CARGAR UN ARCHIVO CSV A UN DATAFRAME

```
# A partir de un archivo CSV
df_records4 = pd.read_csv("records.csv")
```

```
In [581]: df_records2
Out[581]:
```

	tiempo	atleta	país	fecha	ciudad
0	9.58	Usain Bolt	Jamaica	16/08/2009	Berlin
1	9.69	Usain Bolt	Jamaica	16/09/2008	Beijing
2	9.72	Usain Bolt	Jamaica	31/05/2008	New York
3	9.74	Asafa Powell	Jamaica	9/09/2007	Rieti
4	9.77	Asafa Powell	Jamaica	18/08/2006	Zurich

```
In [582]: df_records3
Out[582]:
```

	tiempo	atleta	país	fecha	ciudad
0	9.58	Usain Bolt	Jamaica	16/08/2009	Berlin
1	9.69	Usain Bolt	Jamaica	16/09/2008	Beijing
2	9.72	Usain Bolt	Jamaica	31/05/2008	New York
3	9.74	Asafa Powell	Jamaica	9/09/2007	Rieti
4	9.77	Asafa Powell	Jamaica	18/08/2006	Zurich

```
In [583]: df_records4
Out[583]:
```

	tiempo	atleta	país	fecha	ciudad
0	9.58	Usain Bolt	Jamaica	16/08/2009	Berlin
1	9.69	Usain Bolt	Jamaica	16/09/2008	Beijing
2	9.72	Usain Bolt	Jamaica	31/05/2008	New York
3	9.74	Asafa Powell	Jamaica	9/09/2007	Rieti
4	9.77	Asafa Powell	Jamaica	18/08/2006	Zurich

EJEMPLO: CONTRATOS ESTATALES

GOV.CO | **DATOS ABIERTOS**

Inicio Descubre Publica Visualiza Participa Herramientas Novedades Usos Activos Calidad **Español**

SECOP II - Contratos Electrónicos

Gastos Gubernamentales Visualizar y explorar

Información de los contratos registrados en SECOP II desde su lanzamiento

Actualizado
26 de octubre de 2020

Datos proporcionados por
Colombia Compra Eficiente

Sobre este conjunto de datos

Actualizado
26 de octubre de 2020

Datos actualizados por última vez
26 de octubre de 2020

Última actualización de metadatos
26 de agosto de 2020

Fecha de cración
30 de septiembre de 2019

Vistas
26,3K

Descargas
6.519

Datos ofrecidos por
Colombia Compra Eficiente

Propietario de conjunto de datos
Colombia Compra

Información de la Entidad

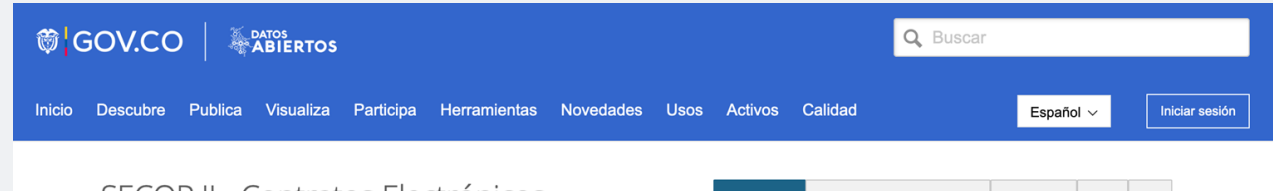
Área o dependencia	Subdirección de IDT
Nombre de la Entidad	Agencia Nacional de Contratación Pública Colombia Compra Eficiente
Departamento	Bogotá D.C.
Municipio	Bogotá D.C.
Orden	Nacional
Sector	Planeación

Información de Datos

Idioma	Español
Cobertura Geográfica	Nacional

<https://www.datos.gov.co/Gastos-Gubernamentales/SECOP-II-Contratos-Electr-nicos/jbjy-vk9h>

EJEMPLO: CONTRATOS ESTATALES



In [586]: data.columns

Out[586]:

```
Index(['Nombre Entidad', 'Nit Entidad', 'Departamento', 'Ciudad', 'Localización', 'Orden', 'Sector', 'Rama',
      'Entidad Centralizada', 'Proceso de Compra', 'ID Contrato', 'Referencia del Contrato', 'Estado Contrato',
      'Codigo de Categoria Principal', 'Descripcion del Proceso', 'Tipo de Contrato', 'Modalidad de Contratacion',
      'Justificacion Modalidad de Contratacion', 'Fecha de Firma', 'Fecha de Inicio del Contrato', 'Fecha de Fin del Contrato',
      'Fecha de Inicio de Ejecucion', 'Fecha de Fin de Ejecucion', 'Condiciones de Entrega', 'TipoDocProveedor',
      'Documento Proveedor', 'Proveedor Adjudicado', 'Es Grupo', 'Es Pyme', 'Habilita Pago Adelantado', 'Liquidación',
      'Obligación Ambiental', 'Obligaciones Postconsumo', 'Reversion', 'Valor del Contrato', 'Valor de pago adelantado',
      'Valor Facturado', 'Valor Pendiente de Pago', 'Valor Pagado', 'Valor Amortizado', 'Valor Pendiente de Amortizacion',
      'Valor Pendiente de Ejecucion', 'Estado BPIN', 'Código BPIN', 'Anno BPIN', 'Saldo CDP', 'Saldo Vigencia',
      'EsPostConflicto', 'URLProceso', 'Destino Gasto', 'Origen de los Recursos', 'Dias Adicionados', 'Puntos del Acuerdo',
      'Pilares del Acuerdo'],
      dtype='object')
```

Vistas		Descargas	
26,3K		6.519	
Orden		Nacional	
Sector		Planeación	
Información de Datos			
Idioma		Español	
Cobertura Geográfica		Nacional	
Datos ofrecidos por Colombia Compra Eficiente		Propietario de conjunto de datos Colombia Compra	

<https://www.datos.gov.co/Gastos-Gubernamentales/SECOP-II-Contratos-Electr-nicos/jbjy-vk9h>

SELECCIÓN DE COLUMNAS

- Para seleccionar columnas se utilizan `[]` y una lista con los nombres de las columnas que nos interesan
- La función `copy()` se usa para que el DataFrame con nuevas columnas esté completamente desconectado del original

```
columnas_valores = ['Valor del Contrato', 'Valor de pago adelantado',
                    'Valor Pendiente de Pago', 'Valor Pagado']
valores = data[columnas_valores].copy()
```

```
In [691]: valores
Out[691]:
```

	Valor del Contrato	Valor de pago adelantado	Valor Pendiente de Pago	Valor Pagado
0	22,279,494.00	0.00	22,279,494.00	0.00
1	52,075,161.00	0.00	52,075,161.00	0.00
2	12,250,000.00	0.00	12,250,000.00	0.00
3	63,963,000.00	0.00	63,963,000.00	0.00
4	13,875,600.00	0.00	13,875,600.00	0.00
...
661630	25,226,674.00	0.00	25,226,674.00	0.00
661631	0.00	0.00	0.00	0.00
661632	764,354,699.00	0.00	764,354,699.00	0.00
661633	21,395,160.00	0.00	21,395,160.00	0.00
661634	5,727,000.00	0.00	3,657,000.00	2,070,000.00

[661635 rows x 4 columns]



ANÁLISIS BÁSICO DE DATAFRAMES

EXPLORACIÓN DE VALORES CATEGÓRICOS

```
In [681]: categoricos.head(5)
```

```
Out[681]:
```

	Sector	Estado Contrato	Fecha de Firma	Es Pyme
0	Ley de Justicia	En ejecución	2020-05-28	True
1	Ambiente y Desarrollo Sostenible	Activo	NaT	True
2	Ambiente y Desarrollo Sostenible	Activo	2019-02-22	False
3	No aplica/No pertenece	Modificado	2019-01-30	False
4	Salud y Protección Social	Modificado	2020-02-19	False

```
In [693]: categoricos.tail()
```

```
Out[693]:
```

	Sector	Estado Contrato	Fecha de Firma	Es Pyme
661630	Inclusión Social y Reconciliación	Activo	2019-01-17	False
661631	Tecnologías de la Información y las Comunicaci...	Borrador	NaT	False
661632	Servicio Público	Activo	2019-11-26	True
661633	Salud y Protección Social	Activo	2020-08-12	False
661634	Planeación	cedido	2018-08-10	False

head() - tail ()

EXPLORACIÓN DE VALORES CATEGÓRICOS

```
In [635]: categoricos.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 661635 entries, 0 to 661634
Data columns (total 4 columns):
#   Column          Non-Null Count  Dtype
---  -
0   Sector           661635 non-null object
1   Estado Contrato  661635 non-null object
2   Fecha de Firma   573975 non-null datetime64[ns]
3   Es Pyme          661635 non-null bool
dtypes: bool(1), datetime64[ns](1), object(2)
memory usage: 15.8+ MB
```

Formato fecha de numpy

Cuántos valores no son nulos en cada columna

```
In [641]: categoricos.describe( )
```

```
Out[641]:
```

	Sector	Estado Contrato	Fecha de Firma	Es Pyme
count	661635	661635	573975	661635
unique	26	12	1498	2
top	Salud y Protección Social	Activo	2018-01-26 00:00:00	False
freq	113811	254172	4932	536738
first	NaN	NaN	2015-06-11 00:00:00	NaN
last	NaN	NaN	2020-09-25 00:00:00	NaN

info() - describe ()

EXPLORACIÓN DE COLUMNAS

```
In [644]: categoricos['Sector']
Out[644]:
0                Ley de Justicia
1    Ambiente y Desarrollo Sostenible
2    Ambiente y Desarrollo Sostenible
3                No aplica/No pertenece
4                Salud y Protección Social
...
661630    Inclusión Social y Reconciliación
661631    Tecnologías de la Información y las Comunicaci...
661632                Servicio Público
661633                Salud y Protección Social
661634                Planeación
Name: Sector, Length: 661635, dtype: object

In [645]: type(categoricos['Sector'])
Out[645]: pandas.core.series.Series
```

```
In [648]: categoricos[['Sector', 'Es Pyme']]
Out[648]:
```

	Sector	Es Pyme
0	Ley de Justicia	True
1	Ambiente y Desarrollo Sostenible	True
2	Ambiente y Desarrollo Sostenible	False
3	No aplica/No pertenece	False
4	Salud y Protección Social	False
...
661630	Inclusión Social y Reconciliación	False
661631	Tecnologías de la Información y las Comunicaci...	False
661632	Servicio Público	True
661633	Salud y Protección Social	False
661634	Planeación	False

[661635 rows x 2 columns]

EXPLORACIÓN DE COLUMNAS

unique()

```
In [655]: categoricos['Sector'].unique()
Out[655]:
array(['Ley de Justicia', 'Ambiente y Desarrollo Sostenible',
      'No aplica/No pertenece', 'Salud y Protección Social',
      'Inclusión Social y Reconciliación', 'Servicio Público', 'Trabajo',
      'Educación Nacional',
      'Inteligencia Estratégica y Contrainteligencia', 'deportes',
      'Hacienda y Crédito Público', 'defensa', 'Minas y Energía',
      'Cultura', 'Planeación', 'Transporte', 'Industria',
      'Información Estadística', 'agricultura',
      'Tecnologías de la Información y las Comunicaciones',
      'Vivienda, Ciudad y Territorio', 'interior',
      'Presidencia de la República', 'Ciencia Tecnología',
      'Relaciones Exteriores', 'No Definido'], dtype=object)

In [656]: valores_posibles = list(categoricos['Sector'].unique())
```

Valores que aparecen **al menos una vez** en la columna 'Sector'

value_counts()

```
In [657]: categoricos['Sector'].value_counts()
Out[657]:
Salud y Protección Social      113811
Servicio Público              87299
defensa                      80099
Inclusión Social y Reconciliación 39121
No aplica/No pertenece       38541
...
Presidencia de la República    5664
Ciencia Tecnología            2158
Relaciones Exteriores          1328
Inteligencia Estratégica y Contrainteligencia 217
No Definido                    1
Name: Sector, Length: 26, dtype: int64

In [658]: type(categoricos['Sector'].value_counts())
Out[658]: pandas.core.series.Series
```

El valor que más aparece en la columna 'Sector'

EXPLORACIÓN DE VALORES NUMÉRICOS

In [680]: valores.head(5)

Out[680]:

	Valor del Contrato	Valor de pago adelantado	Valor Pendiente de Pago	Valor Pagado
0	22,279,494.00	0.00	22,279,494.00	0.00
1	52,075,161.00	0.00	52,075,161.00	0.00
2	12,250,000.00	0.00	12,250,000.00	0.00
3	63,963,000.00	0.00	63,963,000.00	0.00
4	13,875,600.00	0.00	13,875,600.00	0.00

In [632]: valores.info()

<class 'pandas.core.frame.DataFrame'>

RangeIndex: 661635 entries, 0 to 661634

Data columns (total 4 columns):

#	Column	Non-Null Count	Dtype
0	Valor del Contrato	661635 non-null	float64
1	Valor de pago adelantado	661635 non-null	float64
2	Valor Pendiente de Pago	661635 non-null	float64
3	Valor Pagado	661635 non-null	float64

dtypes: float64(4)

memory usage: 20.2 MB

Tipo de dato de las columnas

EXPLORACIÓN DE VALORES NUMÉRICOS

```
In [663]: valores.describe()
```

```
Out[663]:
```

	Valor del Contrato	Valor de pago adelantado	Valor Pendiente de Pago	Valor Pagado
count	661,635.00	661,635.00	661,635.00	661,635.00
mean	407,822,408,655.17	869,674.00	176,414,362.51	13,737,565.91
std	40,808,945,909,560.44	81,863,825.47	4,964,761,222.34	366,541,995.75
min	0.00	0.00	-2,369,879,332.00	0.00
25%	9,000,000.00	0.00	5,279,247.50	0.00
50%	19,818,986.00	0.00	14,540,000.00	0.00
75%	41,096,252.00	0.00	33,672,158.50	0.00
max	9,801,000,000,000,000.00	39,684,085,728.00	936,890,230,136.00	92,710,323,877.00

ESTADÍSTICAS DEL DATAFRAME

```
In [732]: valores.mean()
Out[732]:
Valor del Contrato      407,822,408,655.18
Valor de pago adelantado    869,674.00
Valor Pendiente de Pago    176,414,362.51
Valor Pagado              13,737,565.91
dtype: float64
```

```
In [733]: valores['Valor Pagado'].mean()
Out[733]: 13737565.908408714
```

```
In [670]: valores.max()
Out[670]:
Valor del Contrato      9,801,000,000,000,000.00
Valor de pago adelantado    39,684,085,728.00
Valor Pendiente de Pago    936,890,230,136.00
Valor Pagado              92,710,323,877.00
dtype: float64
```

```
In [671]: valores.idxmax()
Out[671]:
Valor del Contrato      449009
Valor de pago adelantado    124725
Valor Pendiente de Pago    238834
Valor Pagado              562292
dtype: int64
```

mean, mode, std
max, min, idxmax, idxmin

SELECCIÓN DE REGISTROS POR ÍNDICE Y POSICIÓN

In [719]: valores.iloc[449009]

Out[719]:

Valor del Contrato 9,801,000,000,000,000.00

Valor de pago adelantado 0.00

Valor Pendiente de Pago 0.00

Valor Pagado 0.00

Name: 449009, dtype: float64

In [720]: data.loc[449009]

Out[720]:

Nombre Entidad AGENCIA NACIONAL DE INFRAESTRUCTURA – ANI

Nit Entidad 8,301,259,969

Departamento Distrito Capital de Bogotá

Ciudad Bogotá

Localización Colombia, Bogotá, Bogotá

Orden Nacional

Sector Transporte

Rama Ejecutivo

Entidad Centralizada Centralizada

Proceso de Compra C01.BDOS.678434

ID Contrato C01.PCCNTR.741088

Referencia del Contrato VGC-076-2019

Estado Contrato Cancelado

Codigo de Categoria Principal V1.80101603

Descripcion del Proceso EL CONTRATISTA se obliga para con la AGENCIA N...

Tipo de Contrato Prestación de servicios

Modalidad de Contratacion Contratación directa

Justificacion Modalidad de Contratacion ServiciosProfesionales

ESTADÍSTICAS DEL DATAFRAME

quantile()

In [725]: valores.describe()

Out[725]:

	Valor del Contrato	Valor de pago adelantado	Valor Pendiente de Pago	Valor Pagado
count	661,635.00	661,635.00	661,635.00	661,635.00
mean	407,822,408,655.17	869,674.00	176,414,362.51	13,737,565.91
std	40,808,945,909,560.44	81,863,825.47	4,964,761,222.34	366,541,995.75
min	0.00	0.00	-2,369,879,332.00	0.00
25%	9,000,000.00	0.00	5,279,247.50	0.00
50%	19,818,986.00	0.00	14,540,000.00	0.00
75%	41,096,252.00	0.00	33,672,158.50	0.00
max	9,801,000,000,000,000.00	39,684,085,728.00	936,890,230,136.00	92,710,323,877.00

In [726]: valores.quantile(0.50)

Out[726]:

Valor del Contrato	19,818,986.00
Valor de pago adelantado	0.00
Valor Pendiente de Pago	14,540,000.00
Valor Pagado	0.00

Name: 0.5, dtype: float64

In [727]: valores.quantile(0.95)

Out[727]:

Valor del Contrato	226,884,053.60
Valor de pago adelantado	0.00
Valor Pendiente de Pago	194,841,989.00
Valor Pagado	33,000,000.00

Name: 0.95, dtype: float64

OPERACIONES AGREGADAS

```
In [669]: valores.sum()
Out[669]:
Valor del Contrato      269,829,579,350,569,536.00
Valor de pago adelantado    575,406,759,104.00
Valor Pendiente de Pago    116,721,916,738,404.00
Valor Pagado              9,089,254,419,810.00
dtype: float64
```

```
In [742]: valores.div(1000000)
Out[742]:
```

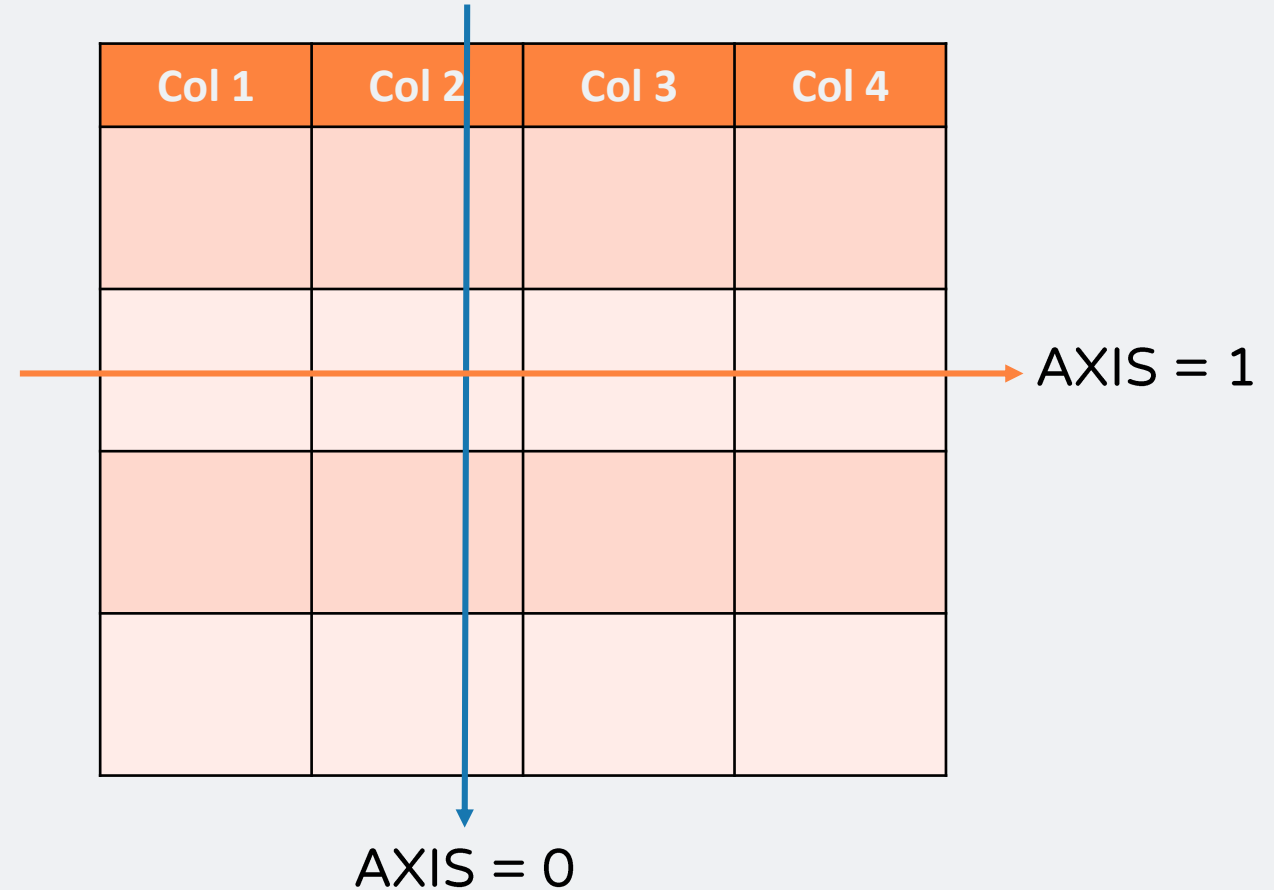
	Valor del Contrato	Valor de pago adelantado	Valor Pendiente de Pago	Valor Pagado
0	22.28	0.00	22.28	0.00
1	52.08	0.00	52.08	0.00
2	12.25	0.00	12.25	0.00
3	63.96	0.00	63.96	0.00
4	13.88	0.00	13.88	0.00
...
661630	25.23	0.00	25.23	0.00
661631	0.00	0.00	0.00	0.00
661632	764.35	0.00	764.35	0.00
661633	21.40	0.00	21.40	0.00
661634	5.73	0.00	3.66	2.07

[661635 rows x 4 columns]

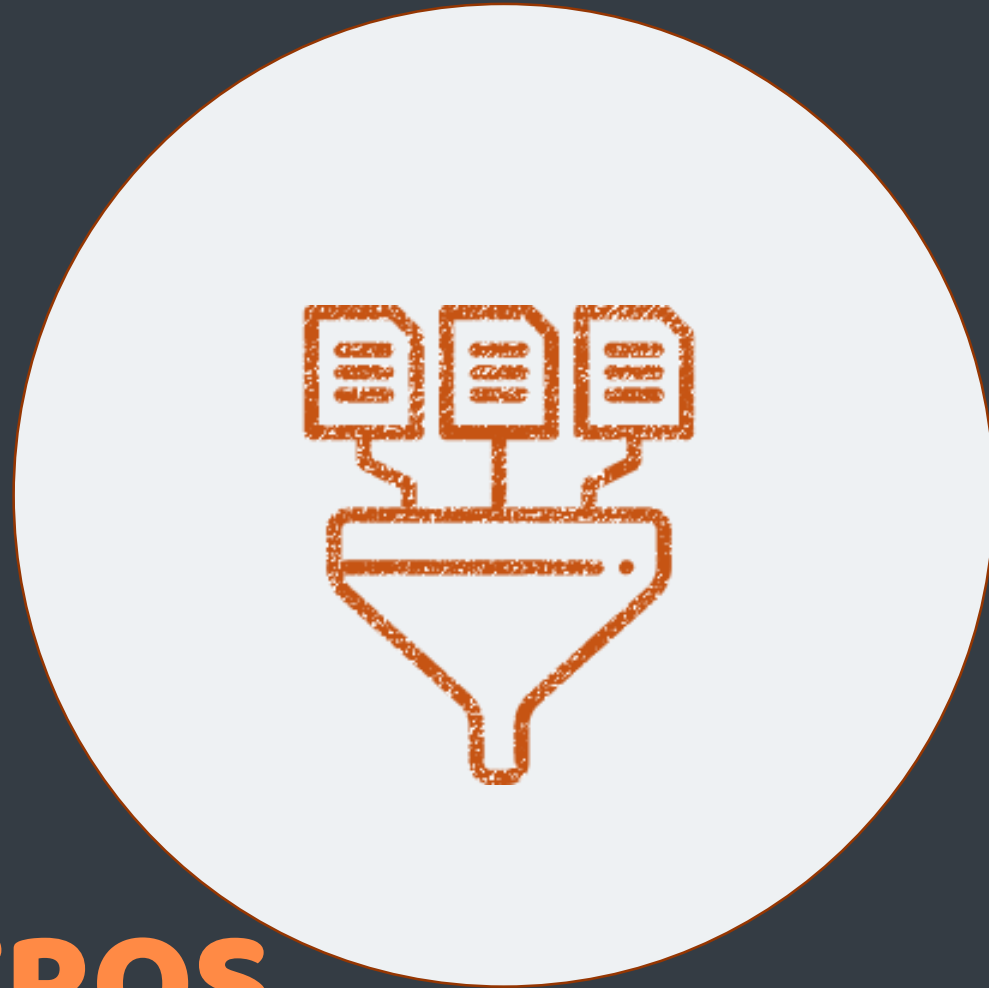
sum, prod, div

OPERACIONES AGREGADAS

```
In [737]: valores.sum(axis=1)
Out[737]:
0          44,558,988.00
1       104,150,322.00
2       24,500,000.00
3      127,926,000.00
4       27,751,200.00
...
661630    50,453,348.00
661631         0.00
661632  1,528,709,398.00
661633    42,790,320.00
661634   11,454,000.00
Length: 661635, dtype: float64
```



sum, prod, div



SELECCIÓN Y FILTROS SOBRE DATAFRAMES

ORDENAMIENTO

In [1245]: categoricos

Out[1245]:

	Sector	Estado	Contrato	Fecha de Firma	Es Pyme
0	Ley de Justicia	En ejecución		2020-05-28	True
1	Ambiente y Desarrollo Sostenible	Activo		NaT	True
2	Ambiente y Desarrollo Sostenible	Activo		2019-02-22	False
3	No aplica/No pertenece	Modificado		2019-01-30	False
4	Salud y Protección Social	Modificado		2020-02-19	False
...
661630	Inclusión Social y Reconciliación	Activo		2019-01-17	False
661631	Tecnologías de la Información y las Comunicaci...	Borrador		NaT	False
661632	Servicio Público	Activo		2019-11-26	True
661633	Salud y Protección Social	Activo		2020-08-12	False
661634	Planeación	cedido		2018-08-10	False

[661635 rows x 4 columns]

sort_values

`ordenados = categoricos.sort_values('Estado Contrato')`

In [1246]: ordenados

Out[1246]:

	Sector	Estado	Contrato	Fecha de Firma	Es Pyme
183364	defensa	Activo		2020-05-12	False
200690	Servicio Público	Activo		2020-09-22	False
200692	deportes	Activo		2018-03-17	False
200694	Educación Nacional	Activo		2019-02-26	False
200695	Servicio Público	Activo		2020-01-29	False
...
274824	Relaciones Exteriores	terminado		2018-05-11	True
274807	Educación Nacional	terminado		2018-01-04	False
274792	defensa	terminado		2018-01-23	False
274874	defensa	terminado		2018-03-01	False
607386	Servicio Público	terminado		2018-01-10	False

[661635 rows x 4 columns]

Los valores del índice no coinciden con las posiciones

SELECCIÓN DE REGISTROS POR POSICIÓN

iloc: integer-based location

```
In [1254]: ordenados.iloc[5]
```

```
Out[1254]:
```

```
Sector                Salud y Protección Social
Estado Contrato      Activo
Fecha de Firma       2019-08-15 00:00:00
Es Pyme              True
Name: 200696, dtype: object
```

```
In [1255]: ordenados.iloc[5:10]
```

```
Out[1255]:
```

	Sector	Estado	Contrato	Fecha de Firma	Es Pyme
200696	Salud y Protección Social		Activo	2019-08-15	True
200698	defensa		Activo	2019-03-05	True
200700	agricultura		Activo	2020-01-28	False
200701	No aplica/No pertenece		Activo	2019-01-18	False
200702	Inclusión Social y Reconciliación		Activo	2019-02-18	False

SELECCIÓN DE REGISTROS POR ÍNDICE O ETIQUETA

loc: label-based location

```
In [1253]: ordenados.loc[5]
Out[1253]:
Sector                Inclusión Social y Reconciliación
Estado Contrato      Cancelado
Fecha de Firma        NaT
Es Pyme               False
Name: 5, dtype: object
```

```
In [783]: ordenados.loc[5:10, ['Sector', 'Es Pyme']]
Out[783]:
```

	Sector	Es Pyme
5	Inclusión Social y Reconciliación	False
212831	defensa	True
146774	Ambiente y Desarrollo Sostenible	True
628922	Servicio Público	False
97490	No aplica/No pertenece	False
...
426159	Salud y Protección Social	False
270281	Industria	False
270316	Salud y Protección Social	False
270323	Inclusión Social y Reconciliación	False
10	Servicio Público	False

[317787 rows x 2 columns]

```
In [1257]: ordenados.loc[4:9]
Out[1257]:
Empty DataFrame
Columns: [Sector, Estado Contrato, Fecha de Firma, Es Pyme]
Index: []
```

No hay ningún registro entre las etiquetas 4 y 9: debe ser que la etiqueta 4 aparece después de la etiqueta 9

SELECCIÓN CON CONDICIONES

```
In [823]: muestra = ordenados.iloc[0:4]
```

```
In [824]: muestra
```

```
Out[824]:
```

	Sector	Estado	Contrato	Fecha de Firma	Es Pyme
183364	defensa		Activo	2020-05-12	False
200690	Servicio Público		Activo	2020-09-22	False
200692	deportes		Activo	2018-03-17	False
200694	Educación Nacional		Activo	2019-02-26	False

¿Conservar registro?

True
False
False
True



```
In [825]: muestra[[True, False, False, True]]
```

```
Out[825]:
```

	Sector	Estado	Contrato	Fecha de Firma	Es Pyme
183364	defensa		Activo	2020-05-12	False
200694	Educación Nacional		Activo	2019-02-26	False

SELECCIÓN CON CONDICIONES

```
In [1273]: categoricos['Estado Contrato'] == 'Activo'
Out[1273]:
0      False
1       True
2       True
3      False
4      False
...
661630   True
661631  False
661632   True
661633   True
661634  False
Name: Estado Contrato, Length: 661635, dtype: bool
```

```
In [1274]: categoricos[categoricos['Estado Contrato'] == 'Activo']
Out[1274]:
```

	Sector	Estado Contrato	Fecha de Firma	Es Pyme
1	Ambiente y Desarrollo Sostenible	Activo	NaT	True
2	Ambiente y Desarrollo Sostenible	Activo	2019-02-22	False
6	No aplica/No pertenece	Activo	2019-02-15	True
8	Inclusión Social y Reconciliación	Activo	2020-01-09	False
9	No aplica/No pertenece	Activo	2020-01-08	False
...
661625	Inclusión Social y Reconciliación	Activo	2018-01-26	False
661626	Trabajo	Activo	2020-02-01	False
661630	Inclusión Social y Reconciliación	Activo	2019-01-17	False
661632	Servicio Público	Activo	2019-11-26	True
661633	Salud y Protección Social	Activo	2020-08-12	False

[254172 rows x 4 columns]

OPERACIONES BOOLEANAS ENTRE CONDICIONES



Es importante no olvidar los paréntesis para cada condición

```
categoricos[(categoricos['Estado Contrato'] == 'Activo') |
             (categoricos['Estado Contrato'] == 'En ejecución') ]
```

& : and
| : or

	Sector	Estado Contrato	Fecha de Firma	Es Pyme
0	Ley de Justicia	En ejecución	2020-05-28	True
1	Ambiente y Desarrollo Sostenible	Activo	NaT	True
2	Ambiente y Desarrollo Sostenible	Activo	2019-02-22	False
6	No aplica/No pertenece	Activo	2019-02-15	True
8	Inclusión Social y Reconciliación	Activo	2020-01-09	False
...
661626	Trabajo	Activo	2020-02-01	False
661628	Trabajo	En ejecución	2020-07-17	False
661630	Inclusión Social y Reconciliación	Activo	2019-01-17	False
661632	Servicio Público	Activo	2019-11-26	True
661633	Salud y Protección Social	Activo	2020-08-12	False

[343892 rows x 4 columns]

OPERACIONES BOOLEANAS ENTRE CONDICIONES

```
contratos_febrero = categoricos[(categoricos['Fecha de Firma'] >= '2020-02-01')
                                & (categoricos['Fecha de Firma'] < '2020-03-01')]
```

& : and
| : or

In [1276]: contratos_febrero

Out[1276]:

	Sector	Estado Contrato	Fecha de Firma	Es Pyme
4	Salud y Protección Social	Modificado	2020-02-19	False
25	Salud y Protección Social	Modificado	2020-02-03	False
32	Salud y Protección Social	Activo	2020-02-14	False
95	Servicio Público	Activo	2020-02-10	False
127	interior	Activo	2020-02-13	False
...
661545	Cultura	Modificado	2020-02-07	True
661595	Educación Nacional	Activo	2020-02-13	True
661599	Trabajo	Modificado	2020-02-10	True
661616	No aplica/No pertenece	Activo	2020-02-27	False
661626	Trabajo	Activo	2020-02-01	False

[49035 rows x 4 columns]

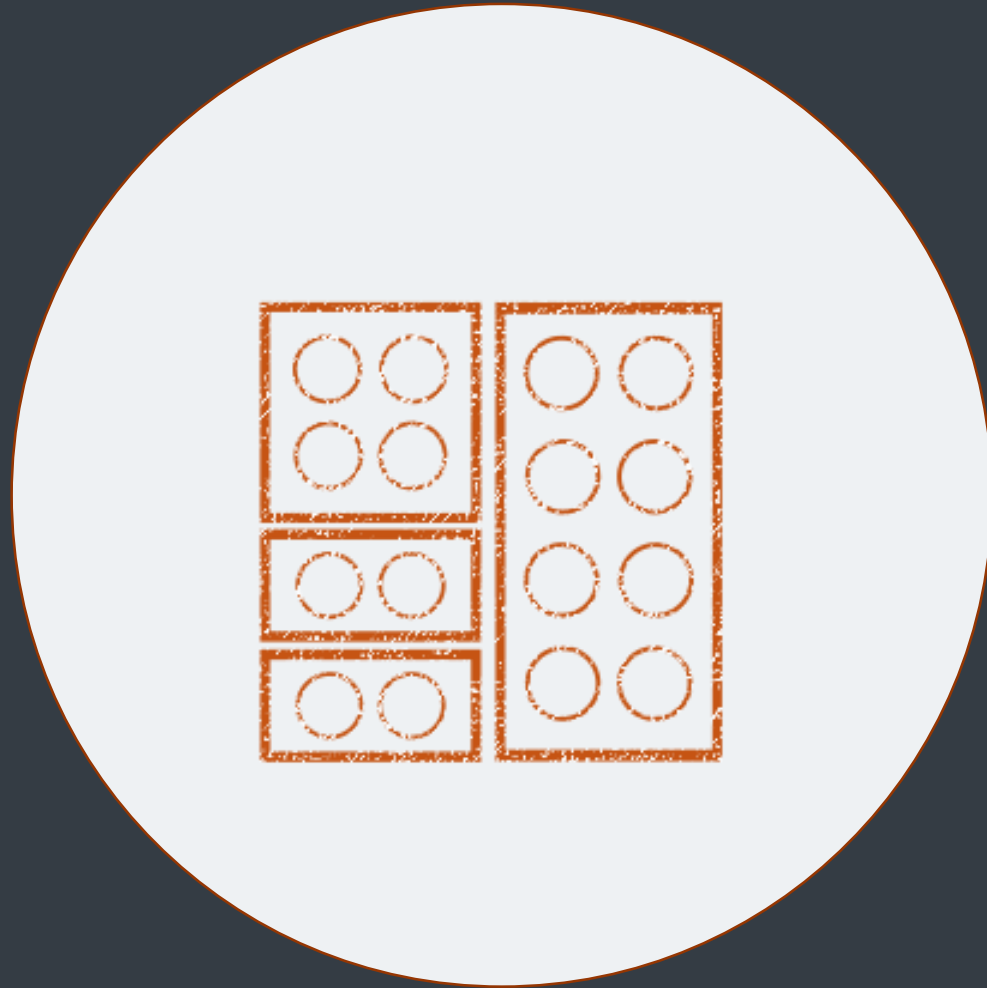
FUNCIÓN ISIN

```
In [1287]: categoricos[categoricos['Sector'].isin(['Servicio Público', 'deportes'])]
```

```
Out[1287]:
```

	Sector	Estado Contrato	Fecha de Firma	Es Pyme
10	Servicio Público	enviado Proveedor	NaT	False
24	deportes	enviado Proveedor	NaT	False
42	Servicio Público	Activo	2020-08-19	False
43	Servicio Público	En ejecución	2020-05-12	False
54	Servicio Público	Activo	2018-11-29	False
...
661606	Servicio Público	En ejecución	2020-06-08	False
661609	Servicio Público	Borrador	NaT	False
661619	Servicio Público	Activo	2019-03-07	False
661629	Servicio Público	Cerrado	2018-03-23	False
661632	Servicio Público	Activo	2019-11-26	True

```
[96687 rows x 4 columns]
```



AGRUPAMIENTOS

AGRUPAMIENTOS

	Sector	Rama	Valor del Contrato	Valor Pagado
549182	Industria	Ejecutivo	1,560,614	0
522143	Salud y Protección Social	Corporación Autónoma	17,036,200	8,745,249
409430	Servicio Público	Ejecutivo	28,700,000	0
220705	Inclusión Social y Reconciliación	Ejecutivo	98,012,280	49,006,140
615551	Presidencia de la República	Ejecutivo	30,803,275	0
74672	Trabajo	Ejecutivo	46,866,000	29,346,000
548854	Cultura	Ejecutivo	34,600,000	0
142874	Salud y Protección Social	Ejecutivo	3,871,680	0
408132	Vivienda, Ciudad y Territorio	Ejecutivo	1,545,000	1,545,000
408889	Ambiente y Desarrollo Sostenible	Corporación Autónoma	55,110,000	0

AGRUPAMIENTOS: POR RAMA

	Sector	Rama	Valor del Contrato	Valor Pagado
549182	Industria	Ejecutivo	1,560,614	0
522143	Salud y Protección Social	Corporación Autónoma	17,036,200	8,745,249
409430	Servicio Público	Ejecutivo	28,700,000	0
220705	Inclusión Social y Reconciliación	Ejecutivo	98,012,280	49,006,140
615551	Presidencia de la República	Ejecutivo	30,803,275	0
74672	Trabajo	Ejecutivo	46,866,000	29,346,000
548854	Cultura	Ejecutivo	34,600,000	0
142874	Salud y Protección Social	Ejecutivo	3,871,680	0
408132	Vivienda, Ciudad y Territorio	Ejecutivo	1,545,000	1,545,000
408889	Ambiente y Desarrollo Sostenible	Corporación Autónoma	55,110,000	0

AGRUPAMIENTOS

Grupo		Sector	Valor del Contrato	Valor Pagado
Corporación Autónoma	522143	Salud y Protección Social	17,036,200	8,745,249
	408889	Ambiente y Desarrollo Sostenible	55,110,000	0
Ejecutivo	549182	Industria	1,560,614	0
	409430	Servicio Público	28,700,000	0
	220705	Inclusión Social y Reconciliación	98,012,280	49,006,140
	615551	Presidencia de la República	30,803,275	0
	74672	Trabajo	46,866,000	29,346,000
	548854	Cultura	34,600,000	0
	142874	Salud y Protección Social	3,871,680	0
	408132	Vivienda, Ciudad y Territorio	1,545,000	1,545,000

AGRUPAMIENTOS

```
por_rama = filtradas.groupby('Rama')
```

```
In [1480]: len(por_rama)
```

```
Out[1480]: 4
```

```
In [1481]: for name, group in por_rama:  
...:         print(name, len(group))  
...:
```

```
Corporación Autónoma 57663
```

```
Ejecutivo 506947
```

```
Judicial 6854
```

```
Legislativo 1890
```

```
ejecutivo = por_rama.get_group('Ejecutivo')
```

groupby get_group

OPERACIONES SOBRE GRUPOS

```
In [1493]: por_rama['Valor Pagado']
```

```
Out[1493]: <pandas.core.groupby.generic.SeriesGroupBy object at 0x7fb8d4e09e50>
```

```
In [1494]: por_rama['Valor Pagado'].mean()
```

```
Out[1494]:
```

```
Rama
```

```
Corporación Autónoma    12,418,392.33
```

```
Ejecutivo                16,418,190.92
```

```
Judicial                 6,922,114.29
```

```
Legislativo              1,294,457.49
```

```
Name: Valor Pagado, dtype: float64
```

```
In [1495]: por_rama['Valor Pagado'].describe()
```

```
Out[1495]:
```

	count	mean	std	min	25%	50%	75%	max
Rama								
Corporación Autónoma	57,663.00	12,418,392.33	130,198,128.85	0.00	0.00	0.00	4,400,000.00	11,143,070,215.00
Ejecutivo	506,947.00	16,418,190.92	416,333,253.76	0.00	0.00	0.00	0.00	92,710,323,877.00
Judicial	6,854.00	6,922,114.29	58,192,539.99	0.00	0.00	0.00	0.00	2,318,434,558.00
Legislativo	1,890.00	1,294,457.49	10,018,152.12	0.00	0.00	0.00	0.00	378,950,000.00

Funciones agregadas: count, sum, mean, ...

GRUPOS MÚLTIPLES

```
multiple = filtradas.groupby(['Rama', 'Sector'])
```

```
In [1514]: for name, group in multiple:
...:     print(name, len(group))
...:
('Corporación Autónoma', 'Ambiente y Desarrollo Sostenible') 7290
('Corporación Autónoma', 'Ciencia Tecnología') 1787
('Corporación Autónoma', 'Cultura') 1117
('Corporación Autónoma', 'Educación Nacional') 2252
('Corporación Autónoma', 'Hacienda y Crédito Público') 2866
('Corporación Autónoma', 'Industria') 2030
('Corporación Autónoma', 'Información Estadística') 3730
('Corporación Autónoma', 'Ley de Justicia') 349
('Corporación Autónoma', 'Minas y Energía') 1025
('Corporación Autónoma', 'No aplica/No pertenece') 6979
('Corporación Autónoma', 'Planeación') 4434
('Corporación Autónoma', 'Relaciones Exteriores') 455
('Corporación Autónoma', 'Salud y Protección Social') 14653
('Corporación Autónoma', 'Servicio Público') 1827
('Corporación Autónoma', 'Tecnologías de la Información y las Comunicaciones') 910
('Corporación Autónoma', 'Trabajo') 233
('Corporación Autónoma', 'Transporte') 330
('Corporación Autónoma', 'Vivienda, Ciudad y Territorio') 2307
('Corporación Autónoma', 'agricultura') 116
('Corporación Autónoma', 'defensa') 2939
('Corporación Autónoma', 'deportes') 25
('Corporación Autónoma', 'interior') 9
('Ejecutivo', 'Ambiente y Desarrollo Sostenible') 19443
('Ejecutivo', 'Ciencia Tecnología') 219
('Ejecutivo', 'Cultura') 22919
('Ejecutivo', 'Educación Nacional') 21693
('Ejecutivo', 'Hacienda y Crédito Público') 8670
('Ejecutivo', 'Inclusión Social y Reconciliación') 34318
```

```
In [1516]: len(multiple)
Out[1516]: 56
```

GRUPOS MÚLTIPLES

In [1522]: multiple.sum()

Out[1522]:

		Valor del Contrato	Valor Pagado
Rama	Sector		
Corporación Autónoma	Ambiente y Desarrollo Sostenible	499,728,421,278.00	61,602,222,617.00
	Ciencia Tecnología	237,656,581,074.00	28,639,030,929.00
	Cultura	58,500,168,567.00	6,409,907,104.00
	Educación Nacional	145,014,301,283.00	53,837,810,793.00
	Hacienda y Crédito Público	71,478,145,821.00	16,152,542,580.00
...
Judicial	interior	155,060,695.00	0.00
Legislativo	Ley de Justicia	239,152,436.00	0.00
	No aplica/No pertenece	52,513,844,554.00	2,440,658,009.00
	Salud y Protección Social	68,249,908,306.00	5,866,650.00
	Servicio Público	249,060,451.00	0.00

[56 rows x 2 columns]

In [1532]: multiple.apply(max)

Out[1532]:

Rama	Sector	Nombre Entidad	JARDÍN BOTÁNICO JOSÉ CELESTINO MUTIS
Corporación Autónoma	Ambiente y Desarrollo Sostenible	Departamento	Valle del Cauca
		Ciudad	Yopal
		Destino Gasto	No Definido
		Estado Contrato	terminado
	
Legislativo	Servicio Público	Valor de pago adelantado	0
		Valor Pendiente de Pago	9,063,911
		Valor Pagado	0.00
		Días Adicionados	0
		fechas corregidas	2020-12-30 00:00:00

Length: 1050, dtype: object

MÁS INFORMACIÓN

DataFrames

Análisis de DataFrames

Agrupamientos

Funciones agregadas

https://pandas.pydata.org/docs/user_guide/dsintro.html#dataframe

https://pandas.pydata.org/pandas-docs/stable/user_guide/basics.html#descriptive-statistics

https://pandas.pydata.org/docs/user_guide/groupby.html