



ISIS-1221

INTRODUCCIÓN A LA PROGRAMACIÓN

Nivel 4 – Proyecto Acceso a Internet en Colombia

Objetivo general

El objetivo de este proyecto es que usted practique los conceptos estudiados en el nivel 4 del curso. Para ello construirá una aplicación en donde analizará el acceso a internet en Colombia. Recuerde que este proyecto debe realizarse de forma **completamente individual**.

Objetivos específicos

1. Implementar algoritmos para construir y recorrer matrices.
2. Utilizar las librerías pandas y matplotlib, así como consultar los sitios web oficiales donde se encuentra la documentación de estas librerías.
3. Descomponer un problema en subproblemas e implementar las funciones que los resuelven.

Contexto

En Colombia, el acceso a Internet ha experimentado un notable crecimiento en los últimos años, con una mayor penetración tanto en áreas urbanas como rurales. El país ha realizado importantes inversiones en infraestructura de telecomunicaciones, lo que ha contribuido a una expansión significativa de la conectividad. Sin embargo, persisten desafíos en términos de cobertura y calidad del servicio en algunas regiones, especialmente en áreas remotas y menos desarrolladas. A pesar de ello, el acceso a Internet sigue siendo fundamental para el desarrollo económico, social y educativo del país, impulsando la inclusión digital y facilitando el acceso a información y oportunidades para un amplio espectro de la población.

En este proyecto usted trabajará con datos del acceso a internet en Colombia proporcionados en <https://www.datos.gov.co/Ciencia-Tecnolog-a-e-Innovaci-n/Internet-Fijo-Penetraci-n-Departamentos/4py7-br84/data> por el gobierno de Colombia. Sin embargo, se trabajará con una versión simplificada y filtrada de los datos originales: suprimimos algunas entradas incompletas y columnas que no vamos a utilizar en este proyecto, además, añadimos una columna de “Provincias”. Con este conjunto de datos usted podrá analizar información esencial de los accesos fijos a Internet en Colombia.

Su aplicación debe tener las siguientes partes:

Parte 1 | Leer la información del archivo

Requerimiento 0: Cargar datos

En primer lugar, usted debe cargar los datos de un archivo a un DataFrame de pandas. Para ello, le debe preguntar al usuario el nombre del archivo csv a cargar. Esto quiere decir que **la función que implemente este requerimiento debe recibir como parámetro el nombre del archivo y debe retornar un DataFrame.**

El archivo que se debe poder cargar es “internet.csv”. A continuación, se presentan las columnas del archivo con sus respectivos significados:

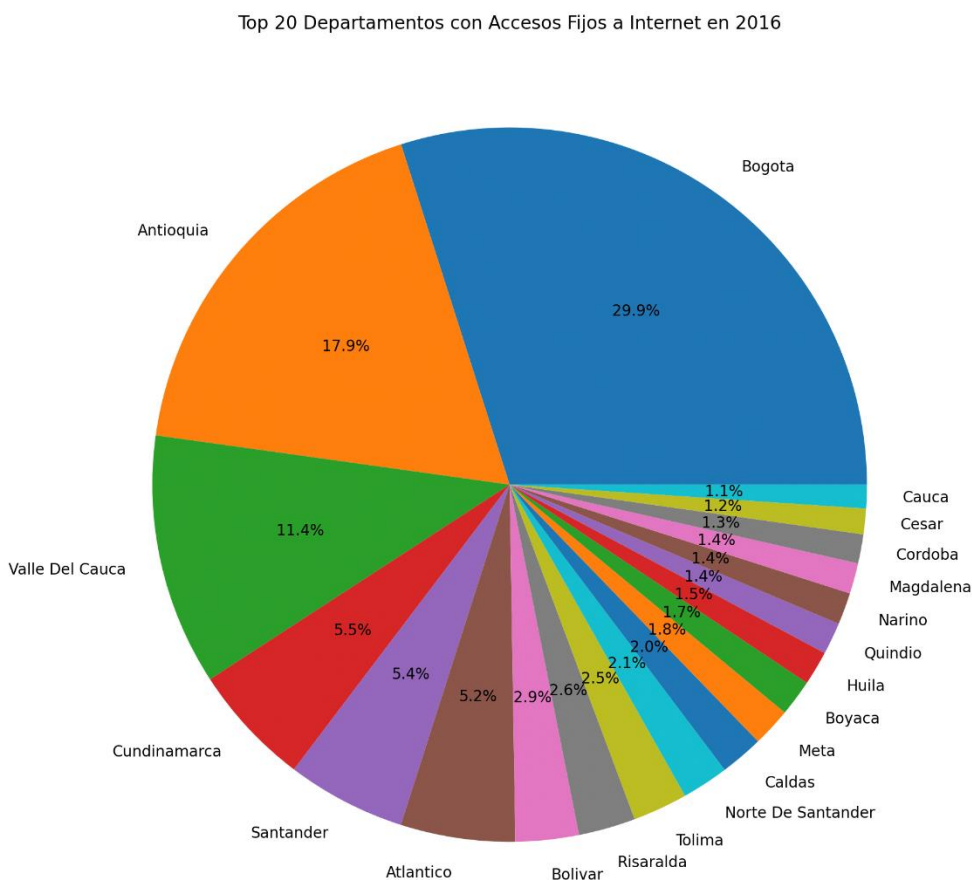
- **AÑO:** Esta columna indica el año al que corresponde el registro en el conjunto de datos.
- **DEPARTAMENTO:** En esta columna se registra el nombre del departamento de Colombia al que pertenece el municipio mencionado en la siguiente columna.
- **PROVINCIA:** En esta columna se especifica la provincia del departamento al que pertenece el municipio al que corresponde esta columna. Sabemos que estas subregiones departamentales no son muy mencionadas, si desea entender más sobre ellas puede consultar el enlace https://es.wikipedia.org/wiki/Subregiones_de_Colombia
- **MUNICIPIO:** Aquí se especifica el nombre del municipio del que se está proporcionando información sobre el acceso a Internet.
- **No. ACCESOS FIJOS:** Esta columna muestra el número de accesos fijos a Internet registrados en el municipio correspondiente al año indicado.
- **POBLACIÓN DANE:** Aquí se registra la población del municipio según los datos proporcionados por el Departamento Administrativo Nacional de Estadística (DANE) de Colombia en ese año.

ATENCIÓN: cuando esté estudiando el problema y el archivo, recuerde que las funciones **describe()**, **unique()** y **filter()** pueden serle de utilidad. La función **describe()** aplicada sobre un DataFrame retorna información estadística de todas las columnas numéricas. La función **unique()** aplicada sobre una columna retorna una lista con los valores únicos que aparezcan en esa columna.

Parte 2 | Análisis de la distribución de los datos

Requerimiento 1: Pie chart del Top 20 departamentos con mayor número de accesos fijos a internet en un año en específico

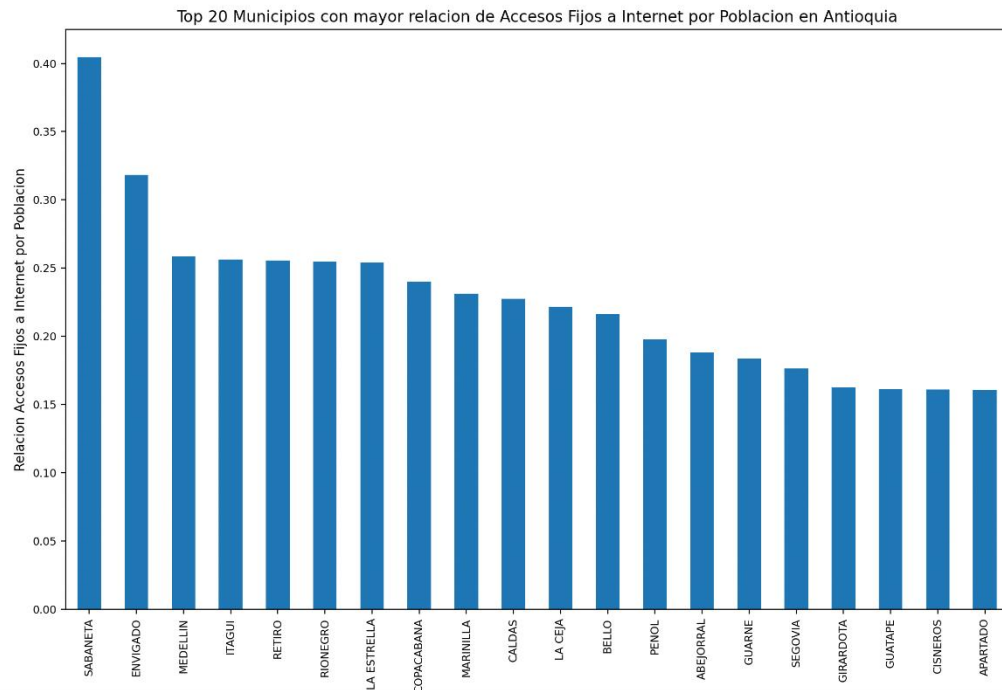
Para este requerimiento se quiere conocer el Top 20 de departamentos con mayor número de accesos fijos a internet en un año. Para esto, el usuario debe indicar el año que desea consultar. La siguiente figura muestra la apariencia del pie chart esperado:



ATENCIÓN: Le recomendamos utilizar el parámetro `figsize = (10, 8)` para que la figura quede con las mismas proporciones. Además, también puede utilizar `plt.tight_layout()` para ajustar los márgenes alrededor de su gráfica. Recuerde ponerle título a sus ejes y a la gráfica.

Requerimiento 2: Gráfica de Barras del Top 20 municipios con mayor ratio (*No. ACCESOS FIJOS/POBLACION DANE*) en un departamento en específico.

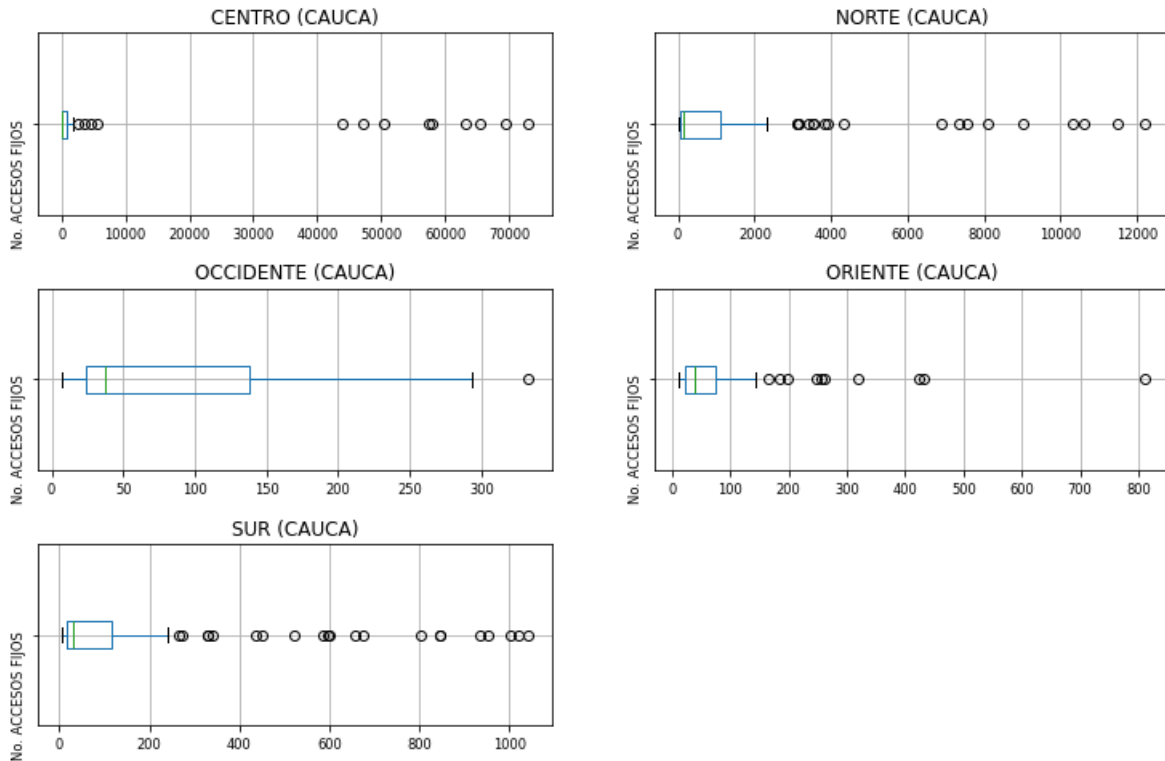
En este requerimiento se desea conocer el Top 20 de municipios con el mayor ratio *No. ACCESOS FIJOS/POBLACION DANE* en un departamento en específico indicado por el usuario. Para calcular este ratio, debe sumar todos los accesos fijos de un municipio durante todos los años disponibles y dividirlo sobre la suma de todos los habitantes de dicho municipio durante todos los años disponibles. Este ratio se debe mostrar en una gráfica de barras como se muestra a continuación:



ATENCIÓN: Le recomendamos utilizar el parámetro `figsize = (13, 7)` para que la figura quede con las mismas proporciones. Además, también puede utilizar el argumento `fontsize= "small"` para ajustar el tamaño de la letra. Recuerde ponerle título a los ejes y a la gráfica.

Requerimiento 3: Diagrama de caja y bigotes de accesos fijos por provincias de un departamento.

En este requerimiento se desea realizar un diagrama de caja y bigotes que muestre la distribución de accesos fijos a internet por provincia en un departamento en específico indicado por el usuario. El diagrama debe verse como el que se muestra a continuación:



ATENCIÓN: Le recomendamos utilizar el parámetro `figsize = (14, 8)` para que la figura quede con las mismas proporciones que la mostrada. Además, también puede utilizar el argumento `fontsize="small"` para ajustar el tamaño de la letra. En caso de que en su diagrama se intercepten los títulos o no se vean con claridad, puede utilizar la función `plt.subplots_adjust(hspace=0.4, left=0.2)`. Por último, recuerde ponerle título a su diagrama.

NOTA IMPORTANTE: Para que los boxplots tengan escala diferente y no estén obligados a usar las mismas magnitudes, utilice el argumento `sharey=False`. Es muy importante usar este argumento, pues no utilizarlo puede hacer que su código falle.

Parte 3 | Estudiar estadísticas del número de accesos fijos a Internet

En esta última parte no se trabajará directamente sobre el DataFrame original, sino que se creará una matriz a partir de este. La matriz debe relacionar el **número de accesos fijos a internet por año** y los **departamentos del territorio nacional**.

Requerimiento 4: Construcción de la matriz de Departamento vs Año

Este requerimiento consiste en construir una matriz que cruce el departamento vs el número de accesos fijos a internet por año. La matriz tiene la siguiente estructura:

	2015	2016	2017	...	2021	2022
Amazonas	527	459	408		1005	941
Antioquia	957544	1033038	1098914	...	1645915	1615103
...
Vaupés	69	67	65		105	84
Vichada	135	766	2217	...	1057	813

ATENCIÓN: La matriz mostrada anteriormente es un ejemplo y NO hace referencia a los valores que usted obtendrá al construir la matriz. Cada posición (f,c) en la matriz contiene el número de accesos fijos a internet en el año en la posición c. Por ejemplo, si Amazonas está en la fila 1 y 2017 en la columna 2, el número de accesos fijos a internet en Amazonas en el año 2017 sería el valor en la casilla (1,3).

Así mismo, se propone la construcción de 2 diccionarios para la referencia de las filas y las columnas. En el primer diccionario cada llave hace referencia al índice de columna en la matriz y el valor hace referencia año correspondiente en la matriz mostrada anteriormente. El segundo diccionario tendrá como llave el índice de la fila en la matriz y como valor el nombre del departamento correspondiente en la matriz. **En el esqueleto, encontrará el fragmento de código que le permitirá construir estos diccionarios.**

A continuación, se muestran los diccionarios para las filas y las columnas:

Llave	Valor
0	2015
1	2016
...	...
7	2021
8	2022

Llave	Valor
0	"Amazonas"
1	"Antioquia"
...	...
31	"Vaupés"
32	"Vichada"

ATENCIÓN:

- Recuerde que tiene a su disposición la función **unique()** para obtener todos los valores únicos de una columna en un DataFrame.
- Para comprobar que los valores de la matriz creada son los correctos puede utilizar filtros, agrupaciones y sumas sobre el DataFrame original y verificar que estos arrojen el mismo resultado que el almacenado en su matriz.
- Dado que el tamaño del DataFrame es bastante grande, la construcción de la matriz podría tardar más tiempo del que suelen tardar otras funciones. Si el tiempo supera los 2 minutos es muy probable que haya un problema en el código. En ese caso, le recomendamos que detenga el programa y revise si existe algún error en sus funciones.

La función para crear la matriz debe retornar una tupla que contenga la matriz creada y los diccionarios de filas y columnas en el siguiente formato (**matriz, dict_columnas, dict_filas**). Tenga en cuenta que usted deberá implementar los siguientes requerimientos utilizando como base la tupla anterior.

Requerimiento 5: Encontrar la cantidad de accesos fijos a Internet en un año

En este requerimiento se debe contar la cantidad total de accesos fijos a Internet en un año específico. Para esto, el usuario debe indicar el año en el cual se desea consultar la información. La función que implemente esta opción debe retornar el valor que representa la cantidad de accesos fijos totales en el año dado por el usuario. En la consola se debe mostrar un mensaje de la forma: "La cantidad de accesos fijos a internet en el año X es de Y" donde X es el año indicado por el usuario y Y el número total de accesos fijos en ese año en todos los departamentos de Colombia.

Requerimiento 6: Encontrar el primer departamento alfabéticamente, cuyo crecimiento frente a un año específico sea mayor a un porcentaje específico

En este requerimiento se encontrar el primer departamento, en orden alfabético, cuyo cambio porcentual en accesos a internet con respecto al año directamente siguiente, sea mayor que el porcentaje especificado. Usted recibirá por parámetro un año desde el cual se quiere hacer el cálculo de crecimiento de accesos a internet, y un porcentaje el cual se quiere comparar con todos los cambios a nivel porcentual frente al año dado por parámetro y el año siguiente. La función que implemente esta opción debe retornar una tupla cuyo primer valor será el nombre del departamento y el segundo valor será el porcentaje que supera al recibido por parámetro. En caso que no haya ningún departamento que supere este porcentaje, los valores en la tupla de respuesta deberán ser "Ninguno" y -101 respectivamente. En la consola se debe mostrar un mensaje de la forma: "X tuvo un crecimiento del Y% del año A al año B", donde X es el nombre del primer departamento encontrado, Y el porcentaje de crecimiento, A es el año a consultar y B es el año siguiente. Si no se encontró ninguno, basta con un mensaje que lo informe. Se debe redondear el porcentaje obtenido a 2 decimales.

ATENCIÓN: Tenga en cuenta que para el último año para el que se tienen registros **NO** es posible calcular el aumento o disminución respecto al año siguiente. En caso de que se reciba este año (2023) retorne lo mismo como si no se hubiera encontrado ningún departamento. Además, la diferencia entre año y año puede ser negativa (en caso de haber un decrecimiento).

Requerimiento 7 (BONO): Mapa de Accesos fijos a Internet

NOTA IMPORTANTE: El siguiente requerimiento es un BONO, si usted lo desea puede implementarlo. Decidir no hacerlo NO penalizará su nota de ninguna manera.

Este último requerimiento de la aplicación consiste en mostrar el número de accesos a Internet para cada departamento en un año en específico. El usuario debe suministrar la matriz de departamentos vs año y el año de su interés.

Una vez obtenidos los accesos fijos a internet en el año especificado para cada departamento, se debe mostrar la información sobre un mapa de Colombia. Para esta visualización utilizaremos el archivo **mapa.png** (que se encuentra adjunto a este enunciado), el cual es una imagen del mapa de Colombia en blanco y negro con resolución de 800x800 píxeles. La idea es indicar dentro del mapa la cantidad de accesos fijos a internet en cada departamento. Para cada uno de estos Departamentos, se pintará un pequeño cuadrado de 13x13 píxeles centrado en las coordenadas (en píxeles) del Departamento dentro el mapa. Las coordenadas de los 33 departamentos se encuentran en el archivo **coordenadas.txt** (que se encuentra adjunto a este enunciado).

Para cargar el mapa como una matriz de píxeles y luego visualizarlo puede guiarse por el siguiente fragmento de código:

```
import matplotlib.image as mpimg
mapa = mpimg.imread("mapa.png").tolist()
plt.imshow(mapa)
plt.show()
```

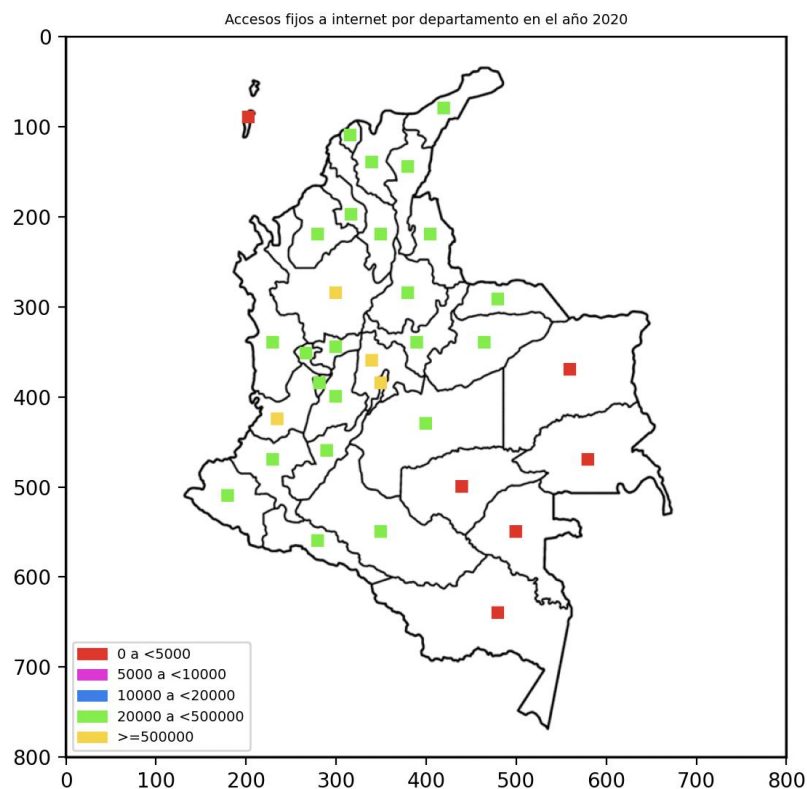
La siguiente función le permitirá cargar el archivo de coordenadas y retornar un diccionario, cuyas llaves son los nombres de los departamentos y los valores son tuplas con las coordenadas (x,y) de cada departamento. El nombre del departamento en este archivo es idéntico al que se encuentra en el DataFrame y en la matriz.

```
def cargar_coordenadas(nombre_archivo:str)->dict:
    deptos = {}
    archivo = open(nombre_archivo, encoding="utf8")
    titulos = archivo.readline()
    linea = archivo.readline()
    while len(linea) > 0:
        linea = linea.strip()
        datos = linea.split(";")
        deptos[datos[0]] = (int(datos[1]),int(datos[2]))
        linea = archivo.readline()
    return deptos
```

Una vez cargado el mapa y construido el diccionario de coordenadas, el siguiente paso es utilizar dichas coordenadas para pintar en el mapa varios puntos con un color que represente la cantidad de accesos fijos a internet. Para ello, se han asignado los siguientes colores a cada grupo como se muestra a continuación:

Rango	Valores RGB	Color
0 a <5000	[0.94, 0.10, 0.10]	Rojo
5000 a <10000	[0.94, 0.10, 0.85]	Magenta
10000 a <20000	[0.10, 0.50, 0.94]	Azul
20000 a <50000	[0.34, 0.94, 0.10]	Verde
>50000	[0.99, 0.82, 0.09]	Amarillo

Cada uno de los cuadrados en el mapa debe utilizar alguno de estos colores, y se debe respetar el orden de la clasificación. La siguiente figura muestra un ejemplo del mapa para el año 2020:



ATENCIÓN: Para poder crear las leyendas de la gráfica anterior se debe utilizar el paquete `mpatches` de la librería de `matplotlib`. Para poder añadir las leyendas, haga uso del siguiente código:

```

colores = {"0 a <5000":[0.94, 0.10, 0.10],
           "5000 a <10000":[0.94, 0.10, 0.85],
           "10000 a <20000":[0.10, 0.50, 0.94],
           "20000 a <500000":[0.34, 0.94, 0.10],
           ">=500000":[0.99, 0.82, 0.09]}

plt.imshow(mapa)

legends = []
for i in colores:
    legends.append(mpatches.Patch(color = colores[i], label=i))
plt.legend(handles = legends, loc = 3, fontsize='x-small')
plt.title("Accesos fijos a internet por departamento en el año" + str(anio),
          fontsize='x-small')
plt.show()

```

Preparación del ambiente de trabajo

1. Cree una carpeta para trabajar, poniéndole su nombre o login.
2. Descargue de Bloque Neón el archivo **internet.zip** que contiene el archivo **internet.csv** con los datos a procesar, los archivos **internet.py** y **consola_internet.py** que debe usar como esqueleto para realizar el proyecto, y una carpeta **bono** con los archivos relevantes si usted desea implementar el bono. El archivo **internet.py** contiene el código para crear los diccionarios de las filas y columnas de la matriz que usted debe construir.
3. Abra Spyder y cambie la carpeta de trabajo para que sea la carpeta donde descargó el archivo con los datos.

Construcción del módulo de funciones

4. Usando Spyder, complete el archivo **internet.py**. En este archivo usted va a construir el módulo en el que va a implementar las funciones que responden a los requerimientos de la aplicación. **Defina, documente e implemente** las funciones en su nuevo archivo. Usted puede crear cuántas funciones considere necesarias dentro de su librería o módulo. Mínimo debe haber una función por cada uno de los requerimientos del programa.

Construir la interfaz de usuario basada en consola

5. En esta actividad usted tiene que construir la interfaz basada en consola para que el usuario interactúe con la aplicación. Para construir esta interfaz usted debe completar el archivo **consola_internet.py**, la cual ya tiene una parte implementada que le facilitará su trabajo. Usted debe modificar los elementos marcados con la etiqueta **TODO**.
6. Pruebe la interfaz basada en consola ejecutando el archivo **consola_internet.py**. Verifique que las funcionalidades de su aplicación se comporten de acuerdo con lo esperado.

Probar el correcto funcionamiento de la aplicación

7. **Ejecute el programa y pruebe** cada una de las funciones para asegurarse que esté funcionando. Puede probar el correcto funcionamiento de su programa cargando la información que se encuentra en el archivo **internet.csv** o creando su propio archivo de prueba de menor tamaño (respetando el mismo formato) que le permita corroborar que los resultados arrojados por su programa son correctos.

Entrega

9. Comprima los dos archivos: **internet.py** y **consola_internet.py** en un solo archivo .zip. El archivo comprimido debe llamarse **N4-PROY-login.zip**, donde login es su nombre de usuario de Uniandes (omite el punto del login para evitar posibles problemas con la extensión de los archivos. Por ejemplo, si su login fuese **p.perez123**, nombre el archivo como **N4-PROY-pperez123.zip**).
10. Entregue el archivo comprimido a través de Bloque Neón en la actividad designada como **Proyecto de Nivel 4**.