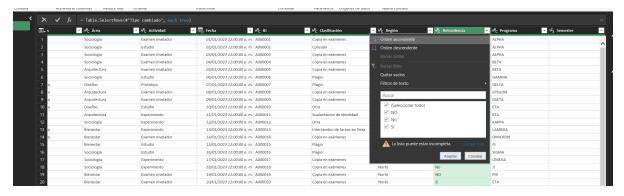
```
In [195...
```

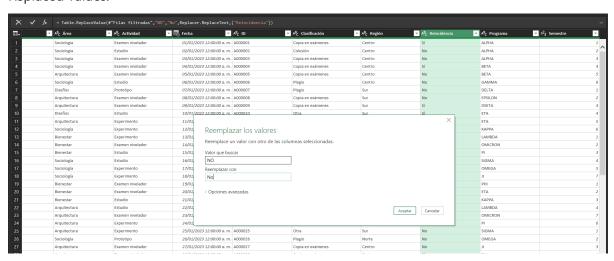
```
import dash
import dash_html_components as htmla
import base64
import numpy as np
import dash_core_components as dcc
from dash import html as html
from dash.dependencies import Input, Output
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
from jupyter_dash import JupyterDash
```

1. Verify the integrity of the data in Excel Power Query

There is an anomaly in the column "Reincidencia"



Replaced values:



2. Analize the data

This dataset shows reports from the Ethics Management Department

```
In [92]:
    df = pd.read_csv('datasets/Assessment_Data_Analyst_Cleaned.csv')
    df.head()
```

Out[92]:		Folio	Nivel de estudios	Campus	Área	Actividad	Fecha	ID	Clasificación	Región
	0	R0001	Kindergardiano	Narnia	Sociología	Examen nivelador	01/01/2023 00:00	A000001	Copia en exámenes	Centrc
	1	R0002	La Secu	Distrito 13	Sociología	Estudio	02/01/2023 00:00	A000002	Colusión	Centrc
	2	R0003	La Secu	Narnia	Sociología	Examen nivelador	03/01/2023 00:00	A000003	Copia en exámenes	Centrc
	3	R0004	La Secu	Narnia	Sociología	Examen nivelador	04/01/2023 00:00	A000004	Copia en exámenes	Centrc
	4	R0005	Big Bang Theory	Narnia	Arquitectura	Examen nivelador	05/01/2023 00:00	A000005	Copia en exámenes	Centrc

Delete null values

In [14]:

df.dropna()

$\cap \cdot \cdot +$	F 1 / I
17111	141
000	

	Folio	Nivel de estudios	Campus	Área	Actividad	Fecha	ID	Clasificación
0	R0001	Kindergardiano	Narnia	Sociología	Examen nivelador	01/01/2023 00:00	A000001	Copia en exámenes
1	R0002	La Secu	Distrito 13	Sociología	Estudio	02/01/2023 00:00	A000002	Colusión
2	R0003	La Secu	Narnia	Sociología	Examen nivelador	03/01/2023 00:00	A000003	Copia en exámenes
3	R0004	La Secu	Narnia	Sociología	Examen nivelador	04/01/2023 00:00	A000004	Copia en exámenes
4	R0005	Big Bang Theory	Narnia	Arquitectura	Examen nivelador	05/01/2023 00:00	A000005	Copia en exámenes
•••		•••						
1615	R1616	High School Musical	Pizza Planeta	Sociología	Experimento	23/09/2023 00:00	A000266	Copia en exámenes
1616	R1617	High School Musical	Pizza Planeta	Sociología	Experimento	24/09/2023 00:00	A000267	Copia en exámenes
1617	R1618	High School Musical	Pizza Planeta	Sociología	Experimento	25/09/2023 00:00	A000268	Copia en exámenes
1618	R1619	High School Musical	Pizza Planeta	Sociología	Experimento	26/09/2023 00:00	A000269	Copia en exámenes
1619	R1620	High School Musical	Pizza Planeta	Sociología	Prototipo	27/09/2023 00:00	A000270	Plagio

1620 rows × 12 columns

Drop not relevant columns

```
In [93]:
          df.drop(['Folio'],axis = 1, inplace=True)
In [108...
          df['Campus'].value counts()
          #daa=['Wonderland','Genovia','Springfield','Distrito 13','Pizza Planeta','Pan
           #'El País de Nunca Jamás', 'Narnia', 'Muy, muy lejano', 'Hogwarts', 'Parque Jurá
           #'Ciudad Gótica','Rarotonga','Empire']
                                    354
         Wonderland
Out[108...
         Genovia
                                    288
                                   156
         Springfield
         Distrito 13
                                   138
         Pizza Planeta
                                   132
         Pandora
                                   108
         El País de Nunca Jamás
                                   84
                                    78
         Narnia
         Muy, muy lejano
                                    78
         Hogwarts
                                    72
         Parque Jurásico
                                    48
         Ciudad Gótica
                                    42
                                    36
         Rarotonga
         Empire
         Name: Campus, dtype: int64
         Add a query for drop cases without Recidivism
In [95]:
          options = ['Sí']
          # selecting rows based on condition
```

```
df s = df.loc[df['Reincidencia'].isin(options)]
df s.head()
```

Out[95]:]: Nivel de estudios		Campus Área		Actividad	Fecha	ID	Clasificación	Región	Re
-	0	Kindergardiano	Narnia	Sociología	Examen nivelador	01/01/2023 00:00	A000001	Copia en exámenes	Centro	
	3	La Secu	Narnia	Sociología	Examen nivelador	04/01/2023 00:00	A000004	Copia en exámenes	Centro	
8		Kindergardiano	Pizza Planeta	Arquitectura	Examen nivelador	09/01/2023 00:00	A000009	Copia en exámenes	Sur	
	9	Big Bang Theory	Pizza Planeta	Diseños	Estudio	10/01/2023 00:00	A000010	Otra	Sur	
	11	La Secu	Pandora	Sociología	Experimento	12/01/2023 00:00	A000012	Otra	Norte	

Graph1 Recidivism by "Campus"

Graph2 Recidivism by "Nivel de estudios"

```
In [117...
     da=df['Nivel de estudios'].value_counts()
     fig2 = px.bar(da, y='Nivel de estudios')
     fig2.show()
```

Create new dataframe to filter data

```
In [156...
    index = ['Sur', 'Centro', 'Norte']
    dba = df.query('Región == "Sur"')
    dbb = df.query('Región == "Centro"')
    dbc = df.query('Región == "Norte"')
    dba = dba['Actividad'].value_counts()
    dbb = dbb['Actividad'].value_counts()
    dbc = dbc['Actividad'].value_counts()
    dbd = [dba,dbb,dbc]
    ud = np.array(dbd)
    dbe = pd.DataFrame(ud,index=index)
    dbe.columns = ['Experimento','Examen regulador','Prototipo','Examen nivelador dbe
```

Out[156...

	Experimento	Examen regulador	Prototipo	Examen nivelador	Examen diagnóstico	Estudio	
Sur	102	54	42	30	24	24	
Centro	186	168	156	150	96	48	
Norte	144	138	84	72	66	36	

Graph3 Recidivism by "Actividad" compared with "Región"

```
In [229...
          external stylesheets = ['https://codepen.io/chriddyp/pen/bWLwgP.css']
          app = JupyterDash( name , external stylesheets=external stylesheets)
          # assume you have a "long-form" data frame
          # see https://plotly.com/python/px-arguments/ for more options
          df bar = pd.DataFrame({
              "Fruit": ["Apples", "Oranges", "Bananas", "Apples", "Oranges", "Bananas"]
              "Amount": [4, 1, 2, 2, 4, 5],
              "City": ["SF", "SF", "SF", "Montreal", "Montreal", "Montreal"]
          })
          #fig = px.bar(df bar, x="Fruit", y="Amount", color="City", barmode="group")
          app.layout = html.Div(children=[
              # All elements from the top of the page
              html.Div([
                  html.Div([
                      html.H2(children='Ethics Management Department: Recidivism of AD(
                      html.Div(children='''
                          Recidivism by Activity
                      '''),
                      dcc.Graph (
                           id='graph1',
                           figure=fig
                      ),
                  ], className='six columns'),
                  html.Div([
                      html.Img(src=app.get asset url('logo.png')),
                      html.Div(children='''
                          Recidivism by study level.
                      111),
                      dcc.Graph (
                          id='graph2',
                           figure=fig2
                  ], className='six columns'),
              ], className='row'),
              # New Div for all elements in the new 'row' of the page
              html.Div([
                  html.H1(children=' '),
                  html.Div(children='''
                      Recidivism by Activity compared with Región.
                  111),
                  html.Div(children='''
                      Blue= Estudio||
                      Red= Examen diagnóstico||
                      Green= Examen nivelador||
                      Purple=Prototipo||
                      Orange = Examen regulador||
                      Light blue = Experimento||
                  '''),
```

```
dcc.Graph(
               id='graph3',
               figure=fig3
          ),
     ], className='row'),
])
if __name__ == '__main__':
    app.run_server(debug=True,port=8051)
```

Dash app running on http://127.0.0.1:8051/

```
In [ ]:
```