

Instala libloteas

```
In [1]: # Manipulación
import pandas as pd
import numpy as np

# Gráficas
import seaborn as sns
import matplotlib.pyplot as plt
```

Lee dataset, primero de tabulador y luego de plantilla de trabajadores de base

```
In [2]: fd_tab = pd.read_excel('./01_tabuladores_2019_2020_2021_sindicato_abril_2022
```

```
In [3]: fd_plan = pd.read_excel('./03_plantilla_trabajadores_de_base.xlsx', header=0)
```

```
In [4]: fd_tab.head()
```

```
Out[4]:
```

	NIVEL	TABULADOR BRUTO	RECONOCIMIENTO MENSUAL BRUTO	ASIGNACIÓN ADICIONAL BRUTO	TOTAL MENSUAL BRUTO
0	29.5	25347	28629	-	53976
1	85.7	25347	-	11853	37200
2	85.6	25347	-	10175	35522
3	20.5	25347	-	1153	26500
4	20.0	23140	-	463	23603

```
In [5]: fd_tab.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 21 entries, 0 to 20
Data columns (total 5 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   NIVEL                                21 non-null     float64
1   TABULADOR BRUTO                      21 non-null     int64
2   RECONOCIMIENTO MENSUAL BRUTO         21 non-null     object
3   ASIGNACIÓN ADICIONAL BRUTO           21 non-null     object
4   TOTAL MENSUAL BRUTO                  21 non-null     int64
dtypes: float64(1), int64(2), object(2)
memory usage: 968.0+ bytes
```

```
In [6]: list(fd_tab)
```

```
Out[6]: ['NIVEL',
'TABULADOR BRUTO',
'RECONOCIMIENTO MENSUAL BRUTO',
'ASIGNACIÓN ADICIONAL BRUTO',
```

```
'TOTAL MENSUAL BRUTO']
```

```
In [7]: fd_tab['NIVEL'].value_counts()
```

```
Out[7]: 29.5    3
      85.7    3
      85.6    3
      20.5    3
      20.0    3
      18.0    3
      16.0    3
      Name: NIVEL, dtype: int64
```

Hay tres niveles salariales iguales con diferente sueldo en la lista, por eso hay $7 \times 3 = 21$ registros, suponemos que cada set es de cada año, 2019, 2020 y 2021

```
In [8]: fd_tab[['NIVEL', 'TOTAL MENSUAL BRUTO']]
```

```
Out[8]:
```

	NIVEL	TOTAL MENSUAL BRUTO
0	29.5	53976
1	85.7	37200
2	85.6	35522
3	20.5	26500
4	20.0	23603
5	18.0	14275
6	16.0	6577
7	29.5	55811
8	85.7	38465
9	85.6	36730
10	20.5	27401
11	20.0	24406
12	18.0	14760
13	16.0	6801
14	29.5	57597
15	85.7	39696
16	85.6	37905
17	20.5	28278
18	20.0	25187
19	18.0	15232
20	16.0	7019

```
In [9]: tab_2019 = fd_tab.iloc[0:7]
tab_2019
```

```
Out[9]:
```

	NIVEL	TABULADOR BRUTO	RECONOCIMIENTO MENSUAL BRUTO	ASIGNACIÓN ADICIONAL BRUTO	TOTAL MENSUAL BRUTO
0	29.5	25347	28629	-	53976
1	85.7	25347	-	11853	37200
2	85.6	25347	-	10175	35522
3	20.5	25347	-	1153	26500
4	20.0	23140	-	463	23603
5	18.0	13995	-	280	14275
6	16.0	6577	-	-	6577

```
In [10]: tab_2020 = fd_tab.iloc[7:14]
tab_2020
```

```
Out[10]:
```

	NIVEL	TABULADOR BRUTO	RECONOCIMIENTO MENSUAL BRUTO	ASIGNACIÓN ADICIONAL BRUTO	TOTAL MENSUAL BRUTO
7	29.5	26064	29747	-	55811
8	85.7	26064	-	12401	38465
9	85.6	26064	-	10666	36730
10	20.5	26064	-	1337	27401
11	20.0	23603	-	803	24406
12	18.0	14275	-	485	14760
13	16.0	6801	-	-	6801

```
In [11]: tab_2021 = fd_tab.iloc[14:21]
tab_2021
```

```
Out[11]:
```

	NIVEL	TABULADOR BRUTO	RECONOCIMIENTO MENSUAL BRUTO	ASIGNACIÓN ADICIONAL BRUTO	TOTAL MENSUAL BRUTO
14	29.5	26886	30711	-	57597
15	85.7	26886	-	12810	39696
16	85.6	26886	-	11019	37905
17	20.5	26886	-	1392	28278
18	20.0	24406	-	781	25187
19	18.0	14760	-	472	15232

```

In [12]: nivel_salarial = fd_tab['NIVEL'].unique()
          print(nivel_salarial)

[29.5 85.7 85.6 20.5 20.  18.  16. ]

```

```

In [13]: fd_plan.head()

```

```

Out[13]:

```

	Núm.	Número UACM	Plaza	Nombre Completo	Nivel Salarial	Puesto	Nombramiento	Asignación de Tiempo
0	1	2005060173	1027	ABURTO LOPEZ ROSALINDA	20.0	ASISTENTE ADMINISTRATIVO	NO APLICA	NO APLICA
1	2	2007050067	1707	ACEVEDO MELO INDIRA SYBILA	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEMPO COMPLETO
2	3	2007090257	1847	ACEVES SALAZAR JOSE GUILLERMO	20.0	CHEF	NO APLICA	NO APLICA
3	4	2008010001	1946	AGUILAR ALCANTARA ADRIANA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	NO APLICA
4	5	2003010001	456	AGUILAR ALMANZA AARON ERNESTO	20.5	ASISTENTE DE DISEÑO GRAFICO ESPECIALIZADO	NO APLICA	NO APLICA

```

In [14]: list(fd_plan)

```

```

Out[14]: ['Núm.',
          'Número UACM',
          'Plaza',
          'Nombre Completo',
          'Nivel Salarial',
          'Puesto',
          'Nombramiento',
          'Asignación de Tiempo',
          'Coordinación',
          'Dirección',
          'Departamento',
          'Sede Plantilla',
          'Academia',
          'Estatus Plantilla',
          'Situación en Plantilla',
          'Contrato Colectivo de Trabajo',

```

```
'Régimen de Contratación',
'Fecha Ingreso',
'Fecha Inicio - Último Movimiento',
```

In [15]: `fd_plan.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1760 entries, 0 to 1759
Data columns (total 20 columns):
#   Column                                     Non-Null Count  Dtype
---  -
0   Núm.                                     1760 non-null   int64
1   Número UACM                             1760 non-null   int64
2   Plaza                                    1760 non-null   object
3   Nombre Completo                         1760 non-null   object
4   Nivel Salarial                          1760 non-null   float64
5   Puesto                                  1760 non-null   object
6   Nombramiento                            1760 non-null   object
7   Asignación de Tiempo                    1760 non-null   object
8   Coordinación                            1760 non-null   object
9   Dirección                               1760 non-null   object
10  Departamento                             1760 non-null   object
11  Sede Plantilla                          1760 non-null   object
12  Academia                                1760 non-null   object
13  Estatus Plantilla                       1760 non-null   object
14  Situación en Plantilla                   1760 non-null   object
15  Contrato Colectivo de Trabajo            1760 non-null   object
16  Régimen de Contratación                 1760 non-null   object
17  Fecha Ingreso                            1760 non-null   object
18  Fecha Inicio - Último Movimiento          1760 non-null   object
19  Fecha Final- Último Movimiento            1760 non-null   object
dtypes: float64(1), int64(2), object(17)
memory usage: 275.1+ KB
```

In [16]: `fd_plan['Nombre Completo'].head()`

```
Out[16]: 0      ABURTO LOPEZ ROSALINDA
1      ACEVEDO MELO INDIRA SYBILA
2      ACEVES SALAZAR JOSE GUILLERMO
3      AGUILAR ALCANTARA ADRIANA
4      AGUILAR ALMANZA AARON ERNESTO
Name: Nombre Completo, dtype: object
```

In [17]: `fd_plan['Nombre Completo'] = fd_plan['Nombre Completo'].str.replace('[^\w\s]',`

```
/tmp/ipykernel_16665/1173231437.py:1: FutureWarning: The default value of reg
ex will change from True to False in a future version.
    fd_plan['Nombre Completo'] = fd_plan['Nombre Completo'].str.replace('[^\w\
s]', '')
```

In [18]: `fd_plan['Plaza'].head()`

```
Out[18]: 0      1027
1      1707
2      1847
3      1946
```

```
4      456
Name: Plaza. dtvne: object
```

```
In [19]: fd_plan['Puesto'].value_counts()
puestos = fd_plan['Puesto'].value_counts()
puestos.to_csv('./cuenta_puestos.csv')
```

```
In [20]: fd_plan['Nombramiento'].value_counts()
```

```
Out[20]: NO APLICA
1541
SIN NOMBRAMIENTO
192
SUTUACM: SECRETARIO DE ASUNTOS LABORALES (TITULAR)
1
ENLACE DE LA COORDINACION DE OBRAS Y CONSERVACION EN EL PLANTEL CASA LIBERTAD
1
SUTUACM: SECRETARIA DE ACCION Y PREVISION SOCIAL\n(SUPLENTE)
1
RESPONSABLE DEL\nPROGRAMA DE AÑO SABATICO
1
SUTUACM: SECRETARIA DE ESTUDIOS Y PROYECTO UNIVERSITARIO (SUPLENTE)
1
SUTUACM: SECRETARIA DE ORGANIZACION (SUPLENTE)
1
SUTUACM: SECRETARIA DE ESTUDIOS Y PROYECTO UNIVERSITARIO (TITULAR)
1
ENCARGADA DEL AREA DE DESARROLLO DE SISTEMAS
1
SUTUACM: SECRETARIA DE FINANZAS (SECRETARIA)
1
ENLACE DE LA COORDINACION DE OBRAS Y CONSERVACION EN EL PLANTEL DEL\nVALLE
1
RESPONSABLE DEL POSGRADO DE ESTUDIOS SOBRE LA CIUDAD
1
SUTUACM: SUBSECRETARIA DE ASUNTOS LABORALES DEL SECTOR\nADMINISTRATIVO
1
SUTUACM: SECRETARIO DE ESTUDIOS Y PROYECTO UNIVERSITARIO (SUPLENTE)
1
ENCARGADO DEL PROGRAMA GALATEA
1
AUXILIAR DE INVENTARIOS
1
SUTUACM: COMISION DE FINANZAS (PRIMER\nVOCAL)
1
ENLACE DE LA COORDINACION DE OBRAS Y CONSERVACION EN EL PLANTEL CENTRO HISTOR
ICO      1
ENLACE DE LA COORDINACION DE OBRAS Y CONSERVACION EN EL PLANTEL SAN LORENZO T
EZONCO   1
RESPONSABLE DEL PROGRAMA DE PROMOCION CULTURAL EN EL PLANTEL DEL VALLE
1
RESPONSABLE DEL PROGRAMA DE\nINTEGRACION
1
SUTUACM: SECRETARIO DE ACCION Y PREVISION SOCIAL\n(SUPLENTE)
1
ENLACE DE LA COORDINACION DE OBRAS Y CONSERVACION EN EL PLANTEL\nCUAUTEPEC
1
```

SUTUACM: SECRETARIA DE COMUNICACION Y RELACIONES EXTERNAS (SUPLENTE)

1

SUTUACM: SECRETARIA DE ORGANIZACION

1

DIRECTOR DE ANDAMIOS

1

RESPONSABLE DE CENTRO VLADY

1

SUTUACM: SECRETARIO DE ORGANIZACION (TITULAR)

1

Name: Nombramiento, dtype: int64

```
In [21]: fd_plan['Nombramiento'] = fd_plan['Nombramiento'].str.replace('[^\w\s]', '')

/tmp/ipykernel_16665/1987717594.py:1: FutureWarning: The default value of reg
ex will change from True to False in a future version.
  fd_plan['Nombramiento'] = fd_plan['Nombramiento'].str.replace('[^\w\s]',
  '')
```

```
In [22]: fd_plan['Nombramiento'].head()
```

```
Out[22]: 0    NO APLICA
1    NO APLICA
2    NO APLICA
3    NO APLICA
4    NO APLICA
Name: Nombramiento, dtype: object
```

necesitamos identificar al personal de confianza del que no lo es y 'Nombramiento' y 'Puesto' no ayudan a hacerlo, tal vez si tubieran el campo 'Contrato Colectivo de Trabajo' los identificaríamos

```
In [23]: fd_plan['Régimen de Contratación'].value_counts()
```

```
Out[23]: LABORAL    1760
Name: Régimen de Contratación, dtype: int64
```

```
In [24]: fd_plan['Contrato Colectivo de Trabajo'].value_counts()
```

```
Out[24]: BASE    1760
Name: Contrato Colectivo de Trabajo, dtype: int64
```

```
In [25]: es_coord = fd_plan['Puesto'].str.contains('COORDINADOR')
puestos_coor = fd_plan[es_coord].value_counts()
puestos_coor.to_csv('./cuenta_puest_coord.csv')
```

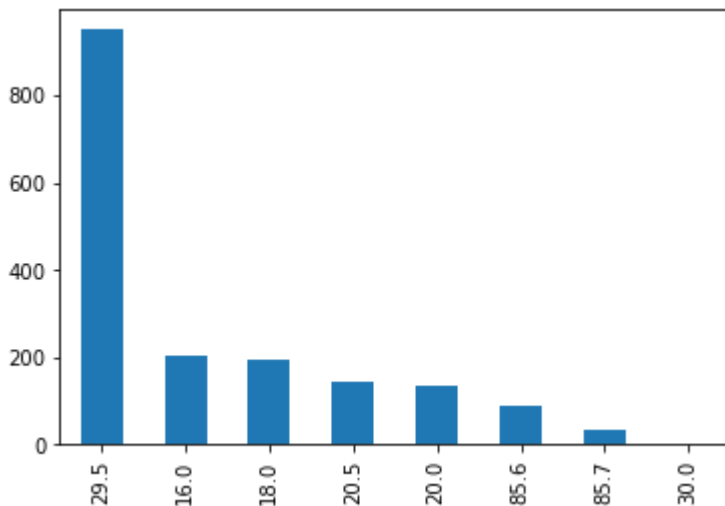
```
In [26]: fd_plan.groupby(['Nivel Salarial'])['Nivel Salarial'].count()
#fd_plan['Nivel Salarial'].value_counts()
```

```
Out[26]: Nivel Salarial
16.0    206
18.0    195
20.0    137
20.5    146
29.5    951
```

```
30.0      1
85.6     88
85.7     36
..      ..
```

```
In [27]: fd_plan['Nivel Salarial'].value_counts().plot.bar()
```

```
Out[27]: <AxesSubplot:>
```



```
In [28]: puestos_nivel = fd_plan.groupby(['Nivel Salarial', 'Puesto'])['Nivel Salarial']
puestos_nivel.to_csv('./cuenta_puest_nivel.csv')
```

la distribucion de puestos determina hacia donde se carga la nomina

```
In [29]: fd_plan[~fd_plan['Nivel Salarial'].isin(nivel_salarial)]
```

```
Out[29]:
```

Núm.	Número UACM	Plaza	Nombre Completo	Nivel Salarial	Puesto	Nombramiento	Asignación de Tiempo
549	550	2017080618	T17002	GARCIA TORRES NEYRA MARGARITA	30.0	AUXILIAR ADMINISTRATIVO	NO APLICA
						NO APLICA	NO APLIC

solo hay una persona que no tiene un nivel salarial estipulado en el tabulador

```
In [30]: fd_plan.groupby(['Nivel Salarial', 'Puesto'])
```

```
Out[30]: <pandas.core.groupby.generic.DataFrameGroupBy object at 0x7f1f58530a60>
```

```
In [31]: fd_plan['Fecha Ingreso'].head()
```

```
Out[31]: 0    2005-06-16 00:00:00
1    2007-04-16 00:00:00
2    2007-09-16 00:00:00
```



```

3    2008-01-01 00:00:00
4    2003-01-01 00:00:00
Name: Fecha Ingreso, dtype: object

```

```
In [32]: fd_plan['Fecha Ingreso'].value_counts()
```

```

Out[32]: NO FECHA          125
2008-01-01 00:00:00      99
2002-01-01 00:00:00      79
2007-09-01 00:00:00      70
2006-09-16 00:00:00      61
...
2006-03-16 00:00:00       1
2011-02-01 00:00:00       1
2007-07-20 00:00:00       1
2016-10-16 00:00:00       1
2016-03-03 00:00:00       1
Name: Fecha Ingreso, Length: 200, dtype: int64

```

```
In [33]: vacantes = fd_plan[fd_plan['Nombre Completo'] == 'ZX VACANTE']
vacantes.to_csv('./vacantes_uacm.csv')
vacantes.info()
```

```

<class 'pandas.core.frame.DataFrame'>
Int64Index: 125 entries, 1635 to 1759
Data columns (total 20 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   Núm.                                125 non-null    int64
1   Número UACM                        125 non-null    int64
2   Plaza                              125 non-null    object
3   Nombre Completo                    125 non-null    object
4   Nivel Salarial                     125 non-null    float64
5   Puesto                             125 non-null    object
6   Nombramiento                       125 non-null    object
7   Asignación de Tiempo               125 non-null    object
8   Coordinación                       125 non-null    object
9   Dirección                          125 non-null    object
10  Departamento                       125 non-null    object
11  Sede Plantilla                     125 non-null    object
12  Academia                           125 non-null    object
13  Estatus Plantilla                  125 non-null    object
14  Situación en Plantilla              125 non-null    object
15  Contrato Colectivo de Trabajo      125 non-null    object
16  Régimen de Contratación           125 non-null    object
17  Fecha Ingreso                      125 non-null    object
18  Fecha Inicio - Último Movimiento   125 non-null    object
19  Fecha Final- Último Movimiento     125 non-null    object
dtypes: float64(1), int64(2), object(17)
memory usage: 20.5+ KB

```

quitamos las 125 vacantes del dataset original para que no nos haga ruido

```
In [34]: fd_plan.drop(fd_plan[fd_plan['Nombre Completo'] == 'ZX VACANTE'].index, inplace=True)
```

In [35]:

fd_plan.info()

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1635 entries, 0 to 1634
Data columns (total 20 columns):
#   Column                                     Non-Null Count  Dtype
---  -
0   Núm.                                     1635 non-null   int64
1   Número UACM                             1635 non-null   int64
2   Plaza                                    1635 non-null   object
3   Nombre Completo                         1635 non-null   object
4   Nivel Salarial                          1635 non-null   float64
5   Puesto                                   1635 non-null   object
6   Nombramiento                            1635 non-null   object
7   Asignación de Tiempo                    1635 non-null   object
8   Coordinación                            1635 non-null   object
9   Dirección                               1635 non-null   object
10  Departamento                             1635 non-null   object
11  Sede Plantilla                          1635 non-null   object
12  Academia                                 1635 non-null   object
13  Estatus Plantilla                       1635 non-null   object
14  Situación en Plantilla                   1635 non-null   object
15  Contrato Colectivo de Trabajo            1635 non-null   object
16  Régimen de Contratación                 1635 non-null   object
17  Fecha Ingreso                            1635 non-null   object
18  Fecha Inicio - Último Movimiento          1635 non-null   object
19  Fecha Final- Último Movimiento            1635 non-null   object
dtypes: float64(1), int64(2), object(17)
memory usage: 268.2+ KB
```

In [36]:

fd_plan['Fecha Ingreso'].value_counts()

Out[36]:

```
2008-01-01    99
2002-01-01    79
2007-09-01    70
2006-09-16    61
2009-01-01    59
..
2017-05-18     1
2006-02-16     1
2007-06-16     1
2015-10-26     1
2007-05-16     1
Name: Fecha Ingreso, Length: 199, dtype: int64
```

In [37]:

fd_plan['Fecha Inicio - Último Movimiento'].head()

Out[37]:

```
0    2015-02-01 00:00:00
1    2021-08-03 00:00:00
2          INDETERMINADO
3    2019-01-09 00:00:00
4    2016-04-25 00:00:00
Name: Fecha Inicio - Último Movimiento, dtype: object
```

In [38]:

fd_plan['Fecha Inicio - Último Movimiento'].value_counts()

```
Out[38]: INDETERMINADO      543
         2018-08-01 00:00:00    61
         2021-02-01 00:00:00    47
         2019-01-10 00:00:00    45
         2016-08-01 00:00:00    44
         ...
         2016-03-01 00:00:00     1
         2020-01-21 00:00:00     1
         2019-09-13 00:00:00     1
         2017-05-16 00:00:00     1
         2019-01-14 00:00:00     1
         Name: Fecha Inicio - Último Movimiento, Length: 247, dtype: int64
```

```
In [39]: fd_plan['Fecha Final- Último Movimiento'].head()
```

```
Out[39]: 0    INDETERMINADO
         1    INDETERMINADO
         2    INDETERMINADO
         3    INDETERMINADO
         4    INDETERMINADO
         Name: Fecha Final- Último Movimiento, dtype: object
```

```
In [40]: fd_plan['Fecha Final- Último Movimiento'].value_counts()
```

```
Out[40]: INDETERMINADO      1543
         2022-06-30 00:00:00    14
         2023-01-09 00:00:00    10
         2022-12-31 00:00:00     5
         2022-08-08 00:00:00     4
         2022-07-31 00:00:00     4
         2023-01-22 00:00:00     4
         2022-04-30 00:00:00     3
         2021-12-31 00:00:00     2
         2023-01-15 00:00:00     2
         2022-07-28 00:00:00     2
         2022-08-09 00:00:00     2
         2023-01-06 00:00:00     2
         2022-05-03 00:00:00     2
         2022-09-30 00:00:00     2
         2010-02-28 00:00:00     1
         2023-11-17 00:00:00     1
         2022-11-10 00:00:00     1
         2022-05-31 00:00:00     1
         2023-01-31 00:00:00     1
         2022-08-31 00:00:00     1
         2024-05-31 00:00:00     1
         2022-01-14 00:00:00     1
         2024-02-07 00:00:00     1
         2023-01-19 00:00:00     1
         2022-08-28 00:00:00     1
         2023-03-01 00:00:00     1
         2020-05-31 00:00:00     1
         2022-04-11 00:00:00     1
         2014-02-16 00:00:00     1
         2022-07-29 00:00:00     1
         2022-08-01 00:00:00     1
         2022-01-16 00:00:00     1
```

```

2022-08-04 00:00:00      1
2022-04-15 00:00:00      1
2022-08-15 00:00:00      1
2022-08-07 00:00:00      1
2022-07-25 00:00:00      1
2021-07-31 00:00:00      1
2022-10-01 00:00:00      1
2023-03-02 00:00:00      1
2022-04-01 00:00:00      1
2014-08-01 00:00:00      1
2023-01-17 00:00:00      1
2023-11-23 00:00:00      1
2022-05-17 00:00:00      1
2022-06-20 00:00:00      1
2023-12-08 00:00:00      1
2022-01-31 00:00:00      1

```

voy a tomar 'Fecha Ingreso' como el campo fecha para poder agrupar al personal de la tabla y así poder calcular el monto mensual de las cuotas, suponiendo que todos son de base

In [41]:

```
print(fd_plan.dtypes)
```

```

Núm.                int64
Número UACM         int64
Plaza               object
Nombre Completo     object
Nivel Salarial      float64
Puesto              object
Nombramiento        object
Asignación de Tiempo object
Coordinación        object
Dirección           object
Departamento       object
Sede Plantilla      object
Academia            object
Estatus Plantilla   object
Situación en Plantilla object
Contrato Colectivo de Trabajo object
Régimen de Contratación object
Fecha Ingreso       object
Fecha Inicio - Último Movimiento object
Fecha Final- Último Movimiento object
dtype: object

```

cambiar la columna de 'Fecha Ingreso' a fecha

In [42]:

```
fd_plan['Fecha Ingreso'] = pd.to_datetime(fd_plan['Fecha Ingreso'])
print(fd_plan.dtypes)
```

```

Núm.                int64
Número UACM         int64
Plaza               object
Nombre Completo     object
Nivel Salarial      float64
Puesto              object
Nombramiento        object
Asignación de Tiempo object

```

```

Coordinación          object
Dirección              object
Departamento          object
Sede Plantilla         object
Academia               object
Estatus Plantilla     object
Situación en Plantilla object
Contrato Colectivo de Trabajo object
Régimen de Contratación object
Fecha Ingreso          datetime64[ns]
Fecha Inicio - Último Movimiento object
Fecha Final- Último Movimiento object

```

```
In [43]: fd_plan.head()
```

```
Out[43]:
```

	Núm.	Número UACM	Plaza	Nombre Completo	Nivel Salarial	Puesto	Nombramiento	Asignación de Tiempo
0	1	2005060173	1027	ABURTO LOPEZ ROSALINDA	20.0	ASISTENTE ADMINISTRATIVO	NO APLICA	NO APLICA
1	2	2007050067	1707	ACEVEDO MELO INDIRA SYBILA	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEMPO (COMPLETO
2	3	2007090257	1847	ACEVES SALAZAR JOSE GUILLERMO	20.0	CHEF	NO APLICA	NO APLICA
3	4	2008010001	1946	AGUILAR ALCANTARA ADRIANA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	NO APLICA
4	5	2003010001	456	AGUILAR ALMANZA AARON ERNESTO	20.5	ASISTENTE DE DISEÑO GRAFICO ESPECIALIZADO	NO APLICA	NO APLICA

```
In [44]: fd_plan.sort_values(by='Fecha Ingreso', inplace=True)
```

```
In [45]: fd_plan.loc[fd_plan['Nombre Completo'].str.contains('ALCANTARA', case=False)]
```

```
Out[45]:
```

	Núm.	Número UACM	Plaza	Nombre Completo	Nivel Salarial	Puesto	Nombramiento	Asignación de Tiempo
--	------	-------------	-------	-----------------	----------------	--------	--------------	----------------------

	Núm.	Número UACM	Plaza	Nombre Completo	Nivel Salarial	Puesto	Nombramiento	Asignación de Tiempo
34	35	2006010003	1119	ALCANTARA PINEDA MARIA ANGELICA	85.6	COORDINADOR DE PROYECTO B DE EVALUACION	NO APLICA	NO APLICA
1431	1432	2007090248	2237	SENTIES ALCANTARA ERIKA JAZMIN	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEMPO COMPLETO

In [46]:

fd_plan

Out[46]:

	Núm.	Número UACM	Plaza	Nombre Completo	Nivel Salarial	Puesto	Nombramiento	Asign de Ti
630	631	2002030146	39	GUERRA JAIME MARTHA	29.5	ASESOR	NO APLICA	NO AI
1482	1483	2002010120	336	TORRES BUSTILLOS ROSA MARIA	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIE COMP
288	289	2003010008	492	CHAVEZ FRAUSTO KARINA\nMARIBEL	20.5	ASISTENTE ACADEMICO B	NO APLICA	NO AI
988	989	2002010099	520	MONTES MARGALLI MARA ITZEL GEORGINA	20.5	ASISTENTE ACADEMICO B	NO APLICA	NO AI
81	82	2002010070	1792	ANDRADE BUTZONITCH MARIANO MARCOS	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIE COMP
...
320	321	2021090684	1796	CORONA SANCHEZ MANUEL ANTONIO	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIE COMP
273	274	2021090683	1255	CASTRO RAMIREZ JOSE	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIE COMP
684	685	2022010110	151	HERNANDEZ HERNANDEZ GERARDO	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIE COMP

	Núm.	Número UACM	Plaza	Nombre Completo	Nivel Salarial	Puesto	Nombramiento	Asign de Ti
1509	1510	2022020370	4070	VALADEZ BLANCO EDGAR OCTAVIO	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIE COMP

1596	1597	2022020369	4074	VIVANCO TORRES	29.5	PROFESOR	NO APLICA	TIE
------	------	------------	------	----------------	------	----------	-----------	-----

ya tenemos el dataset ordenado por fecha, ahora vamos a sacar grupos de personas, tiene su error este dataset porque no hay fecha de terminacion (por que se fue, murio o se le corrio de la universidad), quedan 1635 empleados. Vamos a limpiar la fuente de datos en nombres y puesto.

```
In [47]: fd_tab.columns = fd_tab.columns.str.lower().str.replace(' ', '_')
fd_tab.head()
```

	nivel	tabulador_bruto	reconocimiento_mensual_bruto	asignación_adicional_bruto	total_mensual
0	29.5	25347	28629	-	
1	85.7	25347	-	11853	
2	85.6	25347	-	10175	
3	20.5	25347	-	1153	
4	20.0	23140	-	463	

```
In [48]: import datetime as dt
```

```
In [49]: fd_plan.columns = fd_plan.columns.str.lower().str.replace(' ', '_')
#fd_plan["nombre_completo"] = fd_plan["nombre_completo"].head().str.split().a
#fd_plan["puesto"] = fd_plan["puesto"].head().str.split().apply(lambda rec_li
fd_plan.head()
```

	núm.	número_uacm	plaza	nombre_completo	nivel_salarial	puesto	nombramiento
630	631	2002030146	39	GUERRA JAIME MARTHA	29.5	ASESOR	NO APLICA
1482	1483	2002010120	336	TORRES BUSTILLOS ROSA MARIA	29.5	PROFESOR INVESTIGADOR	NO APLICA
288	289	2003010008	492	CHAVEZ FRAUSTO KARINA\nMARIBEL	20.5	ASISTENTE ACADEMICO B	NO APLICA
988	989	2002010099	520	MONTES MARGALLI MARA ITZEL GEORGINA	20.5	ASISTENTE ACADEMICO B	NO APLICA

núm.	número_uacm	plaza	nombre_completo	nivel_salarial	puesto	nombramiento
------	-------------	-------	-----------------	----------------	--------	--------------

ANDRADE

```
In [50]: emp_2019 = fd_plan[fd_plan['fecha_ingreso'].dt.year <= 2019]
len(emp_2019)
```

Out[50]: 1602

```
In [51]: emp_2019.loc[emp_2019['nombre_completo'].str.contains('ALCANTARA', case=True)]
```

Out[51]:

núm.	número_uacm	plaza	nombre_completo	nivel_salarial	puesto	nombramiento
------	-------------	-------	-----------------	----------------	--------	--------------

34	35	2006010003	1119	ALCANTARA PINEDA MARIA ANGELICA	85.6	COORDINADOR DE PROYECTO B DE EVALUACION	NO APLICA
1431	1432	2007090248	2237	SENTIES ALCANTARA ERIKA JAZMIN	29.5	PROFESOR INVESTIGADOR	NO APLICA
3	4	2008010001	1946	AGUILAR ALCANTARA ADRIANA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA
105	106	2008010011	1931	ARIAS ALCANTARA MARIA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA

```
In [52]: emp_2018 = fd_plan[fd_plan['fecha_ingreso'].dt.year <= 2018]
len(emp_2018)
```

Out[52]: 1602

no se contrato a nadie en 2019

```
In [53]: emp_2020 = fd_plan[fd_plan['fecha_ingreso'].dt.year <= 2020]
len(emp_2020)
len(emp_2020) - len(emp_2019)
```

Out[53]: 17

```
In [54]: emp_2021 = fd_plan[fd_plan['fecha_ingreso'].dt.year <= 2021]
len(emp_2021)
len(emp_2021) - len(emp_2020)
```


Out[54]: 13

```
In [55]: (fd_plan[fd_plan['fecha_ingreso'].dt.year == 2020]).to_csv('./nuevos_2020.csv')
         (fd_plan[fd_plan['fecha_ingreso'].dt.year == 2021]).to_csv('./nuevos_2021.csv')
```

```
In [56]: emp_2019.groupby(['nivel_salarial', 'puesto'])['nivel_salarial'].count()
```

```
Out[56]: nivel_salarial  puesto
16.0                AUXILIAR DE INTENDENCIA
185
18.0                ASESOR ACADEMICO D
1
                ASISTENTE ACADEMICO D
24
                ASISTENTE ADMINISTRATIVO
1
                AUXILIAR ADMINISTRATIVO
41
...
85.7                COORDINADOR DE PROYECTOS A EN PRESTADORES DE SERVICIOS PROFES
IONALES            1 COORDINADOR DE PROYECTOS A EN PROTECCION CIVIL
1
                COORDINADOR DE PROYECTOS A EN SISTEMAS DE INFORMACION
1
                RESPONSABLE DE BIBLIOTECA A
1
                RESPONSABLE DE DESARROLLO
1
Name: nivel_salarial, Length: 163, dtype: int64
```

```
In [57]: lista_nivel_salarial_x_puesto = emp_2019.groupby(['nivel_salarial', 'puesto'])
         lista_nivel_salarial_x_puesto.head()
         #sns.countplot(lista_nivel_salarial_x_puesto, color='gray')
         #sns.histplot(data=lista_nivel_salarial_x_puesto, x="nivel_salarial", hue='co
         #sns.(data=lista_nivel_salarial_x_puesto, x="nivel_salarial", hue='counts', c
         #fig = px.scatter(lista_nivel_salarial_x_puesto, x="nivel_salarial", y="puest
```

```
Out[57]:
```

		counts
nivel_salarial	puesto	
16.0	AUXILIAR DE INTENDENCIA	185
18.0	ASESOR ACADEMICO D	1
	ASISTENTE ACADEMICO D	24
	ASISTENTE ADMINISTRATIVO	1
	AUXILIAR ADMINISTRATIVO	41

estandarizar los nombres de las columnas en los dataframes para poder hacer el merge
volver a generar los tabuladores por año

```
In [58]: tab_2019 = fd_tab.iloc[0:7]
tab_2020 = fd_tab.iloc[7:14]
tab_2021 = fd_tab.iloc[14:21]
```

```
In [59]: tab_2019
```

```
Out[59]:
```

	nivel	tabulador_bruto	reconocimiento_mensual_bruto	asignación_adicional_bruto	total_mensual
0	29.5	25347	28629	-	
1	85.7	25347	-	11853	
2	85.6	25347	-	10175	
3	20.5	25347	-	1153	
4	20.0	23140	-	463	
5	18.0	13995	-	280	
6	16.0	6577	-	-	

```
In [60]: tab_2020
```

```
Out[60]:
```

	nivel	tabulador_bruto	reconocimiento_mensual_bruto	asignación_adicional_bruto	total_mensual
7	29.5	26064	29747	-	
8	85.7	26064	-	12401	
9	85.6	26064	-	10666	
10	20.5	26064	-	1337	
11	20.0	23603	-	803	
12	18.0	14275	-	485	
13	16.0	6801	-	-	

cambiar el nombre de la columna 'nivel salarial' de fd_plan

```
In [61]: fd_plan = fd_plan.rename(columns={'nivel_salarial': 'nivel'})
fd_plan.head()
```

```
Out[61]:
```

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asigna
630	631	2002030146	39	GUERRA JAIME MARTHA	29.5	ASESOR	NO APLICA	
1482	1483	2002010120	336	TORRES BUSTILLOS ROSA MARIA	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEN

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asigna
288	289	2003010008	492	CHAVEZ FRAUSTO KARINA\nMARIBEL	20.5	ASISTENTE ACADEMICO B	NO APLICA	
988	989	2002010099	520	MONTES MARGALLI MARA ITZEL GEORGINA	20.5	ASISTENTE ACADEMICO B	NO APLICA	
				ANDRADE				

In [62]:

```
emp_2019 = emp_2019.rename(columns={'nivel_salarial': 'nivel'})
# limpia la columna de nombre
#emp_2019["nombre_completo"] = emp_2019["nombre_completo"].head().str.split()
emp_2019.head()
```

Out[62]:

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asigna
630	631	2002030146	39	GUERRA JAIME MARTHA	29.5	ASESOR	NO APLICA	
1482	1483	2002010120	336	TORRES BUSTILLOS ROSA MARIA	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEM
288	289	2003010008	492	CHAVEZ FRAUSTO KARINA\nMARIBEL	20.5	ASISTENTE ACADEMICO B	NO APLICA	
988	989	2002010099	520	MONTES MARGALLI MARA ITZEL GEORGINA	20.5	ASISTENTE ACADEMICO B	NO APLICA	
81	82	2002010070	1792	ANDRADE BUTZONITCH MARIANO MARCOS	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEM

In [63]:

```
fd_plan_2019 = emp_2019.merge(tab_2019)
fd_plan_2019.head()
```

Out[63]:

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asignación
0	631	2002030146	39	GUERRA JAIME MARTHA	29.5	ASESOR	NO APLICA	
1	1483	2002010120	336	TORRES BUSTILLOS ROSA MARIA	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEMPO
2	82	2002010070	1792	ANDRADE BUTZONITCH MARIANO	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEMPO

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asignación
				MARCOS				
				RAMOS				
3	1230	2002010107	290	VILLEGAS PEDRO ARTURO	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEMPO
				AMENEYRO				
4	76	2002080174	101	AGUILAR HECTOR ALONSO	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEMPO

In [64]:

```
#fd_plan_2019 = fd_plan_2019[fd_plan_2019['fecha_ingreso'].dt.year <= 2019]
fd_plan_2019.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1601 entries, 0 to 1600
Data columns (total 24 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   núm.                                1601 non-null   int64
1   número_uacm                        1601 non-null   int64
2   plaza                              1601 non-null   object
3   nombre_completo                    1601 non-null   object
4   nivel                              1601 non-null   float64
5   puesto                             1601 non-null   object
6   nombramiento                       1601 non-null   object
7   asignación_de_tiempo               1601 non-null   object
8   coordinación                       1601 non-null   object
9   dirección                          1601 non-null   object
10  departamento                        1601 non-null   object
11  sede_plantilla                     1601 non-null   object
12  academia                           1601 non-null   object
13  estatus_plantilla                   1601 non-null   object
14  situación_en_plantilla              1601 non-null   object
15  contrato_colectivo_de_trabajo       1601 non-null   object
16  régimen_de_contratación            1601 non-null   object
17  fecha_ingreso                       1601 non-null   datetime64[ns]
18  fecha_inicio_último_movimiento      1601 non-null   object
19  fecha_final_último_movimiento       1601 non-null   object
20  tabulador_bruto                     1601 non-null   int64
21  reconocimiento_mensual_bruto       1601 non-null   object
22  asignación_adicional_bruto          1601 non-null   object
23  total_mensual_bruto                 1601 non-null   int64
dtypes: datetime64[ns](1), float64(1), int64(4), object(18)
memory usage: 312.7+ KB
```

In [65]:

```
fd_plan_2019.head()
```

Out[65]:

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asignación
0	631	2002030146	39	GUERRA JAIME MARTHA	29.5	ASESOR	NO APLICA	

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asignación
1	1483	2002010120	336	TORRES BUSTILLOS ROSA MARIA	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEMPO
2	82	2002010070	1792	ANDRADE BUTZONITCH MARIANO MARCOS	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEMPO
3	1230	2002010107	290	RAMOS VILLEGAS PEDRO ARTURO	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEMPO
4	70	2002010174	101	AMENEYRO AGUILAR	29.5	PROFESOR	NO APLICA	TIEMPO

```
In [66]: nom_2019_mes = fd_plan_2019['total_mensual_bruto'].sum()
nom_2019_mes
```

```
Out[66]: 61694674
```

```
In [67]: sind_2019_mes = nom_2019_mes * 0.01
sind_2019_mes
```

```
Out[67]: 616946.74
```

```
In [68]: fd_plan_2019.sample(n = 10)
```

```
Out[68]:
```

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asignación
1457	21	2008010003	1930	AGUILAR RODRIGUEZ MA ANITA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	
833	1511	2009020431	4093	VALDES PALACIOS JOSE ALBERTO	29.5	PROFESOR INVESTIGADOR	NO APLICA	TII
901	287	2002110255	950.1	CHAVEZ CORREA CLAUDIA LAURA	20.5	ASISTENTE ACADEMICO B	NO APLICA	
101	667	2002080191	203	HERNANDEZ AGUILAR JOSE JUAN	29.5	PROFESOR INVESTIGADOR	NO APLICA	TII
457	744	2005110252	839	ISAAC NOVOA YURI YERYE	29.5	PROFESOR INVESTIGADOR	NO APLICA	TII
424	203	2005060174	1028	BURGUEÑO ANGULO KARLA	29.5	ASESOR ACADEMICO A	NO APLICA	

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asign
1542	463	2009010346	2313	FLORES FRANCO MARIA DEL ROCIO	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	
61	722	2002010091	210	HERRERA UNDA ROGELIO EDUARDO	29.5	PROFESOR INVESTIGADOR	SIN NOMBRAMIENTO	TII
1268	1161	2006120264	1381	PIÑA CANCINO FLORINA	18.0	ASISTENTE ACADEMICO D	NO APLICA	
				BELLIDO VALERIO		ASISTENTE		

la suma es correcta para cualquier registro del data set

```
In [69]: emp_2020 = emp_2020.rename(columns={'nivel_salarial': 'nivel'})
fd_plan_2020 = emp_2020.merge(tab_2020)
#fd_plan_2020 = fd_plan_2020[fd_plan_2020['fecha_ingreso'].dt.year <= 2020]
#fd_plan_2020.head()
#len(fd_plan_2020)
nom_2020_mes = fd_plan_2020['total_mensual_bruto'].sum()
nom_2020_mes
```

Out[69]: 64741020

```
In [70]: #emp_2020.loc[emp_2020['nombre_completo'].str.contains('ALCANTARA', case=False)]
fd_plan_2020.loc[fd_plan_2020['nombre_completo'].str.contains('ALCANTARA', ca
```

```
Out[70]:
```

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asigna
671	1432	2007090248	2237	SENTIES ALCANTARA ERIKA JAZMIN	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEM
1207	35	2006010003	1119	ALCANTARA PINEDA MARIA ANGELICA	85.6	COORDINADOR DE PROYECTO B DE EVALUACION	NO APLICA	
1477	4	2008010001	1946	AGUILAR ALCANTARA ADRIANA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	
1497	106	2008010011	1931	ARIAS ALCANTARA MARIA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	

4 rows × 24 columns

```
In [71]: sind_2020_mes = nom_2020_mes * 0.01
sind_2020_mes
```

Out[71]: 647410.20000000001

In [72]: `fd_plan_2020.sample(n = 10)`

Out[72]:

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asi
697	53	2007100308	1835	ALVARADO HERNANDEZ ROJAS VERONICA	29.5	PROFESOR INVESTIGADOR	NO APLICA	.
1068	1098	2004110262	1603	PACINDO VELASCO MOISES	20.0	ASISTENTE ADMINISTRATIVO	NO APLICA	
1170	1632	2002080213	445	ZEPEDA MONTROYA JORGE	85.6	ASISTENTE ACADEMICO A	NO APLICA	
221	412	2004040085	161	DURAN LIZARRAGA MA ELENA	29.5	PROFESOR INVESTIGADOR	NO APLICA	.
466	574	2006060038	1150	GOMEZ GONZALEZ ERIKA OLIVIA	29.5	PROFESOR INVESTIGADOR	SIN NOMBRAMIENTO	.
1550	337	2009010343	2310	CRUZ LAGUNA J FELIX	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	
750	562	2007100352	4022	GODINEZ GUTIERREZ MIGUEL ANGEL	29.5	PROFESOR INVESTIGADOR	NO APLICA	.
275	417	2004100195	1320	ELKISCH MARTINEZ MARIANA	29.5	PROFESOR INVESTIGADOR	NO APLICA	.
1186	775	2004040076	1534	JUAREZ HERNANDEZ SERGIO FERNANDO	85.6	RESPONSABLE DE REGISTRO ESCOLAR B	NO APLICA	
129	1482	2003060109	79	TORRES ABELAIRA BEATRIZ	29.5	ASESOR ACADEMICO A	NO APLICA	

10 rows × 24 columns

In [73]:

```

emp_2021 = emp_2021.rename(columns={'nivel_salarial': 'nivel'})
fd_plan_2021 = emp_2021.merge(tab_2021)
#fd_plan_2021 = fd_plan_2021[fd_plan_2021['fecha_ingreso'].dt.year <= 2021]
fd_plan_2021.head()
nom_2021_mes = fd_plan_2021['total_mensual_bruto'].sum()
nom_2021_mes

```

Out[73]: 67561551

```
In [74]: sind_2021_mes = nom_2021_mes * 0.01  
sind_2021_mes
```

```
Out[74]: 675615.51
```

```
In [75]: fd_plan_2021.sample(n = 10)
```

```
Out[75]:
```

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	a:
1308	993	2007050078	1192.1	MONTIEL SANCHEZ DAVID	18.0	TECNICO EN CONSERVACION	NO APLICA	
979	493	2005070190	951	GAINZAR MARTINEZ VICTOR	20.5	ASISTENTE ADMINISTRATIVO ESPECIALIZADO	NO APLICA	
293	98	2004100218	104	ARECHIGA CORDOBA ERNESTO	29.5	PROFESOR INVESTIGADOR	NO APLICA	
1042	779	2002010032	704	JUAREZ RUIZ BEATRIZ	20.0	BIBLIOTECOLOGO	NO APLICA	
1437	179	2005070189	1017	BLAS CHAVEZ VERONICA	85.7	COORDINADOR DE PROYECTOS A DE OBRAS	ENLACE DE LA COORDINACION DE OBRAS Y CONSERVAC...	
969	1163	2005050134	933	PIÑA GONZALEZ VALENTINA	20.5	ASISTENTE ACADEMICO B	NO APLICA	
1154	240	2007080134	1395	CARLOS AMAYA MARILU	20.0	ASISTENTE ADMINISTRATIVO	NO APLICA	
657	1605	2007090255	1754	YAÑEZ NAVARRO GUSTAVO	29.5	PROFESOR INVESTIGADOR	NO APLICA	
655	354	2007090265	1784	DAVILA JAUREGUI AMARANTO DE JESUS	29.5	PROFESOR INVESTIGADOR	NO APLICA	
1102	1019	2005030038	606	MOYA MARTINEZ PEDRO	20.0	TECNICO ELECTRICISTA	NO APLICA	

10 rows × 24 columns

```
In [76]: fd_plan_2019.groupby(['nivel', 'puesto'])['total_mensual_bruto'].sum()
```

```
Out[76]: nivel  puesto  
16.0      AUXILIAR DE INTENDENCIA
```



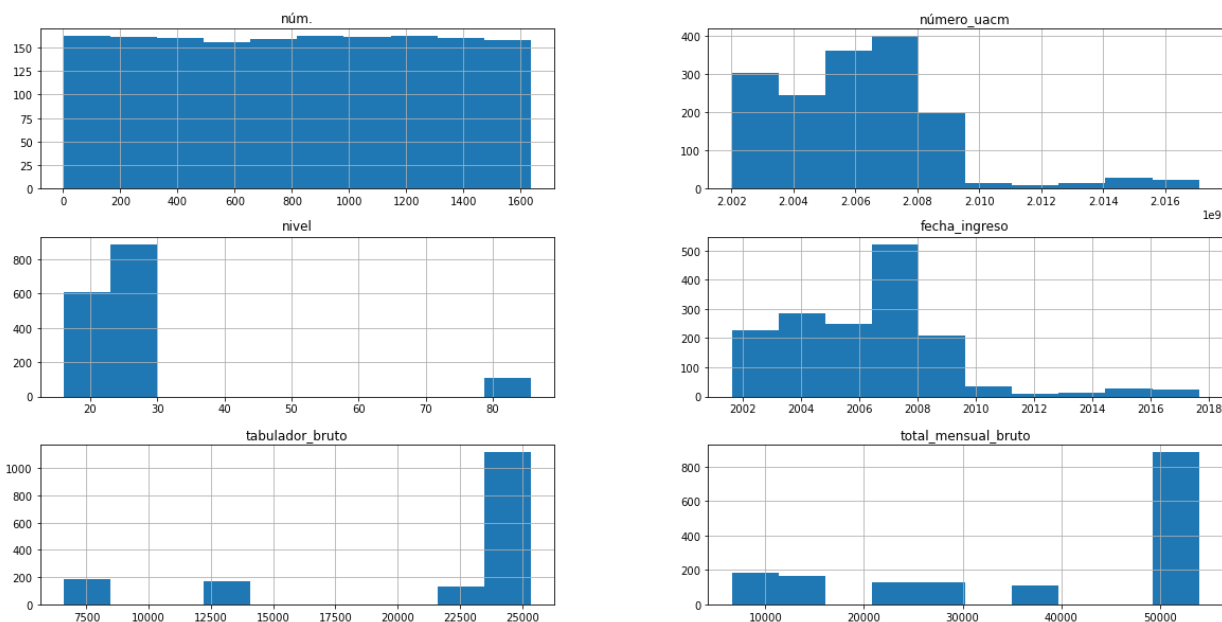
```

1216745
18.0  ASesor ACADEMICO D
14275
ASISTENTE ACADEMICO D
342600
ASISTENTE ADMINISTRATIVO
14275
AUXILIAR ADMINISTRATIVO
585275

...
85.7  COORDINADOR DE PROYECTOS A EN PRESTADORES DE SERVICIOS PROFESIONALES
37200
COORDINADOR DE PROYECTOS A EN PROTECCION CIVIL
37200
COORDINADOR DE PROYECTOS A EN SISTEMAS DE INFORMACION
37200
RESPONSABLE DE BIBLIOTECA A
37200
RESPONSABLE DE DESARROLLO
37200
Name: total mensual bruto, Length: 162, dtype: int64
salario por puesto de trabajo agrupado por nivel salarial

```

In [77]: `_ = fd_plan_2019.hist(figsize = (20,10))`



Vamos a revisar una quincena de 2022 para saber el padron y analizar la concordancia de los datos

```
fd_desc_012022 = pd.read_csv('./padron_sind_012022.csv') fd_desc_012022.info()
```

la lista de descuento base sindicalizada 806 y la lista de confianza descontada 262 que no pueden ser sindicalizados forman la base de descuentos, este ultimo deben fregresarle su dinero

joins con pandas, inner es intercepcion de conjuntos

In [79]:

```
# fd_plan_2021 = A, fd_desc_012022 = B, inner = merge por un campo, vamos a d
padron_sind_012022 = pd.merge(fd_plan_2021,fd_desc_012022,on='nombre_completo
len(padron_sind_012022)
lista_nombres_padron = padron_sind_012022['nombre_completo']
lista_nombres_padron
```

/tmp/ipykernel_16665/2614161066.py:2: FutureWarning: Passing 'suffixes' which cause duplicate columns {'régimen_de_contratación_y', 'asignación_adicional_bruto_y', 'coordinación_y', 'fecha_ingreso_y', 'tabulador_bruto_y', 'sede_plantilla_y', 'número_uacm_y', 'fecha_inicio_último_movimiento_y', 'departamento_y', 'reconocimiento_mensual_bruto_y', 'nombramiento_y', 'núm._y', 'fecha_final_último_movimiento_y', 'asignación_de_tiempo_y', 'academia_y', 'total_mensual_bruto_y', 'plaza_y', 'contrato_colectivo_de_trabajo_y', 'situación_en_plantilla_y', 'estatus_plantilla_y', 'nivel_y', 'dirección_y'} in the result is deprecated and will raise a MergeError in a future version.

```
padron_sind_012022 = pd.merge(fd_plan_2021,fd_desc_012022,on='nombre_completo')
```

Out[79]:

```
0          GUERRA JAIME MARTHA
1      TORRES BUSTILLOS ROSA MARIA
2      ANDRADE BUTZONITCH MARIANO MARCOS
3      AMENEYRO AGUILAR HECTOR ALONSO
4      VAZQUEZ LIMA SAMUEL
...
981      SAMANO REYES ESMERALDA PAZ
982      AGUILAR GONZALEZ MARCOS
983      MORENO CAMARA MIREYA MARISOL
984      ROSALES SILVA KAREN
985      SANDOVAL GONZALEZ JOSE RICARDO
Name: nombre_completo, Length: 986, dtype: object
```

In [80]:

```
padron_sind_012022.to_csv('./padron_sind_012022.csv')
```

In [81]:

```
# fd_plan_2021 = A, fd_desc_012022 = B, right = merge por un campo an B, vamos a d
no_desc_2022 = pd.merge(fd_plan_2021,fd_desc_012022,on='nombre_completo', how='right')
# falta quitar los duplicados del padron, estan en el padron pero no en la lista_nombres_padron
no_desc_2022 = no_desc_2022[~no_desc_2022['nombre_completo'].isin(lista_nombres_padron)]
len(no_desc_2022)
```

/tmp/ipykernel_16665/1706391419.py:2: FutureWarning: Passing 'suffixes' which cause duplicate columns {'régimen_de_contratación_y', 'asignación_adicional_bruto_y', 'coordinación_y', 'fecha_ingreso_y', 'tabulador_bruto_y', 'sede_plantilla_y', 'número_uacm_y', 'fecha_inicio_último_movimiento_y', 'departamento_y', 'reconocimiento_mensual_bruto_y', 'nombramiento_y', 'núm._y', 'fecha_final_último_movimiento_y', 'asignación_de_tiempo_y', 'academia_y', 'total_mensual_bruto_y', 'plaza_y', 'contrato_colectivo_de_trabajo_y', 'situación_en_plantilla_y', 'estatus_plantilla_y', 'nivel_y', 'dirección_y'} in the result is deprecated and will raise a MergeError in a future version.

```
no_desc_2022 = pd.merge(fd_plan_2021,fd_desc_012022,on='nombre_completo', how='left')
```

Out[81]:

645

In [82]:

```
no_desc_2022.to_csv('./no_estan_012022_desc_si_plan.csv')
```

```
In [83]: lista_nombres_no_desc = no_desc_2022['nombre_completo']
```

```
In [84]: # fd_plan_2021 = A, fd_desc_012022 = B, left = merge por un campo an B, vamo
no_plan_2022 = pd.merge(fd_plan_2021,fd_desc_012022,on='nombre_completo', how
# falta quitar los duplicados del padron, estan en la lista de descuentos per
no_plan_2022 = no_plan_2022[~no_plan_2022['nombre_completo'].isin(lista_nomb
len(no_plan_2022)
```

```
/tmp/ipykernel_16665/3595216351.py:2: FutureWarning: Passing 'suffixes' which
cause duplicate columns {'régimen_de_contratación_y', 'asignación_adicional_b
ruto_y', 'coordinación_y', 'fecha_ingreso_y', 'tabulador_bruto_y', 'sede_plan
tilla_y', 'número_uacm_y', 'fecha_inicio_último_movimiento_y', 'departament
o_y', 'reconocimiento_mensual_bruto_y', 'nombramiento_y', 'núm._y', 'fecha_fi
nal_último_movimiento_y', 'asignación_de_tiempo_y', 'academia_y', 'total_men
sual_bruto_y', 'plaza_y', 'contrato_colectivo_de_trabajo_y', 'situación_en_pl
antilla_y', 'estatus_plantilla_y', 'nivel_y', 'dirección_y'} in the result is
deprecated and will raise a MergeError in a future version.
```

```
no_plan_2022 = pd.merge(fd_plan_2021,fd_desc_012022,on='nombre_completo', h
ow='right')
```

```
Out[84]: 0
```

```
In [85]: no_plan_2022.to_csv('./estan_012022_desc_no_plan.csv')
```

```
In [86]: lista_nombres_no_plan = no_plan_2022['nombre_completo']
```

```
In [87]: #padron_sind_012022.merge(no_desc_2022, how = 'inner', indicator=False)
padron_sind_012022['nombre_completo'].equals(no_desc_2022['nombre_completo'])
```

```
Out[87]: False
```

```
In [ ]:
```

```
In [88]: fd_plan_2019.loc[fd_plan_2019['nombre_completo'].str.contains('ALCANTARA', ca
```

```
Out[88]:
```

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asigna
--	------	-------------	-------	-----------------	-------	--------	--------------	--------

671	1432	2007090248	2237	SENTIES ALCANTARA ERIKAZMIN	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEM
1190	35	2006010003	1119	ALCANTARA PINEDA MARIA ANGELICA	85.6	COORDINADOR DE PROYECTO B DE EVALUACION	NO APLICA	
1460	4	2008010001	1946	AGUILAR ALCANTARA ADRIANA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asigna
1480	106	2008010011	1931	ARIAS ALCANTARA MARIA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	

```
In [89]: fd_plan_2021.loc[fd_plan_2021['nombre_completo'].str.contains('ALCANTARA', ca
```

```
Out[89]:
```

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asigna
671	1432	2007090248	2237	SENTIES ALCANTARA ERIKA JAZMIN	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEM
1220	35	2006010003	1119	ALCANTARA PINEDA MARIA ANGELICA	85.6	COORDINADOR DE PROYECTO B DE EVALUACION	NO APLICA	
1490	4	2008010001	1946	AGUILAR ALCANTARA ADRIANA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	
1510	106	2008010011	1931	ARIAS ALCANTARA MARIA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	

4 rows × 24 columns

```
In [90]: fd_plan_2020.loc[fd_plan_2020['nombre_completo'].str.contains('ALCANTARA', ca
```

```
Out[90]:
```

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asigna
671	1432	2007090248	2237	SENTIES ALCANTARA ERIKA JAZMIN	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEM
1207	35	2006010003	1119	ALCANTARA PINEDA MARIA ANGELICA	85.6	COORDINADOR DE PROYECTO B DE EVALUACION	NO APLICA	
1477	4	2008010001	1946	AGUILAR ALCANTARA ADRIANA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	
1497	106	2008010011	1931	ARIAS ALCANTARA MARIA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	

4 rows × 24 columns

```
In [91]: fd_plan_2008 = fd_plan[fd_plan['fecha_ingreso'].dt.year == 2008]
fd_plan_2008.loc[fd_plan_2008['nombre_completo'].str.contains('ALCANTARA', ca
```

```
Out[91]:
```

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asignación
3	4	2008010001	1946	AGUILAR ALCANTARA ADRIANA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	
105	106	2008010011	1931	ARIAS ALCANTARA MARIA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	

```
In [92]: fd_plan_2019 = fd_plan[fd_plan['fecha_ingreso'].dt.year <= 2019]
fd_plan_2019.loc[fd_plan_2019['nombre_completo'].str.contains('ALCANTARA', ca
```

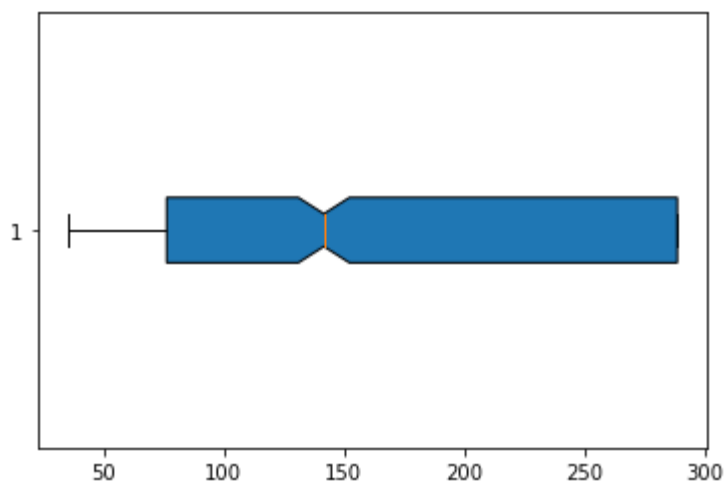
```
Out[92]:
```

	núm.	número_uacm	plaza	nombre_completo	nivel	puesto	nombramiento	asignación
34	35	2006010003	1119	ALCANTARA PINEDA MARIA ANGELICA	85.6	COORDINADOR DE PROYECTO B DE EVALUACION	NO APLICA	
1431	1432	2007090248	2237	SENTIES ALCANTARA ERIKA JAZMIN	29.5	PROFESOR INVESTIGADOR	NO APLICA	TIEM
3	4	2008010001	1946	AGUILAR ALCANTARA ADRIANA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	
105	106	2008010011	1931	ARIAS ALCANTARA MARIA	16.0	AUXILIAR DE INTENDENCIA	NO APLICA	

hay que limpiar mas la fuente

```
In [93]: desc_012022 = fd_desc_012022['descuento']
```

```
In [96]: #fig = plt.figure(figsize=(10, 7))
#ax = fig.add_subplot(111)
fig, ax = plt.subplots()
#ax.boxplot(desc_012022)
bp = ax.boxplot(desc_012022, patch_artist = True,
                 notch = 'True', vert = 0)
plt.show()
```



```
In [97]: desc_012022.describe()
```

```
Out[97]: count    986.000000
mean      172.657414
std       102.032499
min       35.130000
25%       76.240000
50%      141.530000
75%      288.270000
max       288.270000
Name: descuento, dtype: float64
```

```
In [99]: fd_plan_2020['total_mensual_bruto'].describe()
```

```
Out[99]: count      1618.000000
mean     40012.991347
std     19071.293349
min      6801.000000
25%     24406.000000
50%     55811.000000
75%     55811.000000
max     55811.000000
Name: total_mensual_bruto, dtype: float64
```

```
In [101]: 55811 * 0.01 / 2
```

```
Out[101]: 279.055
```

```
In [102]: 6801 * 0.01 / 2
```

```
Out[102]: 34.005
```

notemos que en el limite inferior da un valor esperado pero no es exacto y en el limite superior se acerca pero es menos exacto de los esperado, pero estan dentro de los limites esperados, los descuentos parecen correctos de acuerdo con los valores del tabulador

In [103... desc_012022.sum()

Out[103... 170240.21

In [105... fd_desc_012022.info()

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

RangeIndex: 986 entries, 0 to 985

Data columns (total 75 columns):

#	Column	Non-Null Count	Dtype
0	Unnamed: 0	986 non-null	int64
1	núm.	986 non-null	int64
2	número_uacm	986 non-null	int64
3	plaza	986 non-null	object
4	nombre_completo	986 non-null	object
5	nivel	986 non-null	float64
6	puesto_x	986 non-null	object
7	nombramiento	986 non-null	object
8	asignación_de_tiempo	986 non-null	object
9	coordinación	986 non-null	object
10	dirección	986 non-null	object
11	departamento	986 non-null	object
12	sede_plantilla	986 non-null	object
13	academia	986 non-null	object
14	estatus_plantilla	986 non-null	object
15	situación_en_plantilla	986 non-null	object
16	contrato_colectivo_de_trabajo	986 non-null	object
17	régimen_de_contratación	986 non-null	object
18	fecha_ingreso	986 non-null	object
19	fecha_inicio - último movimiento	986 non-null	object
20	fecha_final- último movimiento	986 non-null	object
21	tabulador_bruto	986 non-null	int64
22	reconocimiento_mensual_bruto	986 non-null	object
23	asignación_adicional_bruto	986 non-null	object
24	total_mensual_bruto	986 non-null	int64
25	Unnamed: 0.1	986 non-null	int64
26	núm._x	986 non-null	int64
27	número_uacm_x	986 non-null	int64
28	plaza_x	986 non-null	object
29	nivel_x	986 non-null	float64
30	puesto_y	986 non-null	object
31	nombramiento_x	986 non-null	object
32	asignación_de_tiempo_x	986 non-null	object
33	coordinación_x	986 non-null	object
34	dirección_x	986 non-null	object
35	departamento_x	986 non-null	object
36	sede_plantilla_x	986 non-null	object
37	academia_x	986 non-null	object
38	estatus_plantilla_x	986 non-null	object
39	situación_en_plantilla_x	986 non-null	object
40	contrato_colectivo_de_trabajo_x	986 non-null	object
41	régimen_de_contratación_x	986 non-null	object
42	fecha_ingreso_x	986 non-null	object
43	fecha_inicio - último movimiento_x	986 non-null	object
44	fecha_final- último movimiento_x	986 non-null	object

```

45 tabulador_bruto_x          986 non-null    int64
46 reconocimiento_mensual_bruto_x  986 non-null    object
47 asignación_adicional_bruto_x    986 non-null    object
48 total_mensual_bruto_x          986 non-null    int64
49 Unnamed: 0.1.1              986 non-null    int64
50 núm._y                      986 non-null    int64
51 número_uacm_y               986 non-null    int64
52 plaza_y                     986 non-null    object
53 nivel_y                      986 non-null    float64
54 puesto_x.1                  986 non-null    object
55 nombramiento_y              986 non-null    object
56 asignación_de_tiempo_y        986 non-null    object
57 coordinación_y              986 non-null    object
58 dirección_y                 986 non-null    object
59 departamento_y              986 non-null    object
60 sede_plantilla_y            986 non-null    object
61 academia_y                  986 non-null    object
62 estatus_plantilla_y          986 non-null    object
63 situación_en_plantilla_y      986 non-null    object
64 contrato_colectivo_de_trabajo_y 986 non-null    object
65 régimen_de_contratación_y     986 non-null    object
66 fecha_ingreso_y             986 non-null    object
67 fecha_inicio_ - último_movimiento_y 986 non-null    object
68 fecha_final_ - último_movimiento_y 986 non-null    object
69 tabulador_bruto_y           986 non-null    int64
70 reconocimiento_mensual_bruto_y 986 non-null    object
71 asignación_adicional_bruto_y   986 non-null    object
72 total_mensual_bruto_y        986 non-null    int64
73 puesto_y.1                  986 non-null    object
74 descuento                   986 non-null    float64
dtypes: float64(4), int64(15), object(56)
memory usage: 577.0+ KB

```

```
In [111]: fd_desc_012022 = fd_desc_012022.assign(descuento_sindical = lambda x: (x['tot
```

```
In [112]: fd_desc_012022['descuento_sindical'].sum()
```

```
Out[112]: 170145.550000000005
```

```
In [118]: len(fd_desc_012022['nombre_completo'].unique())
```

```
Out[118]: 986
```

```
In [138]: descuentos_nivel = fd_desc_012022.groupby('nivel')['total_mensual_bruto_y'].s
descuentos_nivel
```

```
Out[138]: nivel
16.0      1263420
18.0      2162944
20.0      2695009
20.5      2658132
29.5      22405233
85.6      1971060
85.7       873312
```


In [140...

```
#fd_plan_2021.groupby('nivel')['total_mensual_bruto'].sum()  
plantilla_nivel = fd_plan_2021.groupby('nivel')['total_mensual_bruto'].sum()  
plantilla_nivel
```

Out[140...

```
nivel  
16.0      1298515  
18.0      2528512  
20.0      3299497  
20.5      3563028  
29.5     52758852  
85.6      2842875  
85.7      1270272  
Name: total_mensual_bruto, dtype: int64
```

In [141...

```
type(plantilla_nivel)
```

Out[141...

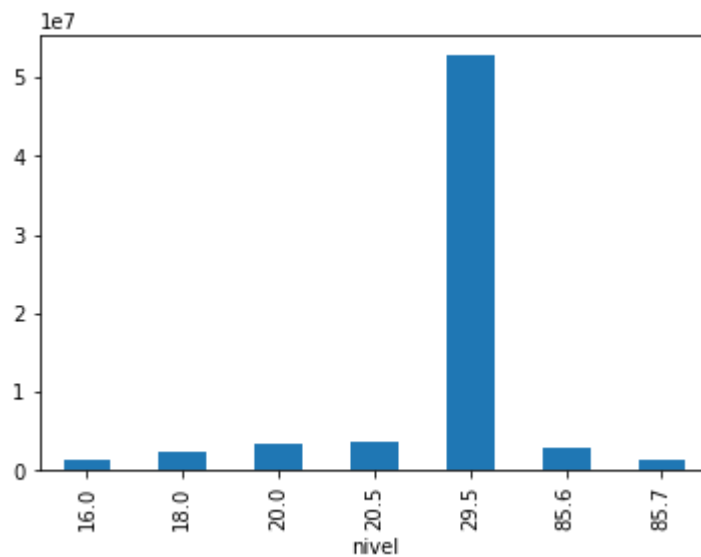
```
pandas.core.series.Series
```

In [145...

```
plantilla_nivel.plot.bar()
```

Out[145...

```
<AxesSubplot:xlabel='nivel'>
```



In [147...

```
tabla_comparativa = pd.concat([descuentos_nivel, plantilla_nivel], axis=1)  
tabla_comparativa
```

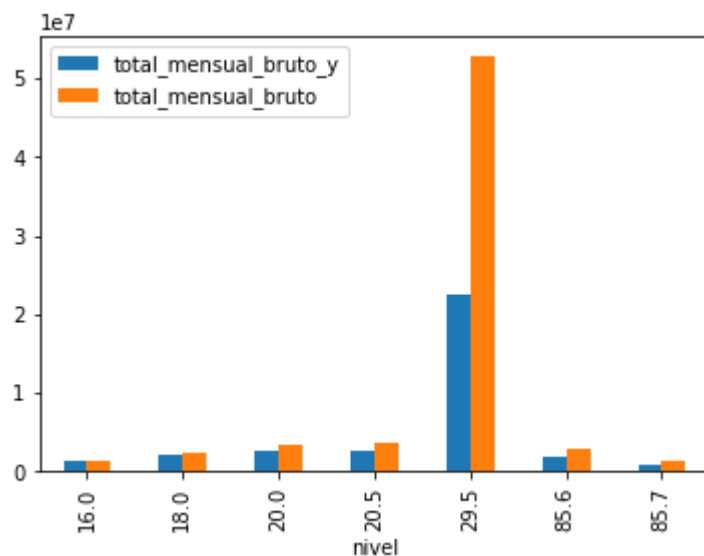
Out[147...

	total_mensual_bruto_y	total_mensual_bruto
nivel		
16.0	1263420	1298515
18.0	2162944	2528512
20.0	2695009	3299497
20.5	2658132	3563028

	total_mensual_bruto_y	total_mensual_bruto
nivel		
29.5	22405233	52758852

In [148... `tabla_comparativa.plot.bar()`

Out[148... `<AxesSubplot:xlabel='nivel'>`



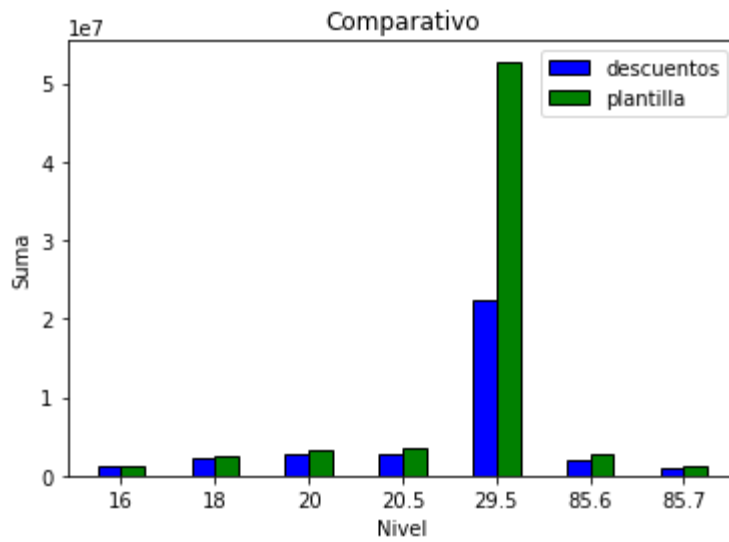
```
In [151...
fig, ax = plt.subplots()
n=7
r = np.arange(n)
width = 0.25

plt.bar(r, descuentos_nivel, color = 'b',
        width = width, edgecolor = 'black',
        label='descuentos')
plt.bar(r + width, plantilla_nivel, color = 'g',
        width = width, edgecolor = 'black',
        label='plantilla')

plt.xlabel("Nivel")
plt.ylabel("Suma")
plt.title("Comparativo")

# plt.grid(linestyle='--')
plt.xticks(r + width/2,['16','18','20','20.5', '29.5', '85.6', '85.7'])
plt.legend()

plt.show()
```



de acuerdo con el analisis hemos perdido mas de la mitad de agremiados del nivel 29.5, que son el sueldo mas grande del conjunto y es el mayor numero de elementos

In []: