

# DESARROLLO DE APLICACIONES PARA ANALISIS DE DATOS PRÁCTICA 3

## 1. (60 Puntos) Manipulación de datos con DataFrames

Ejercicio. Crear un script en Python llamado "DataFrames.py" que convierta el archivo de datos "emails.txt" en un Data Frame y realice lo siguiente:

- a) A cada fila asignarle una letra como nombre o índice. Ejemplo: primera fila = 'a', segunda fila = 'b', tercera fila = 'c', etcétera.
- b) Mostrar la información básica del df, incluido el número de filas, los nombres de las columnas, el número y el tipo de valores en cada columna.
- c) Mostrar solo las columnas animal y age con todas sus filas.
- d) Mostrar todas las columnas con únicamente las filas pares.
- e) Mostrar solo las columnas age y visits con únicamente las filas impares.
- f) Mostrar todas las columnas con las filas cuya edad sea mayor a 3.
- g) Mostrar todas las columnas con las filas que no contengan un dato para edad.
- h) Sustituir las celdas que no contengan un dato para edad con el promedio del resto de edades.
- i) Mostrar todas las columnas con las filas cuya edad este sea mayor a 2 y menor o igual 5.
- j) Obtener la suma de todas las visitas.
- k) Calcular el promedio de la edad de cada tipo diferente de animal. Tip: Investigar y usar el método "groupby".
- l) Desplegar el data frame ordenado por por dos criterios, primero por edad en orden descendente y después por visitas en orden ascendente. Tip: Investigar y usar el método "sort\_values".
- m) Reemplazar sí y no en la columna de prioridad con su valor booleano correspondiente.
- n) Reemplazar las celdas de la columna de animales por su respectivo nombre en español.

## 2. (60 Puntos) Manipulación de texto usando expresiones regulares

- a. Crear un script en python llamado "RegEx.py" que cree una lista a partir del archivo "expenses.txt". Cada línea del archivo debe corresponder a un elemento de la lista.

Incluir el siguiente código para desplegar líneas o patrones que coincidan con una expresión regular.

```
import re

# 1a
pat = r'D'

for line in records:
    if re.search(pat, line) != None:
        print(line)
```

Confirme que la salida son líneas que contienen el carácter "D" en alguna parte.

- b. Comente el patrón de a) y agregue una nueva línea tal como se muestra a continuación:

```
# a)
# pat = r'D'

# b)
pat = r'\'
```

Confirme que la salida son líneas que contienen el carácter ' (comilla simple) en alguna parte.

- c. Comente el patrón de b) y agregue una nueva línea que despliegue líneas que contengan " (comillas dobles) en alguna parte.

**Continúe comentando los patrones previos y agregando nuevas líneas para lo que se solicita en los siguientes incisos:**

- d. Agregue un patrón que despliegue líneas que comiencen con el número 7 (siete).
- e. Agregue un patrón que despliegue líneas que terminen con una "r" o una "t".
- f. Agregue un patrón que despliegue líneas que contengan un punto ".".
- g. Agregue un patrón que despliegue líneas que contengan una "r" seguida por una "g". (La "r" y la "g" no necesariamente tienen que estar en posiciones consecutivas).

- h. Agregue un patrón que despliegue líneas que contengan dos caracteres consecutivos en mayúscula, por ejemplo: AA, DF, LM, YW, ..., etcétera.
- i. Agregue un patrón que despliegue líneas que contengan una “,” (coma).
- j. Agregue un patrón que despliegue líneas que contengan tres o más comas. Las comas no necesariamente tienen que ser consecutivas.
- k. Agregue un patrón que despliegue líneas que NO contengan los caracteres v, w, x, y, o z.
- l. Agregue un patrón que despliegue líneas que contenga cantidades de dinero entre 0.00 and 99.99. Trate de hacer la expresión regular lo más compacta posible
- m. Agregue un patrón que despliegue líneas que contengan exactamente tres comas.
- n. Agregue un patrón que despliegue líneas que contengan el carácter “(“. Es decir, un paréntesis de apertura.
- o. Agregue un patrón que despliegue líneas que describen comidas que cuesten al menos 100.00.
- p. Agregue un patrón que despliegue líneas que tengan una categoría de gastos compuestas por exactamente cuatro caracteres (El patrón debe funcionar inclusive si más líneas son agregadas al archivo, con nuevas categorías que no han sido definidas.
- q. Agregue un patrón que despliegue líneas que para gastos ocurridos en marzo.
- r. Agregue un patrón que despliegue líneas que contengan una “a”, seguida por una “b”, seguida por una “c” (pueden haber otros caracteres la “a” la “b” y la “c”).
- s. Agregue un patrón que despliegue líneas que contengan secuencias de dos caracteres, seguidas por la MISMA secuencia de dos caracteres dos veces. Es decir, cada línea que coincide con la expresión regular debe contener al menos una secuencia de dos caracteres repetida al menos tres veces.
- t. Agregue un patrón que despliegue líneas que contengan en la descripción de gastos una “a” minúscula y un dígito entre 0 y 9 en cualquier orden. Es decir, el carácter “a” puede aparecer antes o después del dígito.
- u. Agregue un patrón que despliegue líneas que NO contengan letras mayúsculas.

- v. Agregue un patrón que despliegue líneas que contengan el carácter “d”, posiblemente seguido de otros caracteres, seguido de una “i”. Coincidencias incluirían palabras tales como: diver, doily, drip, diplomat, etc.