BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

INGENIERÍA EN CIENCIA DE DATOS

NOMBRE:

MARTHA PATRICIA ALVAREZ CARRILLO

MATERIA:

INTRODUCCIÓN A LA CIENCIA DE DATOS

PROFESOR:

JAIME ALEJANDRO ROMERO SIERRA

PRÁCTICA # 2

"LIMPIEZA DE BASE DE DATOS ENSUCIADA"

FECHA DE ENTREGA:

21/OCTUBRE/2024

Práctica # 2:

Unidad 2: Introducción a la Limpieza de Datos

Título de la Práctica:

Limpieza de Base de Datos Ensuciada

Objetivo:

Desarrollar habilidades en el preprocesamiento de datos, incluyendo la identificación y tratamiento de valores faltantes, datos duplicados y formatos inconsistentes en una base de datos. Este proceso es fundamental para asegurar la calidad y la integridad de los análisis posteriores.

1. Recepción de la Base de Datos

Descargar el archivo de la base de datos "ensuciada" proporcionado por el profesor. Este archivo contiene errores comunes, como valores duplicados, valores faltantes (NaNs) y errores en el formato de algunas columnas, que afectan el análisis y la calidad de los datos.

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

```
#Carga el archivo CSV en un DataFrame llamado df.
import pandas as pd

df=pd.read_csv('/content/drive/MyDrive/Colab Notebooks/practicas/CD/Base_fs_sucio.csv')
df
#print(df)

```

	Age	Gender	Education Level	Job Title	Years of Experience	Salary
0	32.0	Male	Bachelor's	Software Engineer	5.0	90000.0
1	28.0	Female	Master's	Data Analyst	3.0	65000.0
2	45.0	Male	PhD	Senior Manager	15.0	150000.0
3	36.0	Female	NaN	NaN	7.0	60000.0
4	52.0	Male	Master's	Director	20.0	200000.0
...	...	...	...	...	...	...
7416	28.0	Male	Bachelor's	Software Engineer	3.0	NaN
7417	30.0	Female	Bachelor's Degree	Marketing Coordinator	5.0	95000.0
7418	35.0	Female	PhD	Director of Marketing	12.0	170000.0
7419	26.0	Male	Master's Degree	Digital Marketing Manager	3.0	50000.0
7420	30.0	Male	Bachelor's Degree	Software Engineer	4.0	65000.0

7421 rows × 6 columns

2. Análisis Inicial de la Base de Datos

Realizar un análisis preliminar para comprender la naturaleza y distribución de los errores.

Realizar un análisis preliminar para comprender la naturaleza y distribución de los errores.

```
df.shape
# 7,421
```

```
(7421, 6)
```

```
df.columns
```

```
Index(['Age', 'Gender', 'Education Level', 'Job Title', 'Years of Experience',
       'Salary'],
      dtype='object')
```

a) Visualizar valores únicos de las columnas

```
# Visualizar valores únicos de la columna [1]
df['Age'].unique()
```

```
array([32., 28., 45., 36., 52., 29., 42., 31., 26., 38., 48., 35., 40.,
       27., nan, 39., 25., 51., 34., 47., 30., 41., 37., 24., 43., 33.,
       50., 46., 49., 23., 53., 44., 61., 57., 62., 55., 56., 54., 60.,
       58., 22., 21.])
```

```
# Visualizar valores únicos de la columna [2]
df['Gender'].unique()

```

```
array(['Male', 'Female', nan, 'Other'], dtype=object)
```

```
# Visualizar valores únicos de la columna [3]
df['Education Level'].unique()

```

```
array(["Bachelor's", "Master's", 'PhD', nan, "Bachelor's Degree",
       "Master's Degree", 'High School', 'phD'], dtype=object)
```

```python
# Visualizar valores únicos de la columna [4]
df['Job Title'].unique()

```

```
array(['Software Engineer', 'Data Analyst', 'Senior Manager', nan,
       'Director', 'Marketing Analyst', 'Product Manager',
       'Sales Manager', 'Marketing Coordinator', 'Senior Scientist',
       'Software Developer', 'HR Manager', 'Financial Analyst',
       'Project Manager', 'Customer Service Rep', 'Operations Manager',
       'Marketing Manager', 'Senior Engineer', 'Data Entry Clerk',
       'Sales Director', 'Business Analyst', 'VP of Operations',
       'IT Support', 'Recruiter', 'Financial Manager',
       'Social Media Specialist', 'Junior Developer', 'Product Designer',
       'CEO', 'Accountant', 'Data Scientist', 'Marketing Specialist',
       'Technical Writer', 'HR Generalist', 'Project Engineer',
       'Customer Success Rep', 'Sales Executive', 'UX Designer',
       'Operations Director', 'Network Engineer',
       'Administrative Assistant', 'Strategy Consultant', 'Copywriter',
       'Account Manager', 'Director of Marketing', 'Help Desk Analyst',
       'Customer Service Manager', 'Business Intelligence Analyst',
       'VP of Finance', 'Graphic Designer', 'UX Researcher',
       'Social Media Manager', 'Director of Operations',
       'Senior Data Scientist', 'Junior Accountant',
       'Digital Marketing Manager', 'IT Manager',
       'Customer Service Representative', 'Business Development Manager',
       'Senior Financial Analyst', 'Web Developer', 'Research Director',
       'Technical Support Specialist', 'Creative Director',
       'Senior Software Engineer', 'Human Resources Director',
       'Content Marketing Manager', 'Technical Recruiter', 'bbb',
       'Chief Technology Officer', 'Junior Designer', 'Financial Advisor',
       'Junior Account Manager', 'Senior Project Manager',
       'Principal Scientist', 'Sales Associate', 'Supply Chain Manager',
       'Senior Marketing Manager', 'Training Specialist',
       'Research Scientist', 'Junior Software Developer',
       'Public Relations Manager', 'Operations Analyst',
       'Event Coordinator', 'Product Marketing Manager',
       'Senior HR Manager', 'Junior Web Developer',
       'Senior Project Coordinator', 'Digital Content Producer',
       'IT Support Specialist', 'Senior Marketing Analyst',
       'Customer Success Manager', 'Senior Graphic Designer',
       'Supply Chain Analyst', 'Senior Business Analyst',
       'Office Manager', 'Junior HR Generalist', 'Senior Product Manager',
       'Junior Operations Analyst', 'Senior HR Generalist',
       'Sales Operations Manager', 'Senior Software Developer',
       'Junior Web Designer', 'Senior Training Specialist',
       'Senior Research Scientist', 'Junior Sales Representative',
       'Junior Marketing Manager', 'Junior Data Analyst',
       'Senior Product Marketing Manager', 'Junior Business Analyst',
       'Junior Marketing Specialist', 'Junior Project Manager',
```

```
'Senior Accountant', 'Director of Sales', 'Junior Recruiter',
'Senior Business Development Manager', 'Senior Product Designer',
'Junior Customer Support Specialist',
'Senior IT Support Specialist', 'Junior Financial Analyst',
'Senior Operations Manager', 'Director of Human Resources',
'Junior Software Engineer', 'Senior Sales Representative',
'Director of Product Management', 'Junior Copywriter',
'Senior Marketing Coordinator', 'Senior Human Resources Manager',
'Junior Business Development Associate', 'Senior Account Manager',
'Senior Researcher', 'Junior HR Coordinator',
'Director of Finance', 'Junior Marketing Coordinator',
'Junior Data Scientist', 'Senior Operations Analyst',
'Senior Human Resources Coordinator', 'Senior UX Designer',
'Junior Product Manager', 'Senior Marketing Specialist',
'Senior IT Project Manager', 'Senior Quality Assurance Analyst',
'Director of Sales and Marketing', 'Senior Account Executive',
'Director of Business Development', 'Junior Social Media Manager',
'Senior Human Resources Specialist', 'Senior Data Analyst',
'Director of Human Capital', 'Junior Advertising Coordinator',
'Senior Sales Manager', 'Junior UX Designer',
'Senior Marketing Director', 'Senior IT Consultant',
'Senior Financial Advisor', 'Junior Business Operations Analyst',
'Junior Social Media Specialist',
'Senior Product Development Manager', 'Junior Operations Manager',
'Senior Software Architect', 'Junior Research Scientist',
'Junior Marketing Analyst', 'Senior Financial Manager',
'Senior HR Specialist', 'Senior Data Engineer',
'Junior Operations Coordinator', 'Director of HR',
'Senior Operations Coordinator', 'Junior Financial Advisor',
'Director of Engineering', 'Software Engineer Manager',
'Back end Developer', 'Senior Project Engineer',
'Full Stack Engineer', 'Front end Developer', 'Developer',
'Front End Developer', 'Director of Data Science',
'Human Resources Coordinator', 'Junior Sales Associate',
'Human Resources Manager', 'Juniour HR Generalist',
'Juniour HR Coordinator', 'Sales Representative',
'Digital Marketing Specialist', 'Receptionist',
'Marketing Director', 'Social M', 'Social Media Man',
'Delivery Driver'], dtype=object)
```

```python
# Visualizar valores únicos de la columna [5]
df['Years of Experience'].unique()

# --->>> Se observa que se debe de cambiar 'bbb' pues no sería un dato para Años de Experiencia

```

```
array(['5.0', '3.0', '15.0', '7.0', '20.0', '2.0', '12.0', '4.0', '1.0',
       '10.0', nan, '18.0', '6.0', '14.0', '16.0', '0.0', '19.0', '9.0',
       '13.0', '11.0', '25.0', '21.0', '8.0', '22.0', 'bbb', '23.0',
       '24.0', '17.0', '1.5', '31.0', '30.0', '28.0', '33.0', '27.0',
       '34.0', '29.0', '26.0', '32.0'], dtype=object)
```

```python
# Visualizar valores únicos de la columna [6]
df['Salary'].unique()

```

```
array([ 90000.,  65000., 150000.,  60000., 200000.,     nan, 120000.,
        80000.,  45000., 110000.,  75000., 130000.,  40000., 125000.,
       115000.,  35000., 180000., 190000.,  50000., 140000., 250000.,
        55000.,  95000., 105000., 170000.,  70000., 160000., 100000.,
        30000., 220000., 135000., 175000., 185000.,  85000., 145000.,
       155000.,    350., 195000., 198000., 196000., 193000.,  92000.,
       165000., 162000., 197000., 142000., 182000., 210000.,    550.,
       122485., 169159., 187081., 166109.,  78354.,  90249., 132720.,
       161568., 127346., 120177.,  69032., 101332., 121450., 166375.,
       185119., 166512., 186963.,  75072., 163398., 103947., 179180.,
       175966., 190004., 152039.,  76742., 191790., 139398.,  95845.,
       160976., 126753., 161393., 139817., 181714., 114776., 105725.,
        52731., 106492.,  73895., 119836.,  99747., 168287., 115920.,
       128078.,  51265., 165919., 188651.,  55538., 193964., 104702.,
       172955., 138032.,  82683., 155414., 154207., 148446., 102859.,
       138662., 181699., 188232.,  51832., 188484., 138286., 181132.,
        73938., 119224., 142360., 151315., 181021., 134641., 173851.,
       104127., 178859.,  98568., 134858.,  94502., 149217., 107895.,
       101186.,  62852., 139095., 106278.,  90452., 168304., 126593.,
       152203., 183138., 130275., 191915.,  62807., 174305., 133326.,
        75656., 155944., 137775.,  51831., 182237., 151901., 158254.,
       167207., 112439., 194214.,  84407., 139413., 143084., 192344.,
       106132., 184816., 150248., 170995.,  88035., 119419., 173582.,
       174436.,  71699., 163558., 166828., 144496., 193746., 122581.,
        79767., 177177.,  89843., 113563., 128712., 161621., 121454.,
       179987.,  72649.,  52612., 184006., 131960., 102465., 149748.,
       171036., 146351., 185462., 107718.,  90944.,  63901., 181902.,
       136533., 136285., 191818., 176643.,  70022.,  99363., 152944.,
       123386., 168906., 183020.,  47898., 135853., 149198., 106662.,
```

```
 89995., 143814., 174726.,  68732., 187951., 137336., 191159.,
102868., 154281., 111535., 107906., 180958., 108607., 178284.,
 75969., 197354., 174324., 123781., 141735., 187120.,  61095.,
179045., 130355., 103282., 157872., 117314., 186321., 129686.,
 68611., 177913.,  68472., 113065., 125091., 172925., 126916.,
 76898.,    579., 103579., 163780., 137878.,  92438.,  84181.,
174821., 126520., 152168., 190543., 192292.,  52807., 174938.,
124071.,  73640., 156486., 138859.,  52831., 182392., 151078.,
158966., 167924., 113334., 194778.,  77606., 140010., 142421.,
192756., 106686., 150729., 171652.,  88552., 119918., 174985.,
174336.,  72389., 163978., 166958., 145052., 195270., 122970.,
 80247., 177862., 114290., 128999., 162454., 122354., 179756.,
 73218., 184480., 102828., 150301., 171468., 147326., 185982.,
108267.,  91397., 100867.,  64182., 182506., 136986., 136662.,
191510., 177347.,  70397., 155795., 132638., 178684., 106218.,
191239.,  65840.,  52779., 185038., 136449., 110707., 151670.,
167015., 146508., 190596., 104378.,  70216., 101733.,  55935.,
180367., 135596., 136062., 191267., 146075., 131547., 100679.,
186794.,  91062., 132442.,  82944., 188288., 141090., 152726.,
124141.,  67556., 182768., 148727.,  91903., 147708., 163209.,
120288., 170226., 134979., 137489.,  83577., 117904., 134482.,
184660., 100151.,  88678., 181285., 154990., 108204., 175684.,
 77766., 192211., 144647., 162231., 121120.,  79652., 177002.,
182013., 108799., 135378., 183530., 150901.,  82697., 194638.,
130356., 152560., 121432.,  63789., 183690., 151310., 100358.,
148437., 168691.,  32000.,  38000.,  89000.,  33000.,  25000.,
 62000., 138000.,  47000.,  26000., 174000.,  41000.,  99000.,
117000., 225000.,  36000., 146000., 113000., 168000., 122000.,
 96000.,  49000.,  68000., 127000.,  71000., 240000., 152000.,
119000., 131000., 101000., 137000., 112000.,  91000., 179000.,
 74000., 228000.,  37000., 204000.,  61000., 157000.,  52000.,
 58000., 219000.,  77000., 104000., 183000.,  43000.,  48000.,
 42000.,    500.,  57000.,  72000.,  31000.,  28000., 215000.,
100052.])
```

b) Contar Datos de cada valor único de las columnas

```
1  # Contar cuantos valores hay de cada valor único de la columna [1]
2  df['Age'].value_counts()
3
```

	Age	count
27		541
29		488
30		475
28		462
33		434
26		412

31	388
32	372
34	323
36	308
25	290
24	256
35	216
42	184
43	168
37	166
39	162
38	157
45	153
41	141
44	126
46	111
50	104
23	103
48	99
49	98
40	94
54	75
47	49
51	31
52	31
55	18
21	18
22	17
56	12
57	10
53	9
58	8
60	6
62	5
61	2

dtype: int64

```python
# Contar cuantos valores hay de cada valor único de la columna [2]
df['Gender'].value_counts()

```

Gender	count
Male	3897
Female	3211
Other	14

dtype: int64

```python
# Contar cuantos valores hay de cada valor único de la columna [3]
df['Education Level'].value_counts()

```

Education Level	count
Bachelor's Degree	2427
Master's Degree	1683
PhD	1446
Bachelor's	789
High School	470
Master's	304
phD	1

dtype: int64

```python
# Contar cuantos valores hay de cada valor único de la columna [4]
df['Job Title'].value_counts()

```

Job Title	count
Software Engineer	537
Data Scientist	472
Software Engineer Manager	403
Data Analyst	378
Full Stack Engineer	327
...	...
Senior IT Support Specialist	1
Junior Customer Support Specialist	1
Business Intelligence Analyst	1
Junior Recruiter	1
Office Manager	1

189 rows × 1 columns

dtype: int64

```python
# Contar cuantos valores hay de cada valor único de la columna [5]
df['Years of Experience'].value_counts() # Validar el Tipo de Datos

```

Years of Experience	count
3	618
2	612
1	558
4	545
6	471
8	441
5	425
9	399
7	383
11	315
12	303
14	274
16	248
13	220
10	199
bbb	140
15	133

18	133
19	126
0	123
17	111
20	65
22	47
21	45
23	43
25	30
24	21
28	17
32	13
27	12
1.5	12
29	11
30	8
33	6
26	6
31	5
34	2

dtype: int64

```
# Contar cuantos valores hay de cada valor único de la columna [6]
df['Salary'].value_counts()
```

Salary	count
140000.0	309
120000.0	295
160000.0	283
55000.0	251
60000.0	240
...	...
106662.0	1
89995.0	1
143814.0	1
174726.0	1
100052.0	1

434 rows × 1 columns

dtype: int64

c) Mostrar un resumen estadístico de los datos

Usar df.describe() para obtener estadísticas descriptivas de las columnas numéricas, lo que ayuda a identificar valores atípicos y la distribución general.

```
# 2. c) Mostrar un resumen estadístico de los datos:

#print(df_Trad.describe())
df.describe() #sólo se mestran los datos numéricos

```

	Age	Salary
count	7122.000000	6971.000000
mean	33.666807	115628.243724
std	7.633494	52565.497372
min	21.000000	350.000000
25%	28.000000	70000.000000
50%	32.000000	115000.000000
75%	38.000000	160000.000000
max	62.000000	250000.000000

Se observa que no se muestra información de todas las columnas, por lo que se tendrá que cambiar los datos a números, para mostrar resumen estadístico completo, por lo que primero se procede con la Traducción de los datos (no incluir acentos en los datos de las columnas)

Traducción y Cambio a Valor Numérico

```
# Se genera df para la traducción de los datos
df_Trad= df
df_Trad.head(5)

```

	Age	Gender	Education Level	Job Title	Years of Experience	Salary
0	32.0	Male	Bachelor's	Software Engineer	5.0	90000.0
1	28.0	Female	Master's	Data Analyst	3.0	65000.0
2	45.0	Male	PhD	Senior Manager	15.0	150000.0
3	36.0	Female	NaN	NaN	7.0	60000.0
4	52.0	Male	Master's	Director	20.0	200000.0

Renombrar columnas

```
# Renombrar la columna [1]
df_Trad = df_Trad.rename(columns={'Age': 'Edad'})
df_Trad.head(5)
```

	Edad	Gender	Education Level	Job Title	Years of Experience	Salary
0	32.0	Male	Bachelor's	Software Engineer	5.0	90000.0
1	28.0	Female	Master's	Data Analyst	3.0	65000.0
2	45.0	Male	PhD	Senior Manager	15.0	150000.0
3	36.0	Female	NaN	NaN	7.0	60000.0
4	52.0	Male	Master's	Director	20.0	200000.0

```
# Renombrar la columna [2]
df_Trad = df_Trad.rename(columns={'Gender': 'Genero'})
df_Trad.head(5)
```

	Edad	Genero	Education Level	Job Title	Years of Experience	Salary
0	32.0	Male	Bachelor's	Software Engineer	5.0	90000.0
1	28.0	Female	Master's	Data Analyst	3.0	65000.0
2	45.0	Male	PhD	Senior Manager	15.0	150000.0
3	36.0	Female	NaN	NaN	7.0	60000.0
4	52.0	Male	Master's	Director	20.0	200000.0

```python
# Renombrar la columna [3]
df_Trad = df_Trad.rename(columns={'Education Level': 'Nivel_Educativo'})
df_Trad.head(5)

```

	Edad	Genero	Nivel_Educativo	Job Title	Years of Experience	Salary
0	32.0	Male	Bachelor's	Software Engineer	5.0	90000.0
1	28.0	Female	Master's	Data Analyst	3.0	65000.0
2	45.0	Male	PhD	Senior Manager	15.0	150000.0
3	36.0	Female	NaN	NaN	7.0	60000.0
4	52.0	Male	Master's	Director	20.0	200000.0

```python
# Renombrar la columna [4]
df_Trad = df_Trad.rename(columns={'Job Title': 'Titulo_Del_Trabajo'})
df_Trad.head(5)

```

	Edad	Genero	Nivel_Educativo	Titulo_Del_Trabajo	Years of Experience	Salary
0	32.0	Male	Bachelor's	Software Engineer	5.0	90000.0
1	28.0	Female	Master's	Data Analyst	3.0	65000.0
2	45.0	Male	PhD	Senior Manager	15.0	150000.0
3	36.0	Female	NaN	NaN	7.0	60000.0
4	52.0	Male	Master's	Director	20.0	200000.0

```python
# Renombrar la columna [5]
df_Trad = df_Trad.rename(columns={'Years of Experience': 'Años_De_Experiencia'})
df_Trad.head(5)

```

	Edad	Genero	Nivel_Educativo	Titulo_Del_Trabajo	Años_De_Experiencia	Salary
0	32.0	Male	Bachelor's	Software Engineer	5.0	90000.0
1	28.0	Female	Master's	Data Analyst	3.0	65000.0
2	45.0	Male	PhD	Senior Manager	15.0	150000.0
3	36.0	Female	NaN	NaN	7.0	60000.0
4	52.0	Male	Master's	Director	20.0	200000.0

```python
# Renombrar la columna [6]
df_Trad = df_Trad.rename(columns={'Salary': 'Salario'})
df_Trad.head(5)

```

	Edad	Genero	Nivel_Educativo	Titulo_Del_Trabajo	Años_De_Experiencia	Salario
0	32.0	Male	Bachelor's	Software Engineer	5.0	90000.0
1	28.0	Female	Master's	Data Analyst	3.0	65000.0
2	45.0	Male	PhD	Senior Manager	15.0	150000.0
3	36.0	Female	NaN	NaN	7.0	60000.0
4	52.0	Male	Master's	Director	20.0	200000.0

Traducción de Datos

```python
# Diccionario para traducir el contenido de la columna [2] Genero
traduccionGenero = {
    'Female': 'Mujer',
    'Male': 'Hombre',
    'Other': 'Otro'
}

```

```python
# Reemplazar (traducir) el contenido de la columna [2] 'Genero'
df_Trad['Genero'] = df_Trad['Genero'].replace(traduccionGenero)
df_Trad.head(5)

```

	Edad	Genero	Nivel_Educativo	Titulo_Del_Trabajo	Años_De_Experiencia	Salario
0	32.0	Hombre	Bachelor's	Software Engineer	5.0	90000.0
1	28.0	Mujer	Master's	Data Analyst	3.0	65000.0
2	45.0	Hombre	PhD	Senior Manager	15.0	150000.0
3	36.0	Mujer	NaN	NaN	7.0	60000.0
4	52.0	Hombre	Master's	Director	20.0	200000.0

```python
# Diccionario para traducir el contenido de la columna [3] Nivel_Educativo
traduccionNivelEducativo = {
    "Bachelor's": 'Licenciatura',
    "Master's": 'Maestria',
    "PhD": 'Doctorado',
    "Master's Degree": 'Titulo de Maestria',
    "Bachelor's Degree": 'Titulo de Licenciatura',
    "High School": 'Escuela Secundaria',
    "phD": 'Doctorado'
}

```

```python
# Reemplazar (traducir) el contenido de la columna [3] 'Nivel_Educativo'
df_Trad['Nivel_Educativo'] = df_Trad['Nivel_Educativo'].replace(traduccionNivelEducativo)
df_Trad.head(5)

```

	Edad	Genero	Nivel_Educativo	Titulo_Del_Trabajo	Años_De_Experiencia	Salario
0	32.0	Hombre	Licenciatura	Software Engineer	5.0	90000.0
1	28.0	Mujer	Maestria	Data Analyst	3.0	65000.0
2	45.0	Hombre	Doctorado	Senior Manager	15.0	150000.0
3	36.0	Mujer	NaN	NaN	7.0	60000.0
4	52.0	Hombre	Maestria	Director	20.0	200000.0

```python
# Diccionario para traducir el contenido de la columna [4] Titulo_Del_Trabajo
traduccionTituloTrabajo = {
```

```
'Software Engineer': 'Ingeniero de Software',
'Data Analyst': 'Analista de Datos',
'Senior Manager': 'Gerente Senior',
'Director': 'Director',
'Product Manager': 'Gerente de Producto',
'Sales Manager': 'Gerente de Ventas',
'Marketing Coordinator': 'Coordinador de Marketing',
'Financial Analyst': 'Analista Financiero',
'Project Manager': 'Gerente de Proyectos',
'Customer Service Rep': 'Representante de Servicio al Cliente',
'Data Entry Clerk': 'Empleado de Ingreso de Datos',
'Business Analyst': 'Analista de Negocios',
'VP of Operations': 'VP de Operaciones',
'IT Support': 'Soporte Tecnico',
'Financial Manager': 'Gerente Financiero',
'Social Media Specialist': 'Especialista en Redes Sociales',
'Junior Developer': 'Desarrollador Junior',
'Product Designer': 'Diseñador de Producto',
'CEO': 'CEO',
'Data Scientist': 'Cientifico de Datos',
'Marketing Specialist': 'Especialista en Marketing',
```

```
'Technical Writer': 'Redactor Tecnico',
'HR Generalist': 'Generalista de Recursos Humanos',
'Project Engineer': 'Ingeniero de Proyectos',
'Customer Success Rep': 'Representante de exito del Cliente',
'UX Designer': 'Diseñador UX',
'Operations Director': 'Director de Operaciones',
'Administrative Assistant': 'Asistente Administrativo',
'Strategy Consultant': 'Consultor de Estrategia',
'Copywriter': 'Redactor Publicitario',
'Director of Marketing': 'Director de Marketing',
'Help Desk Analyst': 'Analista de Mesa de Ayuda',
'VP of Finance': 'VP de Finanzas',
'Graphic Designer': 'Diseñador Grafico',
'Senior Engineer': 'Ingeniero Senior',
'Social Media Manager': 'Gerente de Redes Sociales',
'Director of Operations': 'Director de Operaciones',
'Marketing Analyst': 'Analista de Marketing',
'HR Manager': 'Gerente de Recursos Humanos',
'Senior Data Scientist': 'Cientifico de Datos Senior',
'Junior Accountant': 'Contador Junior',
'Digital Marketing Manager': 'Gerente de Marketing Digital',
'Business Development Manager': 'Gerente de Desarrollo de Negocios',
'Web Developer': 'Desarrollador Web',
'Recruiter': 'Reclutador',
'Research Director': 'Director de Investigacion',
'Technical Support Specialist': 'Especialista en Soporte Tecnico',
'Creative Director': 'Director Creativo',
'Operations Manager': 'Gerente de Operaciones',
'Senior Software Engineer': 'Ingeniero de Software Senior',
'Technical Recruiter': 'Reclutador Tecnico',
'bbb': 'bbb',
'Chief Technology Officer': 'Director de Tecnologia',
'Financial Advisor': 'Asesor Financiero',
'Junior Account Manager': 'Gerente de Cuentas Junior',
'Principal Scientist': 'Cientifico Principal',
'Supply Chain Manager': 'Gerente de Cadena de Suministro',
'Senior Marketing Manager': 'Gerente de Marketing Senior',
'Training Specialist': 'Especialista en Capacitacion',
'Junior Software Developer': 'Desarrollador de Software Junior',
'Operations Analyst': 'Analista de Operaciones',
'Event Coordinator': 'Coordinador de Eventos',
'Product Marketing Manager': 'Gerente de Marketing de Producto',
'Senior HR Manager': 'Gerente de Recursos Humanos Senior',
'Junior Web Developer': 'Desarrollador Web Junior',
'Senior Project Coordinator': 'Coordinador de Proyectos Senior',
'Digital Content Producer': 'Productor de Contenido Digital',
'Customer Success Manager': 'Gerente de exito del Cliente',
'Supply Chain Analyst': 'Analista de Cadena de Suministro',
```

```
'Senior Business Analyst': 'Analista de Negocios Senior',
'Senior Financial Analyst': 'Analista Financiero Senior',
'Office Manager': 'Gerente de Oficina',
'Senior Product Manager': 'Gerente de Producto Senior',
'Junior Operations Analyst': 'Analista de Operaciones Junior',
'Customer Service Manager': 'Gerente de Servicio al Cliente',
'Senior Scientist': 'Cientifico Senior',
'Senior HR Generalist': 'Generalista de Recursos Humanos Senior',
'Junior Web Designer': 'Diseñador Web Junior',
'Senior Training Specialist': 'Especialista en Capacitacion Senior',
'Senior Research Scientist': 'Cientifico Investigador Senior',
'Junior Sales Representative': 'Representante de Ventas Junior',
'Senior Project Manager': 'Gerente de Proyectos Senior',
'Junior Data Analyst': 'Analista de Datos Junior',
'Junior Business Analyst': 'Analista de Negocios Junior',
'Junior Project Manager': 'Gerente de Proyectos Junior',
'Senior Accountant': 'Contador Senior',
'Director of Sales': 'Director de Ventas',
'Senior Business Development Manager': 'Gerente de Desarrollo de Negocios Senior',
'Senior Product Designer': 'Diseñador de Producto Senior',
'Junior Customer Support Specialist': 'Especialista en Soporte al Cliente Junior',
'Senior Marketing Analyst': 'Analista de Marketing Senior',
'Senior IT Support Specialist': 'Especialista en Soporte Tecnico Senior',
'Junior Financial Analyst': 'Analista Financiero Junior',
'Senior Operations Manager': 'Gerente de Operaciones Senior',
'Director of Human Resources': 'Director de Recursos Humanos',
'Junior Software Engineer': 'Ingeniero de Software Junior',
'Senior Sales Representative': 'Representante de Ventas Senior',
'Director of Product Management': 'Director de Gestion de Producto',
'Junior Copywriter': 'Redactor Junior',
'Senior Marketing Coordinator': 'Coordinador de Marketing Senior',
'Senior Human Resources Manager': 'Gerente Senior de Recursos Humanos',
'Junior Business Development Associate': 'Asociado de Desarrollo de Negocios Junior',
'Senior Account Manager': 'Gerente de Cuentas Senior',
'Senior Researcher': 'Investigador Senior',
'Junior HR Coordinator': 'Coordinador de Recursos Humanos Junior',
'Director of Finance': 'Director de Finanzas',
'Junior Data Scientist': 'Cientifico de Datos Junior',
'Senior Operations Analyst': 'Analista de Operaciones Senior',
'Senior Human Resources Coordinator': 'Coordinador de Recursos Humanos Senior',
'Senior UX Designer': 'Diseñador UX Senior',
'Junior Product Manager': 'Gerente de Producto Junior',
'Senior Marketing Specialist': 'Especialista en Marketing Senior',
'Senior IT Project Manager': 'Gerente de Proyectos de TI Senior',
'Senior Quality Assurance Analyst': 'Analista de Aseguramiento de Calidad Senior',
'Senior Account Executive': 'Ejecutivo de Cuentas Senior',
'Director of Business Development': 'Director de Desarrollo de Negocios',
'Junior Social Media Manager': 'Gerente de Redes Sociales Junior',
```

'Senior Human Resources Specialist': 'Especialista en Recursos Humanos Senior',
'Senior Data Analyst': 'Analista de Datos Senior',
'Director of Human Capital': 'Director de Capital Humano',
'Junior Advertising Coordinator': 'Coordinador de Publicidad Junior',
'Junior UX Designer': 'Diseñador UX Junior',
'Senior Marketing Director': 'Director de Marketing Senior',
'Junior HR Generalist': 'Generalista de Recursos Humanos Junior',
'Junior Marketing Coordinator': 'Coordinador de Marketing Junior',
'Senior Financial Advisor': 'Asesor Financiero Senior',
'Junior Business Operations Analyst': 'Analista de Operaciones de Negocios Junior',
'Junior Social Media Specialist': 'Especialista en Redes Sociales Junior',
'Junior Operations Manager': 'Gerente de Operaciones Junior',
'Senior Software Architect': 'Arquitecto de Software Senior',
'Junior Marketing Specialist': 'Especialista en Marketing Junior',
'Senior Software Developer': 'Desarrollador de Software Senior',
'Junior Marketing Analyst': 'Analista de Marketing Junior',
'Senior IT Consultant': 'Consultor de TI Senior',
'Senior Financial Manager': 'Gerente Financiero Senior',
'Junior Marketing Manager': 'Gerente de Marketing Junior',
'Junior Operations Coordinator': 'Coordinador de Operaciones Junior',
'Director of HR': 'Director de Recursos Humanos',
'Senior Operations Coordinator': 'Coordinador de Operaciones Senior',
'Senior Data Engineer': 'Ingeniero de Datos Senior',
'Junior Financial Advisor': 'Asesor Financiero Junior',
'Director of Engineering': 'Director de Ingenieria',
'Senior Project Engineer': 'Ingeniero de Proyectos Senior',
'Full Stack Engineer': 'Ingeniero Full Stack',
'Front end Developer': 'Desarrollador Front End',
'Back end Developer': 'Desarrollador Back End',
'Software Engineer Manager': 'Gerente de Ingenieros de Software',
'Front End Developer': 'Desarrollador Front End',
'Software Developer': 'Desarrollador de Software',
'Director of Data Science': 'Director de Ciencia de Datos',
'Marketing Manager': 'Gerente de Marketing',
'Human Resources Coordinator': 'Coordinador de Recursos Humanos',
'Junior Sales Associate': 'Asociado de Ventas Junior',
'Human Resources Manager': 'Gerente de Recursos Humanos',
'Juniour HR Generalist': 'Generalista de Recursos Humanos Junior',
'Juniour HR Coordinator': 'Coordinador de Recursos Humanos Junior',
'Senior Product Marketing Manager': 'Gerente Senior de Marketing de Producto',
'Sales Associate': 'Asociado de Ventas',
'Content Marketing Manager': 'Gerente de Marketing de Contenidos',
'Sales Director': 'Director de Ventas',
'Sales Representative': 'Representante de Ventas',
'Research Scientist': 'Cientifico Investigador',
'Digital Marketing Specialist': 'Especialista en Marketing Digital',
'Receptionist': 'Recepcionista',
'Marketing Director': 'Director de Marketing',

```
    'Social Media Man': 'Gerente de Redes Sociales',
    'Customer Service Representative': 'Representante de Servicio al Cliente',
    'Delivery Driver': 'Conductor de Entrega',
    'Sales Executive': 'Ejecutivo de Ventas',
    'Junior Research Scientist': 'Cientifico Investigador Junior',
    'Sales Operations Manager': 'Grente de Operaciones de Ventas'
}
```

```
# Reemplazar (traducir) el contenido de la columna [4] 'Titulo_Del_Trabajo'
df_Trad['Titulo_Del_Trabajo'] = df_Trad['Titulo_Del_Trabajo'].replace(traduccionTituloTrabajo)
df_Trad.head(5)
```

	Edad	Genero	Nivel_Educativo	Titulo_Del_Trabajo	Años_De_Experiencia	Salario
0	32.0	Hombre	Licenciatura	Ingeniero de Software	5.0	90000.0
1	28.0	Mujer	Maestria	Analista de Datos	3.0	65000.0
2	45.0	Hombre	Doctorado	Gerente Senior	15.0	150000.0
3	36.0	Mujer	NaN	NaN	7.0	60000.0
4	52.0	Hombre	Maestria	Director	20.0	200000.0

Convertir las columnas a valores numéricos

```
# Se genera df para mostrar resumen estadístico
dfNum = df_Trad
dfNum.head(5)
```

	Edad	Genero	Nivel_Educativo	Titulo_Del_Trabajo	Años_De_Experiencia	Salario
0	32.0	Hombre	Licenciatura	Ingeniero de Software	5.0	90000.0
1	28.0	Mujer	Maestria	Analista de Datos	3.0	65000.0
2	45.0	Hombre	Doctorado	Gerente Senior	15.0	150000.0
3	36.0	Mujer	NaN	NaN	7.0	60000.0
4	52.0	Hombre	Maestria	Director	20.0	200000.0

```python
# Convertir la columna [2] 'Genero' a valores numéricos
dfNum['Genero'] = dfNum['Genero'].map({
    'Hombre': 1,
    'Mujer': 2,
    'Otro': 3
    })
dfNum.head(5)

```

	Edad	Genero	Nivel_Educativo	Titulo_Del_Trabajo	Años_De_Experiencia	Salario
0	32.0	1.0	Licenciatura	Ingeniero de Software	5.0	90000.0
1	28.0	2.0	Maestria	Analista de Datos	3.0	65000.0
2	45.0	1.0	Doctorado	Gerente Senior	15.0	150000.0
3	36.0	2.0	NaN	NaN	7.0	60000.0
4	52.0	1.0	Maestria	Director	20.0	200000.0

```python
# Convertir la columna [3] 'Nivel_Educativo' a valores numéricos
dfNum['Nivel_Educativo'] = dfNum['Nivel_Educativo'].map({
    'Licenciatura': 1,
    'Maestria': 2,
    'Doctorado': 3,
    'Titulo de Maestria': 4,
    'Titulo de Licenciatura': 5,
    'Escuela Secundaria': 6,
    'Doctorado': 7
    })
dfNum.head(5)

```

	Edad	Genero	Nivel_Educativo	Titulo_Del_Trabajo	Años_De_Experiencia	Salario
0	32.0	1.0	1.0	Ingeniero de Software	5.0	90000.0
1	28.0	2.0	2.0	Analista de Datos	3.0	65000.0
2	45.0	1.0	7.0	Gerente Senior	15.0	150000.0
3	36.0	2.0	NaN	NaN	7.0	60000.0
4	52.0	1.0	2.0	Director	20.0	200000.0

```python
# Convertir la columna [4] 'Titulo_Del_Trabajo' a valores numéricos
dfNum['Titulo_Del_Trabajo'] = dfNum['Titulo_Del_Trabajo'].map({
    'Ingeniero de Software': 1,
    'Analista de Datos': 2,
    'Gerente Senior': 3,
    'Director': 4,
    'Gerente de Producto': 5,
    'Gerente de Ventas': 6,
    'Coordinador de Marketing': 7,
    'Analista Financiero': 8,
    'Gerente de Proyectos': 9,
```

```
'Representante de Servicio al Cliente': 10,
'Empleado de Ingreso de Datos': 11,
'Analista de Negocios': 12,
'VP de Operaciones': 13,
'Soporte Tecnico': 14,
'Gerente Financiero': 15,
'Especialista en Redes Sociales': 16,
'Desarrollador Junior': 17,
'Diseñador de Producto': 18,
'CEO': 19,
'Cientifico de Datos': 20,
'Especialista en Marketing': 21,
'Redactor Tecnico': 22,
'Generalista de Recursos Humanos': 23,
'Ingeniero de Proyectos': 24,
'Representante de exito del Cliente': 25,
'Diseñador UX': 26,
'Director de Operaciones': 27,
'Asistente Administrativo': 28,
'Consultor de Estrategia': 29,
'Redactor Publicitario': 30,
'Director de Marketing': 31,
'Analista de Mesa de Ayuda': 32,
'VP de Finanzas': 33,
'Diseñador Grafico': 34,
'Ingeniero Senior': 35,
'Gerente de Redes Sociales': 36,
'Director de Operaciones': 37,
'Analista de Marketing': 38,
'Gerente de Recursos Humanos': 39,
'Cientifico de Datos Senior': 40,
'Contador Junior': 41,
'Gerente de Marketing Digital': 42,
'Gerente de Desarrollo de Negocios': 43,
'Desarrollador Web': 44,
'Reclutador': 45,
'Director de Investigacion': 46,
'Especialista en Soporte Tecnico': 47,
'Director Creativo': 48,
'Gerente de Operaciones': 49,
'Ingeniero de Software Senior': 50,
'Reclutador Tecnico': 51,
'bbb': 52,
'Director de Tecnologia': 53,
'Asesor Financiero': 54,
'Gerente de Cuentas Junior': 55,
'Cientifico Principal': 56,
'Gerente de Cadena de Suministro': 57,
```

```
    'Gerente de Marketing Senior': 58,
    'Especialista en Capacitacion': 59,
    'Desarrollador de Software Junior': 60,
    'Analista de Operaciones': 61,
    'Coordinador de Eventos': 62,
    'Gerente de Marketing de Producto': 63,
    'Gerente de Recursos Humanos Senior': 64,
    'Desarrollador Web Junior': 65,
    'Coordinador de Proyectos Senior': 66,
    'Productor de Contenido Digital': 67,
    'Gerente de exito del Cliente': 68,
    'Analista de Cadena de Suministro': 69,
    'Analista de Negocios Senior': 70,
    'Analista Financiero Senior': 71,
    'Gerente de Oficina': 72,
    'Gerente de Producto Senior': 73,
    'Analista de Operaciones Junior': 74,
    'Gerente de Servicio al Cliente': 75,
    'Cientifico Senior': 76,
    'Generalista de Recursos Humanos Senior': 77,
    'Diseñador Web Junior': 78,
    'Especialista en Capacitacion Senior': 79,
    'Cientifico Investigador Senior': 80,
    'Representante de Ventas Junior': 81,
    'Gerente de Proyectos Senior': 82,
    'Analista de Datos Junior': 83,
    'Analista de Negocios Junior': 84,
    'Gerente de Proyectos Junior': 85,
    'Contador Senior': 86,
    'Director de Ventas': 87,
    'Gerente de Desarrollo de Negocios Senior': 88,
    'Diseñador de Producto Senior': 89,
    'Especialista en Soporte al Cliente Junior': 90,
    'Analista de Marketing Senior': 91,
    'Especialista en Soporte Tecnico Senior': 92,
    'Analista Financiero Junior': 93,
    'Gerente de Operaciones Senior': 94,
    'Director de Recursos Humanos': 95,
    'Ingeniero de Software Junior': 96,
    'Representante de Ventas Senior': 97,
    'Director de Gestion de Producto': 98,
    'Redactor Junior': 99,
    'Coordinador de Marketing Senior': 100,
    'Gerente Senior de Recursos Humanos': 101,
    'Asociado de Desarrollo de Negocios Junior': 102,
    'Gerente de Cuentas Senior': 103,
    'Investigador Senior': 104,
    'Coordinador de Recursos Humanos Junior': 105,
```

```
'Director de Finanzas': 106,
'Cientifico de Datos Junior': 107,
'Analista de Operaciones Senior': 108,
'Coordinador de Recursos Humanos Senior': 109,
'Diseñador UX Senior': 110,
'Gerente de Producto Junior': 111,
'Especialista en Marketing Senior': 112,
'Gerente de Proyectos de TI Senior': 113,
'Analista de Aseguramiento de Calidad Senior': 114,
'Ejecutivo de Cuentas Senior': 115,
'Director de Desarrollo de Negocios': 116,
'Gerente de Redes Sociales Junior': 117,
'Especialista en Recursos Humanos Senior': 118,
'Analista de Datos Senior': 119,
'Director de Capital Humano': 120,
'Coordinador de Publicidad Junior': 121,
'Diseñador UX Junior': 122,
'Director de Marketing Senior': 123,
'Generalista de Recursos Humanos Junior': 124,
'Coordinador de Marketing Junior': 125,
'Asesor Financiero Senior': 126,
'Analista de Operaciones de Negocios Junior': 127,
'Especialista en Redes Sociales Junior': 128,
'Gerente de Operaciones Junior': 129,
'Arquitecto de Software Senior': 130,
'Especialista en Marketing Junior': 131,
'Desarrollador de Software Senior': 132,
'Analista de Marketing Junior': 133,
'Consultor de TI Senior': 134,
'Gerente Financiero Senior': 135,
'Gerente de Marketing Junior': 136,
'Coordinador de Operaciones Junior': 137,
'Director de Recursos Humanos': 138,
'Coordinador de Operaciones Senior': 139,
'Ingeniero de Datos Senior': 140,
'Asesor Financiero Junior': 141,
'Director de Ingenieria': 142,
'Ingeniero de Proyectos Senior': 143,
'Ingeniero Full Stack': 144,
'Desarrollador Front End': 145,
'Desarrollador Back End': 146,
'Gerente de Ingenieros de Software': 147,
'Desarrollador Front End': 148,
'Desarrollador de Software': 149,
'Director de Ciencia de Datos': 150,
'Gerente de Marketing': 151,
'Coordinador de Recursos Humanos': 152,
'Asociado de Ventas Junior': 153,
```

```
    'Gerente de Recursos Humanos': 154,
    'Generalista de Recursos Humanos Junior': 155,
    'Coordinador de Recursos Humanos Junior': 156,
    'Gerente Senior de Marketing de Producto': 157,
    'Asociado de Ventas': 158,
    'Gerente de Marketing de Contenidos': 159,
    'Director de Ventas': 160,
    'Representante de Ventas': 161,
    'Cientifico Investigador': 162,
    'Especialista en Marketing Digital': 163,
    'Recepcionista': 164,
    'Director de Marketing': 165,
    'Gerente de Redes Sociales': 166,
    'Representante de Servicio al Cliente': 167,
    'Conductor de Entrega': 168,
    'Ejecutivo de Ventas': 169,
    'Cientifico Investigador Junior': 170,
    'Grente de Operaciones de Ventas': 171
    })
dfNum.head(5)
```

	Edad	Genero	Nivel_Educativo	Titulo_Del_Trabajo	Años_De_Experiencia	Salario
0	32.0	1.0	1.0	1.0	5.0	90000.0
1	28.0	2.0	2.0	2.0	3.0	65000.0
2	45.0	1.0	7.0	3.0	15.0	150000.0
3	36.0	2.0	NaN	NaN	7.0	60000.0
4	52.0	1.0	2.0	4.0	20.0	200000.0

Mostrar un resumen estadístico de los datos (Final)

```python
# Mostrar un resumen estadístico de los datos:
#print(df_Trad.describe())
df_Trad.describe()

```

	Edad	Genero	Nivel_Educativo	Titulo_Del_Trabajo	Salario
count	7122.000000	7122.000000	7120.000000	7103.00000	6971.000000
mean	33.666807	1.454788	4.664747	85.74011	115628.243724
std	7.633494	0.501919	1.777825	64.21260	52565.497372
min	21.000000	1.000000	1.000000	1.00000	350.000000
25%	28.000000	1.000000	4.000000	20.00000	70000.000000
50%	32.000000	1.000000	5.000000	83.00000	115000.000000
75%	38.000000	2.000000	6.000000	148.00000	160000.000000
max	62.000000	3.000000	7.000000	171.00000	250000.000000

```python
# Disminuyendo el número de decimales
df_Trad.describe().style.format(precision=3)

# De aquí podemos ver por ejemplo que:
# El 75% de los datos son Mujeres de 38 años
# El 50% de los datos son Hombres de 32 años
# El 25% de los datos son Hombres de 28 años

```

	Edad	Genero	Nivel_Educativo	Titulo_Del_Trabajo	Salario
count	7122.000	7122.000	7120.000	7103.000	6971.000
mean	33.667	1.455	4.665	85.740	115628.244
std	7.633	0.502	1.778	64.213	52565.497
min	21.000	1.000	1.000	1.000	350.000
25%	28.000	1.000	4.000	20.000	70000.000
50%	32.000	1.000	5.000	83.000	115000.000
75%	38.000	2.000	6.000	148.000	160000.000
max	62.000	3.000	7.000	171.000	250000.000

d) Calcular el porcentaje de valores faltantes por columna:

Utilizar df.isnull().mean() * 100 para calcular el porcentaje de valores faltantes en cada columna, lo que ayuda a determinar la severidad de los problemas de datos.

```
# d) Calcular el porcentaje de valores faltantes por columna:

# Porcentaje de valores faltantes
'''
Edad                   4.029107
Genero                 4.029107
Nivel_Educativo        4.056057
Titulo_Del_Trabajo     4.285137
Años_De_Experiencia    4.056057
Salario                6.063873
dtype: float64
'''
missing_percentage = df_Trad.isnull().mean() * 100
#print(missing_percentage)
missing_percentage

```

	0
Edad	4.029107
Genero	4.029107
Nivel_Educativo	4.056057
Titulo_Del_Trabajo	4.042582
Años_De_Experiencia	4.056057
Salario	6.063873

dtype: float64

e) Identificar si hay filas duplicadas:

Usar df.duplicated().sum() para contar el número de filas duplicadas, lo que es crucial para asegurar que cada observación sea única.

```
# e) Identificar si hay filas duplicadas:

# Total de filas duplicadas
before_total_duplicates = df_Trad.duplicated().sum()
print(f'Total de filas duplicadas inicial: {before_total_duplicates}')
# 4,223
```

```
Total de filas duplicadas inicial: 4223
```

f) Analizar los tipos de datos de las columnas:

Utilizar df.dtypes para verificar que cada columna tenga el tipo de dato esperado, lo cual es importante para evitar errores en los análisis posteriores.

```python
# f) Analizar los tipos de datos de las columnas:

# Tipos de datos
'''
Age                    float64
Gender                  object
Education Level         object
Job Title               object
Years of Experience     object
Salary                 float64
dtype: object
'''
print(df.dtypes)
#df_Trad.dtypes

# --->>> Edad debe de ser int
# --->>> Años_De_Experiencia de ser int

```

```
Age                    float64
Gender                  object
Education Level         object
Job Title               object
Years of Experience     object
Salary                 float64
dtype: object
```

3. Limpieza de Datos

```
# Se genera df para limpieza de datos
df_Clean = df
df_Clean.shape # 7,421

```

```
(7421, 6)
```

Realizar las siguientes tareas de limpieza:

a) Eliminación o imputación de valores faltantes:

```
# a) Eliminación o imputación de valores faltantes:

# Identificar valores faltantes
missing_values_before = df_Clean.isnull().sum()
#print(missing_values_before)
missing_values_before

```

	0
Edad	299
Genero	299
Nivel_Educativo	301
Titulo_Del_Trabajo	318
Años_De_Experiencia	301
Salario	450

```
# Se genera df para Cambiar los datos nulos
df_DropNa = df_Trad

```

```
df_DropNa.shape # No se elimina ningún dato
# 7,421

```

```
(7421, 6)
```

```python
# Se crea una lista de todos los nombres de las columnas
lista_col = df_DropNa.columns
lista_col

```

```
Index(['Edad', 'Genero', 'Nivel_Educativo', 'Titulo_Del_Trabajo',
       'Años_De_Experiencia', 'Salario'],
      dtype='object')
```

```python
# Hay que validar que NO debe de tener NaN

for nom_colum in lista_col:
    df_DropNa= df_DropNa.dropna(subset=[nom_colum])

```

```python
df_DropNa.shape # No se elimina ningún dato
# 5,688

```

```
(5688, 6)
```

```python
# Inicialmente se tenían 7,421 registros y al utilizar el dropna quedarían 5,688 por lo que perderíamos 1,733

 # --->>> Se recomienda No utilizar dropna

# De acuerdo al Análisis, se va a reemplazar o rellenar los datos NaN para después decidir si se utilizan
# se requiere hacer el cambio de los NaN, para la modificación de los Tipos de Datos de las columnas.

df_Clean.shape # Por lo que seguiremos trabajando el df de la Limpieza en donde No se elimina ningún dato

```

```
(7421, 6)
```

b) Calcular el porcentaje de valores faltantes por columna

Porcentaje de valores faltantes (Inicial)

```
# Porcentaje de valores faltantes por columna

'''
Edad                   4.029107
Genero                 4.029107
Nivel_Educativo        4.056057
Titulo_Del_Trabajo     4.042582
Años_De_Experiencia    4.056057
Salario                6.063873
dtype: float64
'''
#Porcentaje de valores faltantes por columna:
missing_percentage_before = df_Clean.isnull().mean() * 100
#print(missing_percentage_before)
missing_percentage_before

```

	0
Edad	4.029107
Genero	4.029107
Nivel_Educativo	4.056057
Titulo_Del_Trabajo	4.042582
Años_De_Experiencia	4.056057
Salario	6.063873

dtype: float64

```
df_Clean.info()
# El número de registros de cada columna NO es igual al número de renglones 7,421 dado que tiene NaN

```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7421 entries, 0 to 7420
Data columns (total 6 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Edad                 7122 non-null   float64
 1   Genero               7122 non-null   object
 2   Nivel_Educativo      7120 non-null   object
 3   Titulo_Del_Trabajo   7121 non-null   object
 4   Años_De_Experiencia  7120 non-null   object
 5   Salario              6971 non-null   float64
dtypes: float64(2), object(4)
memory usage: 348.0+ KB
```

```
# Se crea una lista de todos los nombres de las columnas
lista_col_dNa = df_Clean.columns
lista_col_dNa
```

```
Index(['Age', 'Gender', 'Education Level', 'Job Title', 'Years of Experience',
       'Salary'],
      dtype='object')
```

```
# Verificando si hay invalid_value
for i in lista_col_dNa:
    print(f"En la columna {i} los invalid_value son: {df_Clean[df_Clean[i] == 'invalid_value'].shape[0]}")

```

```
En la columna Age los invalid_value son: 0
En la columna Gender los invalid_value son: 0
En la columna Education Level los invalid_value son: 0
En la columna Job Title los invalid_value son: 0
En la columna Years of Experience los invalid_value son: 0
En la columna Salary los invalid_value son: 0
```

Reemplazar NaN por valores validos

```
df_Clean.shape
# 7,421

```

```
(7421, 6)
```

```
df_Clean.columns

```

```
Index(['Edad', 'Genero', 'Nivel_Educativo', 'Titulo_Del_Trabajo',
       'Años_De_Experiencia', 'Salario'],
      dtype='object')
```

```
# Reemplazar NaN por valores validos -->> 0 (cero)
# df_Clean['Edad'].fillna(0, inplace=True) # --->>> No tenemos registros con Edad 0 años (NO es valor de la columna)

```

```
# Reemplazar NaN por valores validos -->> 'Other'
df_Clean['Genero'].fillna("Otro", inplace=True) # --->>> Si tenemos más registros con Genero= Otro (si es valor de la columna)

```

```
<ipython-input-63-f4f7079d867c>:2: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting values always behaves as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value) instead, to perform


  df_Clean['Genero'].fillna("Otro", inplace=True) # --->>> Si tenemos más registros con Genero= Otro (si es valor de la columna)
<ipython-input-63-f4f7079d867c>:2: FutureWarning: Setting an item of incompatible dtype is deprecated and will raise an error in a future version of pandas. Value 'Otr
  df_Clean['Genero'].fillna("Otro", inplace=True) # --->>> Si tenemos más registros con Genero= Otro (si es valor de la columna)
```

```python
# Reemplazar NaN por valores validos -->> 'Other'
df_Clean['Nivel_Educativo'].fillna("Otro", inplace=True) # --->>> Podría no ser indispensable tener el Nivel Educativo

```

```
<ipython-input-64-1c5d8e8f642e>:2: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting values always behaves as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value) instead, to perform


  df_Clean['Nivel_Educativo'].fillna("Otro", inplace=True) # --->>> Podría no ser indispensable tener el Nivel Educativo
<ipython-input-64-1c5d8e8f642e>:2: FutureWarning: Setting an item of incompatible dtype is deprecated and will raise an error in a future version of pandas. Value 'Otr
  df_Clean['Nivel_Educativo'].fillna("Otro", inplace=True) # --->>> Podría no ser indispensable tener el Nivel Educativo
```

```python
# Reemplazar NaN por valores validos -->> 'Other'
df_Clean['Titulo_Del_Trabajo'].fillna("Otro", inplace=True) # --->>> Podría no ser indispensable tener el Títul del Trabajo

```

```
<ipython-input-65-57dac0427d3b>:2: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting values always behaves as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value) instead, to perform


  df_Clean['Titulo_Del_Trabajo'].fillna("Otro", inplace=True) # --->>> Podría no ser indispensable tener el Títul del Trabajo
<ipython-input-65-57dac0427d3b>:2: FutureWarning: Setting an item of incompatible dtype is deprecated and will raise an error in a future version of pandas. Value 'Otr
  df_Clean['Titulo_Del_Trabajo'].fillna("Otro", inplace=True) # --->>> Podría no ser indispensable tener el Títul del Trabajo
```

```python
# Reemplazar NaN por valores validos -->> 0 (cero)
df_Clean['Años_De_Experiencia'].fillna(0, inplace=True)  # --->>> Si tenemos más registros con Años de Expeciencia = 0 (si es valor de la columna)

```

```
<ipython-input-66-e1033a4b0257>:2: FutureWarning: A value is trying to be set on a copy of a DataFrame or Series through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting values always behaves as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value) instead, to perform


  df_Clean['Años_De_Experiencia'].fillna(0, inplace=True)  # --->>> Si tenemos más registros con Años de Expeciencia = 0 (si es valor de la columna)
```

```python
# Reemplazar NaN por valores validos -->> 0 (cero)
#df_Clean['Salario'].fillna(0, inplace=True)  #  --->>> No tenemos registros con Edad 0 años (NO es valor de la columna)

```

Validar Porcentaje de valores faltantes

```
# Porcentaje de valores faltantes por columna:

'''                     Original
Edad                    4.029107  --->>> NO Cambia a 0  # --->>> No tenemos registros con Edad 0 años (NO es valor de la columna)
Genero                  4.029107  --->>> Cambia a 0
Nivel_Educativo         4.056057  --->>> Cambia a 0
Titulo_Del_Trabajo      4.042582  --->>> Cambia a 0
Años_De_Experiencia     4.056057  --->>> Cambia a 0
Salario                 6.063873  --->>> NO Cambia a 0  # --->>> No tenemos registros con Edad 0 años (NO es valor de la columna)
dtype: float64
'''
# Porcentaje de valores faltantes
missing_percentage_after = df_Clean.isnull().mean() * 100
#print(missing_percentage_after)
missing_percentage_after
```

	0
Edad	4.029107
Genero	0.000000
Nivel_Educativo	0.000000
Titulo_Del_Trabajo	0.000000
Años_De_Experiencia	0.000000
Salario	6.063873

dtype: float64

Eliminar NaN

```
df_Clean.shape
# 7,421
```

```
# Se genera df para Cambiar los datos nulos
df_DropNa = df_Clean

```

```
# Se crea una lista de todos los nombres de las columnas
#lista_col = df_DropNa.columns
lista_col_dNa = ['Edad',  'Salario']
lista_col_dNa

```

```
df_DropNa.shape
# 6,693

```

```
(6693, 6)
```

Porcentaje de valores faltantes (Final)

```
# Porcentaje de valores faltantes por columna:

'''                     Original
Edad                    4.029107  --->>> Cambia a 0  # --->>> Se borran con dropna
Genero                  0.000000
Nivel_Educativo         0.000000
Titulo_Del_Trabajo      0.000000
Años_De_Experiencia     0.000000
Salario                 6.063873  --->>> Cambia a 0  # --->>> Se borran con dropna
dtype: float64
'''
# Porcentaje de valores faltantes
missing_percentage_after = df_DropNa.isnull().mean() * 100
#print(missing_percentage_after)
missing_percentage_after

```

	0
Edad	0.0
Genero	0.0
Nivel_Educativo	0.0
Titulo_Del_Trabajo	0.0
Años_De_Experiencia	0.0
Salario	0.0

dtype: float64

```
df_DropNa.info()

# Como ya se reemplazaron todos los datos NaN, se puede observar que
# el número de registros es igual al número de renglones 7,421

```

```
<class 'pandas.core.frame.DataFrame'>
Index: 6693 entries, 0 to 7420
Data columns (total 6 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Edad                 6693 non-null   float64
 1   Genero               6693 non-null   object
 2   Nivel_Educativo      6693 non-null   object
 3   Titulo_Del_Trabajo   6693 non-null   object
 4   Años_De_Experiencia  6693 non-null   object
 5   Salario              6693 non-null   float64
dtypes: float64(2), object(4)
memory usage: 366.0+ KB
```

c) Eliminación de duplicados:

Identificar y eliminar filas duplicadas usando df.drop_duplicates(), garantizando que los datos sean únicos.

```
# Se crea  df después de Reemplazar NaN
df_Dupli = df_DropNa
df_Dupli.shape

```

```
(6693, 6)
```

```
#Retomando la lista de todos los nombres de las columnas
lista_col = df_Dupli.columns
lista_col

```

```
Index(['Age', 'Gender', 'Education Level', 'Job Title', 'Years of Experience',
       'Salary'],
      dtype='object')
```

```
# c) Eliminación de duplicados:

duplicates_before = df_Dupli.duplicated().sum()
df_Dupli.drop_duplicates(inplace=True) # Eliminar filas duplicadas
duplicates_after = df_Dupli.duplicated().sum()
print(f"Número de Duplicados antes: {duplicates_before}, Número de Duplicados después: {duplicates_after}")

```

```
Número de Duplicados antes: 4207, Número de Duplicados después: 0
```

```
# Validar el número de registros después de Eliminar duplicados
df_Dupli.shape # 7,421 - 4,207 = 2,486

```

```
df_Dupli.info() # el número de registros es igual al número de renglones

```

```
<class 'pandas.core.frame.DataFrame'>
Index: 2486 entries, 0 to 7415
Data columns (total 6 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   Edad                2486 non-null   float64
 1   Genero              2486 non-null   object 
 2   Nivel_Educativo     2486 non-null   object 
 3   Titulo_Del_Trabajo  2486 non-null   object 
 4   Años_De_Experiencia 2486 non-null   object 
 5   Salario             2486 non-null   float64
dtypes: float64(2), object(4)
memory usage: 136.0+ KB
```

d) Corrección de tipos de datos:

Asegurarse de que las columnas tengan tipos de datos adecuados, utilizando astype() para convertir tipos incorrectos, por ejemplo, asegurando que la columna de edad sea un entero.

```
# d) Corrección de tipos de datos:

# Verificar tipos de datos
print(df_Dupli.dtypes)

```

```
Edad                  float64
Genero                 object
Nivel_Educativo        object
Titulo_Del_Trabajo     object
Años_De_Experiencia    object
Salario               float64
dtype: object
```

```
df_Dupli.info()

```

```
<class 'pandas.core.frame.DataFrame'>
Index: 2486 entries, 0 to 7415
Data columns (total 6 columns):
 #   Column               Non-Null Count  Dtype  
---  ------               --------------  -----  
 0   Edad                 2486 non-null   float64
 1   Genero               2486 non-null   object 
 2   Nivel_Educativo      2486 non-null   object 
 3   Titulo_Del_Trabajo   2486 non-null   object 
 4   Años_De_Experiencia  2486 non-null   object 
 5   Salario              2486 non-null   float64
dtypes: float64(2), object(4)
memory usage: 136.0+ KB
```

```
# Convertir tipos de datos
# Asegurar que la columna 'Edad' tenga tipo de datos adecuado.
df_Dupli['Edad'] = df_Dupli['Edad'].astype(int)  # Asegurar que 'Edad' sea int

```

```
# Convertir tipos de datos
# Asegurar que las columnas 'Salario' tenga tipo de datos adecuado.
df_Dupli['Salario'] = df_Dupli['Salario'].astype(float)  # Asegurar que 'Salario' sea float

# --->>> La columna de 'Salario' tiene la cadena 'bbb' por lo que se tiene que corregir los valores inválidos

```

```python
# Convertir tipos de datos
# Asegurar que la columna 'Años_De_Experiencia' tenga tipo de datos adecuado.
df_Dupli['Años_De_Experiencia'] = df_Dupli['Años_De_Experiencia'].astype(float)
# Asegurar que 'Años_De_Experiencia' sea int antes verifiquemos que sea float

# --->>> La columna de 'Años_De_Experiencia' tiene la cadena 'bbb'
# por lo que se tiene que corregir los valores inválidos

```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-90-0b1b878130b2> in <cell line: 3>()
      1 # Convertir tipos de datos
      2 # Asegurar que la columna 'Años_De_Experiencia' tenga tipo de datos adecuado.
----> 3 df_Dupli['Años_De_Experiencia'] = df_Dupli['Años_De_Experiencia'].astype(float)
      4 # Asegurar que 'Años_De_Experiencia' sea int antes verifiquemos que sea float
      5

                                 6 frames
/usr/local/lib/python3.10/dist-packages/pandas/core/dtypes/astype.py in _astype_nansafe(arr, dtype, copy, skipna)
    131     if copy or arr.dtype == object or dtype == object:
    132         # Explicit copy, or required since NumPy can't view from / to object.
--> 133         return arr.astype(dtype, copy=True)
    134
    135     return arr.astype(dtype, copy=copy)

ValueError: could not convert string to float: 'bbb'
```

e) Corrección de valores "inválidos":

Identificar y corregir valores incorrectos, como cadenas erróneas ('bbb'), reemplazándolos con un valor adecuado o eliminándolos.

```python
# Se genera df para Cambiar los datos nulos
df_Final = df_Dupli

```

```python
df_Final.shape
# 2,486
```

```python
# e) Corrección de valores "inválidos":

# Reemplazar valores inválidos
# df['Years of Experience'].replace('bbb', pd.NA, inplace=True)  # Reemplazar 'bbb' con NaN

df_Final['Años_De_Experiencia'].replace('bbb', pd.NA, inplace=True)  # Reemplazar 'bbb' con NaN

```

```python
# Reemplazar NaN por valores validos
df_Final['Años_De_Experiencia'].fillna(0, inplace=True)  # --->>> Si tenemos más registros con Años de Expeciencia = 0 (si es valor de la columna)

```

```python
# Convertir tipos de datos
df_Final['Años_De_Experiencia'] = df_Final['Años_De_Experiencia'].astype(float)  # Asegurar que 'Años_De_Experiencia' sea float

```

```python
# Convertir tipos de datos
df_Final['Años_De_Experiencia'] = df_Final['Años_De_Experiencia'].astype(int)  # Asegurar que 'Años_De_Experiencia' sea int

```

```python
df_Final.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 2486 entries, 0 to 7415
Data columns (total 6 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   Edad                2486 non-null   int64  
 1   Genero              2486 non-null   object 
 2   Nivel_Educativo     2486 non-null   object 
 3   Titulo_Del_Trabajo  2486 non-null   object 
 4   Años_De_Experiencia  2486 non-null   int64  
 5   Salario             2486 non-null   float64
dtypes: float64(1), int64(2), object(3)
memory usage: 136.0+ KB
```

```python
# Convertir tipos de datos
df_Final['Genero'] = df_Final['Genero'].astype(str)  # Asegurar que 'Genero' sea str

```

```python
# Convertir tipos de datos
df_Final['Nivel_Educativo'] = df_Final['Nivel_Educativo'].astype(str)  # Asegurar que 'Nivel_Educativo' sea str

```

```python
# Convertir tipos de datos
df_Final['Titulo_Del_Trabajo'] = df_Final['Titulo_Del_Trabajo'].astype(str)  # Asegurar que 'Titulo_Del_Trabajo' sea str

```

```python
df_Final.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 2486 entries, 0 to 7415
Data columns (total 6 columns):
 #   Column              Non-Null Count  Dtype  
---  ------              --------------  -----  
 0   Edad                2486 non-null   int64  
 1   Genero              2486 non-null   object 
 2   Nivel_Educativo     2486 non-null   object 
 3   Titulo_Del_Trabajo  2486 non-null   object 
 4   Años_De_Experiencia  2486 non-null   int64  
 5   Salario             2486 non-null   float64
dtypes: float64(1), int64(2), object(3)
memory usage: 136.0+ KB
```

4. Guardar CSV

```
# Guardar resultados en un CSV
df_Final.to_csv("Base_fs_limpia_MPAC.csv", index=False)
```

https://drive.google.com/drive/folders/1IvaHSNnYQ95-h8gpXAcQHXEZw4l7T-Pw