

# Guía para la evaluación de la herramienta:

## Pandalyze

- Esta herramienta fue realizada por estudiantes de la **Facultad de Informática** como trabajo de finalización de carrera. La herramienta, que está orientada a la introducción al **análisis de datos**, te permitirá cargar archivos CSV, visualizar datos, realizar análisis y gráficos básicos.
- A continuación, te daremos orientaciones y ejercicios para que puedas evaluar la funcionalidad y usabilidad de la herramienta.
- Realizá una copia de este documento y respondé las consignas. Subí el documento final a la tarea correspondiente del Classroom.
- Accedé a esta url: <https://668fdd58badbd2000869cde0--pandalyze.netlify.app/> y realizá los siguientes puntos:

### Paso 1: Inicio de uso de la herramienta

1. **Iniciar el Tutorial:** Accedé a través del navegador a la herramienta y seguí el tutorial inicial para familiarizarte con las funciones y la interfaz.
2. **Bloques de Ejemplo:** Al ocultar el tutorial, la herramienta mostrará bloques de ejemplo. Estos bloques están diseñados para que entiendas cómo se puede utilizar la herramienta para mostrar datos.
  - a. **Ejecutá el código**, observá la salida de los bloques de ejemplo. Identificá y respondé, debajo de cada pregunta:
    - i. ¿Qué tipo de información se muestra?

La información que se muestra son las primeras 10 líneas que se lee del dataset insertado.
    - ii. ¿Qué dataset se está utilizando?

El dataset utilizado es lagos.csv.
    - iii. ¿Qué columna del dataset se está utilizando para filtrar datos?

Las columnas que se usan son: Nombre, Ubicación, Superficie (km<sup>2</sup>), Profundidad máxima (m), Profundidad media (m), Coordenadas, Sup Tamaño, Latitud en GMS, Longitud en GMS, lat e lng
    - iv. ¿Cuál es el criterio utilizado para filtrar los datos?

El criterio utilizado es que solo se muestren las primeras 10 líneas del dataset.

- v. El primer print ¿qué tipo de variable utiliza?

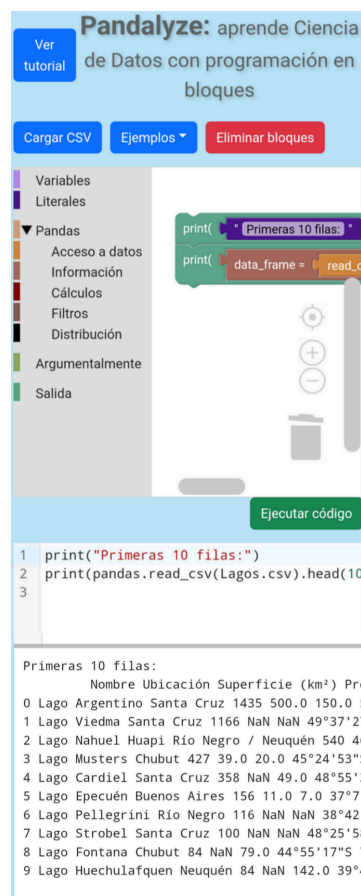
En el primer print se puede observar un bloque sacado de literales, el cual solamente va a mostrar un texto una vez se ejecute el código.

- b. ¿En cuántos sectores está dividida la interfaz de la herramienta?

Identificá y anotá cuántas secciones o partes distintas hay (por ejemplo, sección de bloques...).

Hay 4 secciones: sección de bloques, sección para unir los bloques elegidos, una para ver el código escrito y otra para ejecutarlo.

- i. Copiá debajo una captura de la interfaz remarcando para qué se utiliza cada sección y/o ícono.



Podemos observar en la parte superior izquierda, un tutorial de la aplicación para que el manejo de este se facilite al usuario.

Abajo del tutorial se contemplan tres opciones:

- Cargar csv: para subir el archivo que el usuario desee.
- Ejemplos: te arma un ejemplo de código.
- Eliminar bloques: te elimina absolutamente todos los bloques que estén en el área de trabajo.

Para ajustar la pantalla a nuestro gusto tenemos el + y el - para agrandar o achicar la pantalla y, abajo de este hay un icono de basura para eliminar los bloques no deseados.

Respecto a su uso tenemos una sección de bloques para armar código, solo se necesita arrastrar al área de trabajo lo que se necesite y luego unir los bloques.

Una vez hecho el código podemos concluirlo en la parte donde dice **ejecutar** y veremos el resultado en la consola debajo de todo.

- ii. Observa las categorías de bloques: ¿qué funcionalidad está incluida en cada categoría?

En la categorías de bloques podemos encontrar:

- Variables: su función es la creación de variables)
- Literales: podemos insertar algún texto, número u opción de falso/verdadero.
- Pandas: opciones para incrustar el dataset que se elija y trabajar sobre él.
- Argumentalmente
- Salida: encontramos bloques de salidas.

- iii. **Eliminar bloques innecesarios:** Si no necesitas los bloques de ejemplo para los siguientes ejercicios, puedes eliminarlos. ¿Cómo se puede borrar un bloque?

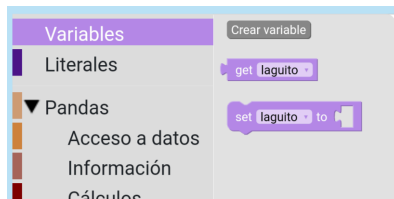
Hay dos opciones, la primera es hacer click en dónde dice “eliminar bloques” y la segunda es arrastrar los bloques innecesarios en el icono de la basura en la parte inferior derecha de la pantalla.

### Paso 3: Ejercicios introductorios de análisis exploratorio

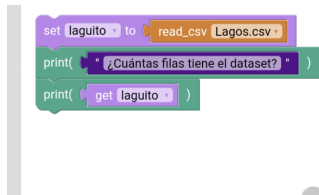
1. Utilizá el archivo de lagos que ya está cargado, sino probá cargarlo descargando el archivo desde <https://archivos.linti.unlp.edu.ar/index.php/s/LBNPS7EdmNSwt8z>.
2. **Conociendo las secciones de los bloques** (recordá ir eliminando los bloques que no son necesarios para cada inciso):
  - a. Encontrá el bloque para “leer” un archivo CSV. Indica cuál es la subsección donde se encuentra el bloque.

Se encuentra en la subsección de pandas con el nombre “acceso a datos”

- b. Definí una variable para guardar el contenido del dataset “lagos”. Agregá acá una captura de pantalla del bloque que permite **definir** una variable con tu variable ya definida.



- c. Utilizando el bloque para **acceder** a la variable, imprimí el contenido del dataset lagos utilizando el bloque “**print**”. ¿Cuántas filas tiene el dataset? Agregá acá una captura de pantalla de los bloques utilizados.



- d. Imprimí la salida de bloque info del dataset. **Copía** acá cuáles son las columnas que tienen datos faltantes.



También lo hice con la sentencia `isnull().sum()`



- e. Recorré la sección “**Información**”: Encontrá el bloque que permite conocer los nombres de las columnas del dataset, imprimí dicha información. **Copía** acá el **código** de Python correspondiente.

`print("¿Cuántas columnas tiene el dataset?")`

```
print(pandas.read_csv(Lagos.csv).columns)
```

- f. Copiá acá los valores únicos de la columna “Ubicación”.

```
print(
    column = data_frame = read_csv(Lagos.csv)
    Ubicación
    .unique()
)
```

Sus valores únicos son:

Santa Cruz, Río Negro / Neuquén, Chubut, Buenos Aires, Río Negro,  
Neuquén, Tierra del Fuego, Antártida e Islas del Atlántico Sur

## Paso 4: Ejercicios de cálculo y filtrado de datos

1. Imprimí el valor mínimo, máximo y promedio de la columna Superficie. **Tip:** podés investigar la sección “Cálculos”.

```
set lagos to read_csv Lagos.csv
print(
    Su valor maximo es:
)
print(
    data_frame = get lagos Superficie (km²)
    .max()
)
print(
    Su valor mínimo es:
)
print(
    data_frame = get lagos Superficie (km²)
    .min()
)
print(
    Su promedio es:
)
print(
    data_frame = get lagos Superficie (km²)
    .mean()
)
```

```
1 lagos = pandas.read_csv(Lagos.csv)
2 print("Su valor maximo es:")
3 print(lagos["Superficie (km²)"].max())
4 print("Su valor mínimo es:")
5 print(lagos["Superficie (km²)"].min())
6 print("Su promedio es:")
7 print(lagos["Superficie (km²)"].mean())
8
```

```
Su valor máximo es:
1435
Su valor mínimo es:
10
Su promedio es:
111.1923076923077
```

2. Utilizá el bloque “value\_counts()” para obtener la cantidad de lagos que hay por provincia. Dentro del bloque tenés que poner el bloque con la columna que contiene

los nombres de las provincias donde están los lagos. Imprimí y **copiá acá** la información mostrada. **Tip:** podés investigar la sección “Cálculos”

```
set laguito to read_csv Lagos.csv
print( data_frame = get laguito Ubicación .value_counts() )

1 laguito = pandas.read_csv(Lagos.csv)
2 print(laguito ["Ubicación"].value_counts)
3
```

```
Ubicación
Santa Cruz 16
Neuquén 14
Chubut 13
Río Negro 4
Tierra del Fuego, Antártida e Islas del Atlántico
Río Negro / Neuquén 1
Buenos Aires 1
Nombre: count, tipo: int64
```

La cifra de números al lado de las provincias son la cantidad de lagos

- Investigando la sección de “**Filtros**” y consultando los ejemplos precargados, imprimí los lagos que se encuentran en la Provincia de Neuquén. **Tip:** los bloques de la sección “Literal” pueden ayudarte.

```
set laguito to read_csv Lagos.csv
print( data_frame = get laguito [ columna = data_frame = get laguito Ubicación == Chubut ] )
```

```

1 laguito = pandas.read_csv(Lagos.csv)
2 print(laguito [laguito ["Ubicación"] ==
3

```

|    | Nombre            | Ubicación | Superficie (km²)          | Profun |
|----|-------------------|-----------|---------------------------|--------|
| 3  | Lago Musters      | Chubut    | 427 39.0 20.0 45°24'53"S  | 69     |
| 8  | Lago Fontana      | Chubut    | 84 NaN 79.0 44°55'17"S    | 71°2   |
| 11 | Lago La Plata     | Chubut    | 73 NaN NaN 44°51'33"S     | 71°    |
| 13 | Lago Menéndez     | Chubut    | 58 287.0 149.0 42°41'28"S |        |
| 18 | Lago Puelo        | Chubut    | 47 180.0 111.0 42°9'3"S   | 71°3   |
| 21 | Lago Futalaufquen | Chubut    | 42 168.0 101.0 42°49'     |        |
| 30 | Lago Blanco       | Chubut    | 28 NaN NaN 45°53'16"S     | 71°13  |
| 33 | Lago Rivadavia    | Chubut    | 23 147.0 104.0 42°36'39"  |        |
| 36 | Lago Río Chico    | Chubut    | 20 NaN NaN 45°35'40"S     | 68     |
| 38 | Lago Epuyén       | Chubut    | 17 148.0 92.0 42°11'15"S  | 71     |
| 40 | Lago Cholila      | Chubut    | 17 108.0 49.0 42°27'31"S  | 7      |
| 49 | Lago Pico N° 1    | Chubut    | 10 NaN NaN 44°14'31"S     | 71     |
| 51 | Lago Rosario      | Chubut    | 10 57.0 25.0 43°15'36"S   | 71     |

## Paso 5: Realización de gráficos

1. En base a los bloques que se muestran abajo, realizar un gráfico de torta con los mismos datos en la variable "prov".

```

set lagos to read_csv(Lagos.csv)
set prov to data_frame = get(lagos, Ubicación).value_counts()
plotly.bar(data_frame = get(prov), x = get(prov).values, y = get(prov).index, title = "Lagos por provincia", show())

```

Gráfico:

