

Data Cleaning : students / mentor seekers from Chile

In [1]: *#1) importing libraries for data cleaning, analysis and exploration*

```
In [2]: import seaborn as sns
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from scipy.stats.stats import pearsonr
import plotly.express as px
%matplotlib inline
import plotly.graph_objects as go
import numpy as np
```

In [3]: *#2) import dataset obtained from the survey in GoogleForm converted into .csv format*

In [4]: `df1 = pd.read_csv('mentees_esp.csv')`

In [5]: *#3) overview of data, n° of columns, rows and type of variables to evaluate what data types will be handled*

In [6]: `df1.head(2)`

Out[6]:

	Timestamp	intelligence_type	mentor_attribute	mentor_rol	mentor_career	mentor_skills	session_freq	mentor_age	mentor_
0	2021/06/16 8:42:43 am EET	Inteligencia Existencial: ¿Te intriga el senti...	Calma	Ofrecer orientación;Entregar apoyo en la toma ...	Tecnología de la Información (TI)	Conocimientos especializados en el área selecc...	Dos veces al mes (15 - 30 minutos de sesión)	No es relevante	No es re
1	2021/06/21 5:22:07 pm EET	Inteligencia Natural: ¿Te llama la atención la...	Empatía	Ofrecer orientación;Entregar apoyo en la toma ...	Arte	Habilidad para enseñar	Dos veces al mes (15 - 30 minutos de sesión)	23-29	No es re

In [7]: *#3.a) proof of data types and check for null values*

In [8]: df1.info()

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 64 entries, 0 to 63
Data columns (total 11 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Timestamp              64 non-null    object
1   intelligence_type       64 non-null    object
2   mentor_attribute        64 non-null    object
3   mentor_rol              64 non-null    object
4   mentor_career           64 non-null    object
5   mentor_skills           64 non-null    object
6   session_freq            64 non-null    object
7   mentor_age              64 non-null    object
8   mentor_gender            64 non-null    object
9   mentee_age              64 non-null    object
10  mentee_gender            64 non-null    object
dtypes: object(11)
memory usage: 5.6+ KB
```

Now we check most repeated values, unique values, among others. For instance its possible to see from ".describe" that there are 41 intelligence types, which is certainly not correct as we worked with only 9, however the user inputs represent 41 probable combination of intelligence type. This give us insights about what data must be handled and how.

In [9]: df1.describe()

Out[9]:

	Timestamp	intelligence_type	mentor_attribute	mentor_rol	mentor_career	mentor_skills	session_freq	mentor_age	mentor_gender
count	64	64	64	64	64	64	64	64	64
unique	64	41	5	15	10	5	4	4	4
top	2021/06/25 1:30:19 am EET	Inteligencia Intrapersonal: ¿Reflexionas frecu...	Empatía	Ofrecer orientación; Tener disponibilidad para ...	Medicina	Habilidad para enseñar	Una vez por semana (sesión de 15 a 30 minutos)	No es relevante	No es
freq	1	3	20	11	19	36	39	40	40

In [10]: df1.isnull() # evaluate if there are null values in the data

Out[10]:

	Timestamp	intelligence_type	mentor_attribute	mentor_rol	mentor_career	mentor_skills	session_freq	mentor_age	mentor_gender
0	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False
...
59	False	False	False	False	False	False	False	False	False
60	False	False	False	False	False	False	False	False	False
61	False	False	False	False	False	False	False	False	False
62	False	False	False	False	False	False	False	False	False
63	False	False	False	False	False	False	False	False	False

64 rows × 11 columns

4 it must be said that it does not make much sense to explore data which is still not translated and cleaned. It is important though, to know if there are any unexpected strings or inputs which are stored for instance as upper case when they should be in lower case. Thats why a preliminar exploration must be done but detailed analysis will be carried on with the final data set

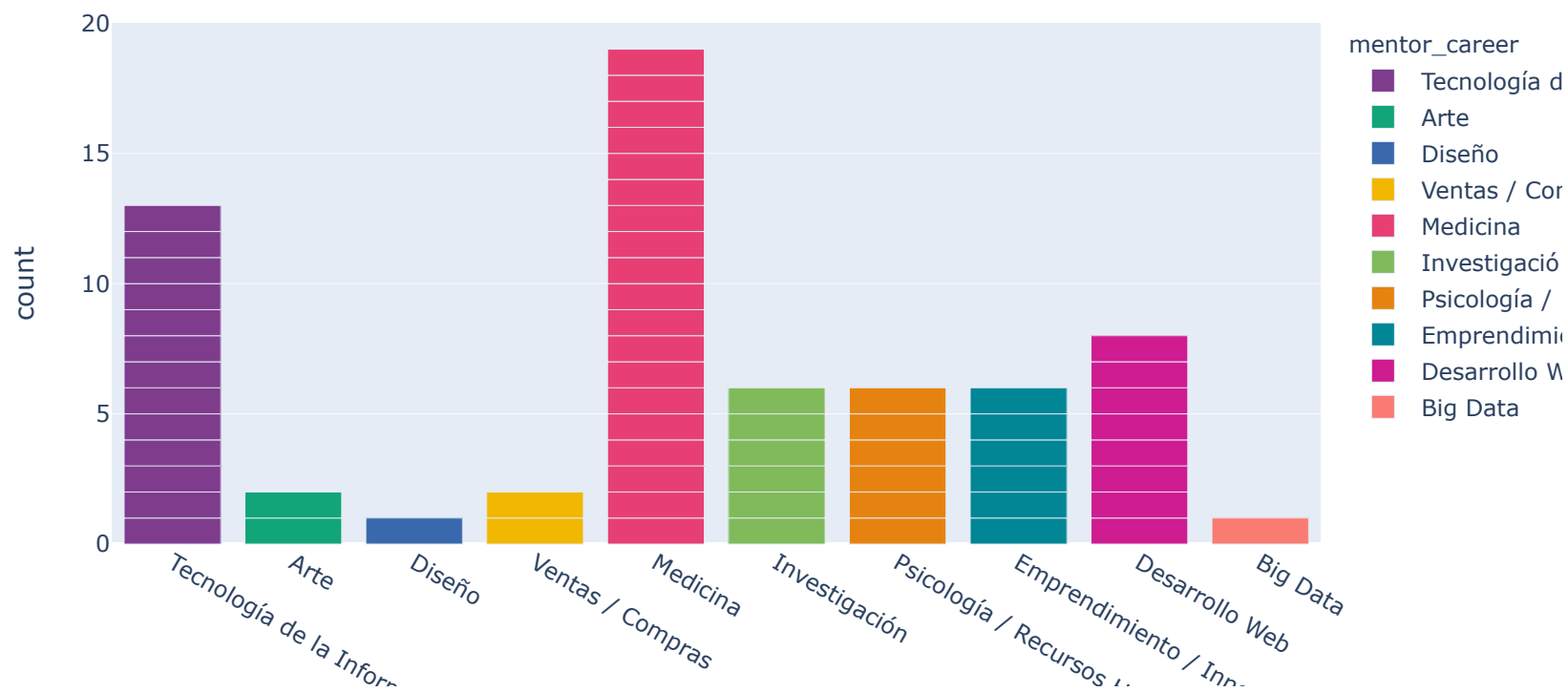
```
In [11]: #Data exploration - see top ranked answers
```

```
In [12]: df1['mentor_career'].value_counts()
```

```
Out[12]: Medicina 19
Tecnología de la Información (TI) 13
Desarrollo Web 8
Emprendimiento / Innovación 6
Psicología / Recursos Humanos 6
Investigación 6
Ventas / Compras 2
Arte 2
Big Data 1
Diseño 1
Name: mentor_career, dtype: int64
```

```
In [13]: # Most interesting fields selected by mentees
fig = px.bar(df1, x="mentor_career", title="Most interesting professional fields", color="mentor_career", color_discrete_sequence=px.colors.qualitative.Bold)
fig.show()
```

Most interesting professional fields



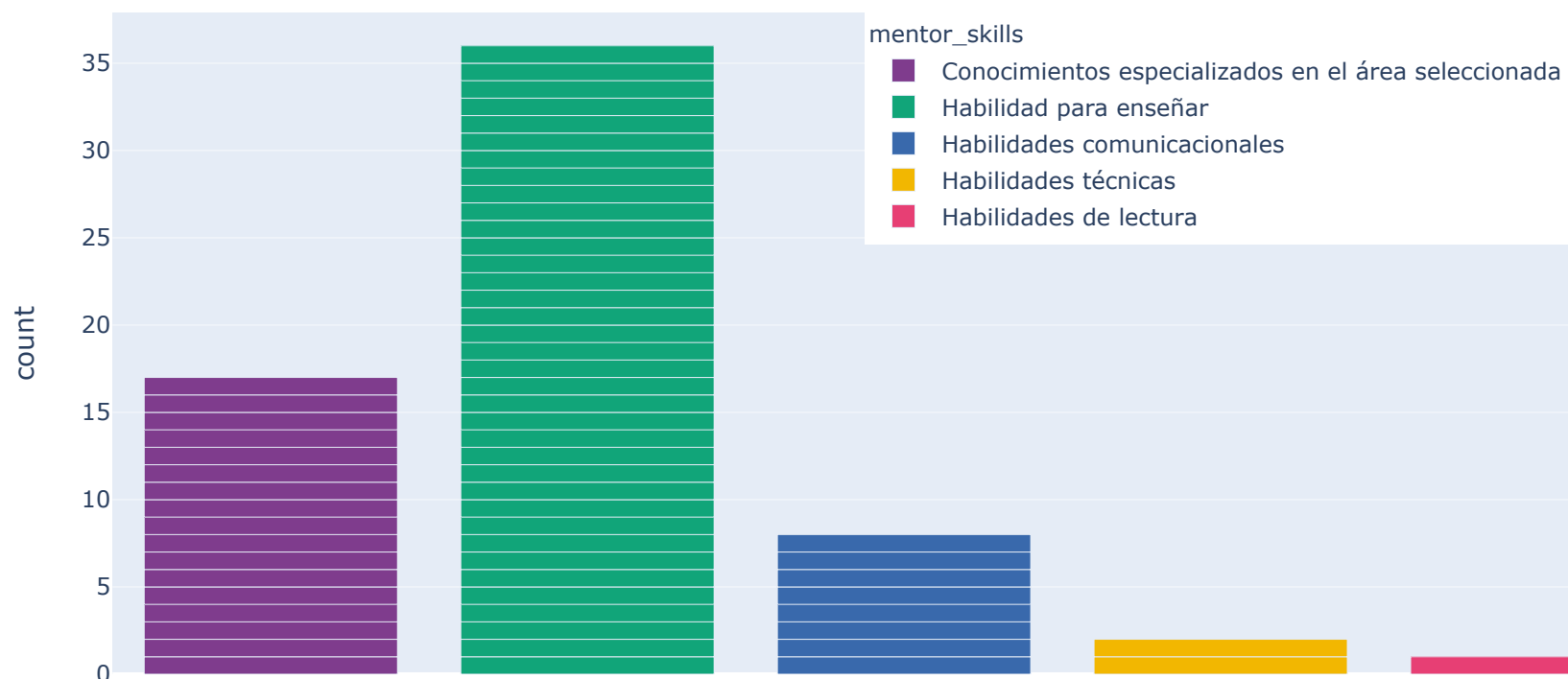
```
In [14]: # top chosen values for mentors
```

```
In [15]: df1['mentor_skills'].value_counts()
```

```
Out[15]: Habilidad para enseñar          36  
Conocimientos especializados en el área seleccionada en la pregunta anterior  17  
Habilidades comunicacionales          8  
Habilidades técnicas                  2  
Habilidades de lectura                1  
Name: mentor_skills, dtype: int64
```

```
In [16]: # Most interesting fields selected by mentees
fig = px.bar(df1, x="mentor_skills", title="Most valuable skills which a mentor should have", color="mentor_s
kills", color_discrete_sequence=px.colors.qualitative.Bold)
fig.show()
```

Most valuable skills which a mentor should have



5 Data cleaning Intelligence Types - converting the values into a list and separate each input by columns (users were allowed to give up to three inputs for intelligence types)

```
In [17]: df2 = df1['intelligence_type'].str.split(';', expand=True) # create a list separating by ";"
df2.head(10)
```

Out[17]:

	0	1	2
0	Inteligencia Existencial: ¿Te intriga el senti...	Inteligencia Interpersonal: ¿Te gusta estar co...	Inteligencia Intrapersonal: ¿Reflexionas frecu...
1	Inteligencia Natural: ¿Te llama la atención la...	Inteligencia Musical: ¿Te sientes atraído por ...	Inteligencia Intrapersonal: ¿Reflexionas frecu...
2	Inteligencia Musical: ¿Te sientes atraído por ...	Inteligencia Existencial: ¿Te intriga el senti...	Inteligencia Interpersonal: ¿Te gusta estar co...
3	Inteligencia Musical: ¿Te sientes atraído por ...	Inteligencia Matemática: ¿Te llama la atención...	Inteligencia Existencial: ¿Te intriga el senti...
4	Inteligencia Existencial: ¿Te intriga el senti...	None	None
5	Inteligencia Natural: ¿Te llama la atención la...	Inteligencia lingüística: ¿Te gustan los idiom...	Inteligencia visual-espacial: ¿Te interesan la...
6	Inteligencia corporal-kinestésica: ¿Amas los d...	None	None
7	Inteligencia Musical: ¿Te sientes atraído por ...	Inteligencia Matemática: ¿Te llama la atención...	Inteligencia Intrapersonal: ¿Reflexionas frecu...
8	Inteligencia Musical: ¿Te sientes atraído por ...	Inteligencia lingüística: ¿Te gustan los idiom...	Inteligencia visual-espacial: ¿Te interesan la...
9	Inteligencia Interpersonal: ¿Te gusta estar co...	None	None

general values are defined for replacing the whole text string by intelligence type to ensure uniform data across the 4 datasets - the chosen variables will be single words that represent the intelligence type, e.g. "natural", "musical" and so on.


```
In [18]: df3 = df2.replace(regex=[ r'^Inteligencia Natural: ¿Te llama la atención la naturaleza y sientes que te relaj
a?.$': 'natural',
                                r'^Inteligencia Musical: ¿Te sientes atraído por la música y por entenderla & crearla
a?.$': 'musical',
                                r'^Inteligencia Matemática: ¿Te llama la atención la simetría de las cosas, unir dif
erentes elementos o los juegos de estrategia?.$': 'mathematical',
                                r'^Inteligencia Existencial: ¿Te intriga el sentido de la vida o saber que más hay a
llá en el universo?.$': 'existencial',
                                r'^Inteligencia Interpersonal: ¿Te gusta estar con gente e interactuar con diferente
s personas?.$': 'interpersonal',
                                r'^Inteligencia corporal-kinestésica: ¿Amas los deportes o crear cosas con tus mano
s?.$': 'corporal',
                                r'^Inteligencia lingüística: ¿Te gustan los idiomas, los juegos de palabras o contar
historias?.$': 'linguistic',
                                r'^Inteligencia visual-espacial: ¿Te interesan laberintos, rompecabezas o puedes vis
ualizar el resultado de algo antes de hacerlo?.$': 'visual_spacial',
                                r'^Inteligencia Intrapersonal: ¿Reflexionas frecuentemente sobre tus emociones? ¿Te
gusta planear cuidadosamente tus metas?.$': 'intrapersonal',
                                ])
```

```
In [19]: #renaming columns defining "itype" as equivalent to intelligence type
df3.rename(columns={0: "itype1", 1: "itype2", 2: 'itype3'}, inplace=True)
```

```
In [20]: df3.fillna(0) # search for empty values
```

```
Out[20]:
```

	itype1	itype2	itype3
0	existencial	interpersonal	Inteligencia Intrapersonal: ¿Reflexionas frecu...
1	natural	musical	Inteligencia Intrapersonal: ¿Reflexionas frecu...
2	musical	existencial	interpersonal
3	musical	mathematical	existencial
4	existencial	0	0
...
59	natural	mathematical	Inteligencia Intrapersonal: ¿Reflexionas frecu...
60	musical	mathematical	corporal
61	musical	mathematical	corporal
62	mathematical	existencial	visual_spacial
63	mathematical	existencial	visual_spacial

64 rows × 3 columns

somehow working with REGEX was fine for all of the data types from intelligence types but it did not work for intrapersonal.

Therefore this was a way around using "string contains" to force the change and have cleaned data.

locate intrapersonal and change all values where Intelligence intrapersonal is true to only intrapersonal

```
In [21]: df3.loc[df3['itype3'].str.contains('Intrapersonal') == True, 'itype3'] = 'intrapersonal'
```

this is the new dataframe for intelligence types which will be concatenated to the main one

```
In [22]: df3.head()
```

Out[22]:

	itype1	itype2	itype3
0	existencial	interpersonal	intrapersonal
1	natural	musical	intrapersonal
2	musical	existencial	interpersonal
3	musical	mathematical	existencial
4	existencial	None	None

```
In [23]: #concatanate both dataframes
```

```
In [24]: df4 = pd.concat([df1, df3], axis=1)
```

```
In [25]: # now we can drop columns that are not needed (timestamp, and also intelligence_type as now we have 3 columns  
itype1,2,3)  
df4.drop(['Timestamp', 'intelligence_type'], axis=1, inplace=True)
```

In [26]: df4.head()

Out[26]:

	mentor_attribute	mentor_rol	mentor_career	mentor_skills	session_freq	mentor_age	mentor_gender	mentee_age	mentee
0	Calma	Ofrecer orientación;Entregar apoyo en la toma ...	Tecnología de la Información (TI)	Conocimientos especializados en el área selecc...	Dos veces al mes (15 - 30 minutos de sesión)	No es relevante	No es relevante	25	
1	Empatía	Ofrecer orientación;Entregar apoyo en la toma ...	Arte	Habilidad para enseñar	Dos veces al mes (15 - 30 minutos de sesión)	23-29	No es relevante	22	
2	Calma	Ofrecer orientación;Motivar	Diseño	Habilidades comunicacionales	Dos veces al mes (15 - 30 minutos de sesión)	30-36	No es relevante	26	
3	Calma	Ayudar a resolver problemas;Motivar	Tecnología de la Información (TI)	Habilidad para enseñar	Una vez al mes (sesión de 30 a 60 minutos)	No es relevante	No es relevante	18	
4	Entusiasmo	Ofrecer orientación;Ayudar a resolver problemas	Tecnología de la Información (TI)	Habilidad para enseñar	Una vez por semana (sesión de 15 a 30 minutos)	23-29	No es relevante	18	

```
In [27]: # same data cleaning process will be for the column "mentor_rol" as users were able to give up to two priorities
df5 = df4['mentor_rol'].str.split(';', expand=True)
df5.head(10)
```

Out[27]:

	0	1
0	Ofrecer orientación	Entregar apoyo en la toma de decisiones
1	Ofrecer orientación	Entregar apoyo en la toma de decisiones
2	Ofrecer orientación	Motivar
3	Ayudar a resolver problemas	Motivar
4	Ofrecer orientación	Ayudar a resolver problemas
5	Ofrecer orientación	Ayudar a resolver problemas
6	Motivar	None
7	Entregar apoyo en la toma de decisiones	Tener disponibilidad para contestar preguntas
8	Ofrecer orientación	Ayudar a resolver problemas
9	Entregar apoyo en la toma de decisiones	Ayudar a resolver problemas

here the name itype was considered indiscriminately, despite the fact it is not an intelligence type variable.

this will be however modified afterwards

```
In [28]: df5.rename(columns={0: "itype4", 1: "itype5"}, inplace=True)
df5.fillna(0) # check for null values
```

Out[28]:

	itype4	itype5
0	Ofrecer orientación	Entregar apoyo en la toma de decisiones
1	Ofrecer orientación	Entregar apoyo en la toma de decisiones
2	Ofrecer orientación	Motivar
3	Ayudar a resolver problemas	Motivar
4	Ofrecer orientación	Ayudar a resolver problemas
...
59	Ofrecer orientación	0
60	Entregar apoyo en la toma de decisiones	Tener disponibilidad para contestar preguntas
61	Entregar apoyo en la toma de decisiones	Tener disponibilidad para contestar preguntas
62	Ofrecer orientación	Tener disponibilidad para contestar preguntas
63	Ofrecer orientación	Tener disponibilidad para contestar preguntas

64 rows × 2 columns

Now we handle the data of mentor_role with string contains as a more efficient way of handling data and changing the names into an uniform variable name by each column by that will be assigned to all the variables in english from the 4 datasets

```
In [29]: df5.loc[df5['itype4'].str.contains('resolver') == True, 'itype4'] = 'solve_problems'
df5.loc[df5['itype5'].str.contains('resolver') == True, 'itype5'] = 'solve_problems'
df5.loc[df5['itype4'].str.contains('apoyo') == True, 'itype4'] = 'support'
df5.loc[df5['itype5'].str.contains('apoyo') == True, 'itype5'] = 'support'
df5.loc[df5['itype4'].str.contains('disponibilidad') == True, 'itype4'] = 'availability'
df5.loc[df5['itype5'].str.contains('disponibilidad') == True, 'itype5'] = 'availability'
df5.loc[df5['itype4'].str.contains('freceer') == True, 'itype4'] = 'guidance'
df5.loc[df5['itype5'].str.contains('freceer') == True, 'itype5'] = 'guidance'
df5.loc[df5['itype4'].str.contains('tivar') == True, 'itype4'] = 'motivate'
df5.loc[df5['itype5'].str.contains('tivar') == True, 'itype5'] = 'motivate'
```

```
In [30]: # this is how the dataset looks like now, and this will be concatenated to the other dataframe
df5.head(10)
```

Out[30]:

	itype4	itype5
0	guidance	support
1	guidance	support
2	guidance	motivate
3	solve_problems	motivate
4	guidance	solve_problems
5	guidance	solve_problems
6	motivate	None
7	support	availability
8	guidance	solve_problems
9	support	solve_problems

here the data is already prepared in terms of list separations, and now it is possible to translate the rest of data into english

```
In [31]: # concatenating mentor_role dataset to general data_set having a final one with variables in english and spanish.
df_final = pd.concat([df4, df5], axis=1)
```

```
In [32]: df_final.head(2)
```

```
Out[32]:
```

	mentor_attribute	mentor_rol	mentor_career	mentor_skills	session_freq	mentor_age	mentor_gender	mentee_age	mentee_c
0	Calma	Ofrecer orientación;Entregar apoyo en la toma ...	Tecnología de la Información (TI)	Conocimientos especializados en el área selecc...	Dos veces al mes (15 - 30 minutos de sesión)	No es relevante	No es relevante	25	F
1	Empatía	Ofrecer orientación;Entregar apoyo en la toma ...	Arte	Habilidad para enseñar	Dos veces al mes (15 - 30 minutos de sesión)	23-29	No es relevante	22	F

```
In [33]: # drop mentor_rol as it was converted into new columns (itype4 and itype5)
df_final.drop(['mentor_rol'], axis=1, inplace=True)
```


In [34]: *#quick overview of the dataset that will be translated*
`df_final.head(5)`

Out[34]:

	mentor_attribute	mentor_career	mentor_skills	session_freq	mentor_age	mentor_gender	mentee_age	mentee_gender	itype1
0	Calma	Tecnología de la Información (TI)	Conocimientos especializados en el área selecc...	Dos veces al mes (15 - 30 minutos de sesión)	No es relevante	No es relevante	25	Hombre	existencial
1	Empatía	Arte	Habilidad para enseñar	Dos veces al mes (15 - 30 minutos de sesión)	23-29	No es relevante	22	Hombre	natural
2	Calma	Diseño	Habilidades comunicacionales	Dos veces al mes (15 - 30 minutos de sesión)	30-36	No es relevante	26	Hombre	musical
3	Calma	Tecnología de la Información (TI)	Habilidad para enseñar	Una vez al mes (sesión de 30 a 60 minutos)	No es relevante	No es relevante	18	Hombre	musical
4	Entusiasmo	Tecnología de la Información (TI)	Habilidad para enseñar	Una vez por semana (sesión de 15 a 30 minutos)	23-29	No es relevante	18	Hombre	existencial

In [35]: *#Translating values into english for column mentor_attribute*
`df_final.loc[df_final['mentor_attribute'].str.contains('alma'), 'mentor_attribute'] = 'calmness'`
`df_final.loc[df_final['mentor_attribute'].str.contains('Empat'), 'mentor_attribute'] = 'empathy'`
`df_final.loc[df_final['mentor_attribute'].str.contains('tusiasmo'), 'mentor_attribute'] = 'enthusiasm'`
`df_final.loc[df_final['mentor_attribute'].str.contains('umildad'), 'mentor_attribute'] = 'modesty'`
`df_final.loc[df_final['mentor_attribute'].str.contains('culosidad'), 'mentor_attribute'] = 'meticulousness'`

```
In [36]: #Translating values into english for column mentor_career
df_final.loc[df_final['mentor_career'].str.contains('iseño'), 'mentor_career'] = 'design'
df_final.loc[df_final['mentor_career'].str.contains('TI'), 'mentor_career'] = 'IT'
df_final.loc[df_final['mentor_career'].str.contains('rte'), 'mentor_career'] = 'art'
df_final.loc[df_final['mentor_career'].str.contains('nabilidad'), 'mentor_career'] = 'sustainability'
df_final.loc[df_final['mentor_career'].str.contains('dicina'), 'mentor_career'] = 'medicine'
df_final.loc[df_final['mentor_career'].str.contains('dimiento'), 'mentor_career'] = 'entrepreneurship'
df_final.loc[df_final['mentor_career'].str.contains('sicolo'), 'mentor_career'] = 'psychology/rrhh'
df_final.loc[df_final['mentor_career'].str.contains('keting'), 'mentor_career'] = 'marketing'
df_final.loc[df_final['mentor_career'].str.contains('mpras'), 'mentor_career'] = 'purchase/sales'
df_final.loc[df_final['mentor_career'].str.contains('yes'), 'mentor_career'] = 'law'
df_final.loc[df_final['mentor_career'].str.contains('vestiga'), 'mentor_career'] = 'research'
df_final.loc[df_final['mentor_career'].str.contains('esarrollo'), 'mentor_career'] = 'web development'
```

```
In [37]: #Translating values into english for mentor skills
df_final.loc[df_final['mentor_skills'].str.contains('onocimientos'), 'mentor_skills'] = 'career know-how'
df_final.loc[df_final['mentor_skills'].str.contains('écnica'), 'mentor_skills'] = 'IT skills'
df_final.loc[df_final['mentor_skills'].str.contains('escritura'), 'mentor_skills'] = 'writing skills'
df_final.loc[df_final['mentor_skills'].str.contains('lectura'), 'mentor_skills'] = 'reading skills'
df_final.loc[df_final['mentor_skills'].str.contains('nicacional'), 'mentor_skills'] = 'communication skills'
df_final.loc[df_final['mentor_skills'].str.contains('enseñar'), 'mentor_skills'] = 'teaching skills'
```

```
In [38]: #Translating values into english for session_freq
df_final.loc[df_final['session_freq'].str.contains('os veces al mes'), 'session_freq'] = 'Twice per month ( 15 - 30 minute session)'
df_final.loc[df_final['session_freq'].str.contains('na vez al mes'), 'session_freq'] = 'Once per month ( 30 - 60 minute session)'
df_final.loc[df_final['session_freq'].str.contains('lugar de se'), 'session_freq'] = 'only weekly email communication)'
df_final.loc[df_final['session_freq'].str.contains('semana'), 'session_freq'] = 'Once per week ( 15 - 30 minute session)'
```

```
In [39]: #Translating values into english for mentor age
df_final.loc[df_final['mentor_age'].str.contains('relevante'), 'mentor_age'] = 'Not relevant'
#Translating values into english for mentee_gender
df_final.loc[df_final['mentee_gender'].str.contains('Hombre'), 'mentee_gender'] = 'Male'
df_final.loc[df_final['mentee_gender'].str.contains('Mujer'), 'mentee_gender'] = 'Female'
df_final.loc[df_final['mentee_gender'].str.contains('Hombre transgénero'), 'mentee_gender'] = 'Transgender male'
df_final.loc[df_final['mentee_gender'].str.contains('Mujer transgénero'), 'mentee_gender'] = 'Transgender female'
#Translating values into english formentor_gender
df_final.loc[df_final['mentor_gender'].str.contains('Hombre'), 'mentor_gender'] = 'Male'
df_final.loc[df_final['mentor_gender'].str.contains('Mujer'), 'mentor_gender'] = 'Female'
df_final.loc[df_final['mentor_gender'].str.contains('Hombre transgénero'), 'mentor_gender'] = 'Transgender male'
df_final.loc[df_final['mentor_gender'].str.contains('Mujer transgénero'), 'mentor_gender'] = 'Transgender female'
df_final.loc[df_final['mentor_gender'].str.contains('relevante'), 'mentor_gender'] = 'Not relevant'
```

```
In [40]: #rename columns from the originally coliumn "mentor_role"
df_final.rename(columns={"itype4": "mentor_rol_1", "itype5": "mentor_rol_2"}, inplace=True)
df_final.head(6)
```

Out[40]:

	mentor_attribute	mentor_career	mentor_skills	session_freq	mentor_age	mentor_gender	mentee_age	mentee_gender	itype1
0	calmness	IT	career know-how	Twice per month (15 - 30 minute session)	Not relevant	Not relevant	25	Male	existencial
1	empathy	art	teaching skills	Twice per month (15 - 30 minute session)	23-29	Not relevant	22	Male	natural
2	calmness	design	communication skills	Twice per month (15 - 30 minute session)	30-36	Not relevant	26	Male	musical
3	calmness	IT	teaching skills	Once per month (30 - 60 minute session)	Not relevant	Not relevant	18	Male	musical
4	enthusiasm	IT	teaching skills	Once per week (15 - 30 minute session)	23-29	Not relevant	18	Male	existencial
5	modesty	IT	IT skills	Once per week (15 - 30 minute session)	23-29	Male	18	Male	natural

as the survey was anonym it was necessary to create a way of assigning a value to each user for the scope of the project. The first approach was to assign user ID's but considering the back-end development and the data it was being asked, for each user a fake identity was created (see Appendix 6, Faker Data)

```
In [41]: # add index df_final["user_id"] = df_final.index + 1
#insert new column and add index simultaneously
df_final.insert(0, 'user_id', df_final.index + 1)
```

```
In [42]: df_final.head()
```

Out[42]:

	user_id	mentor_attribute	mentor_career	mentor_skills	session_freq	mentor_age	mentor_gender	mentee_age	mentee_gender
0	1	calmness	IT	career know-how	Twice per month (15 - 30 minute session)	Not relevant	Not relevant	25	Male
1	2	empathy	art	teaching skills	Twice per month (15 - 30 minute session)	23-29	Not relevant	22	Male
2	3	calmness	design	communication skills	Twice per month (15 - 30 minute session)	30-36	Not relevant	26	Male
3	4	calmness	IT	teaching skills	Once per month (30 - 60 minute session)	Not relevant	Not relevant	18	Male
4	5	enthusiasm	IT	teaching skills	Once per week (15 - 30 minute session)	23-29	Not relevant	18	Male

```
In [43]: # this is the first dataset obtained from the 4 surveys and prepared for being merged with its mentor chilean pair.
df_final.to_csv('chile_students_cleaned.csv')
```