



Carátula para entrega de prácticas

Facultad de Ingeniería

Laboratorio de docencia

Laboratorios de computación salas A y B

Profesor: Jesús Cruz Navarro

Asignatura: Estructuras de Datos y Algoritmos 2

Grupo: 1

No. de Práctica(s): 10

Integrante(s): Ugalde Velasco Armando

*No. de Equipo de
cómputo empleado:*

No. de Lista o Brigada: 32

Semestre: 2021-1

Fecha de entrega: 10 de diciembre de 2020

Observaciones:

CALIFICACIÓN: _____

PRÁCTICA 10: ARCHIVOS

Objetivo: El estudiante conocerá e identificará aspectos sobre los archivos, como las operaciones, el tipo de acceso y organización lógica.

Desarrollar un programa que lea el archivo con formato **CSV** de casos confirmados de COVID-19 y procesar los datos para obtener y mostrar en la consola la siguiente información:

- Total de casos.
- Número de casos por estado.
- Estado con más casos.
- Estado con menos casos.
- Número de casos por género.
- Promedio de Edad.
- Edad máxima.
- Edad mínima.

Además, generar un archivo de nombre *'resumenCovid.eda2'* con estos mismos resultados, uno por renglón (bonito). El archivo debe crearse en el directorio *'PracticaResumen'* en el directorio de trabajo, usando rutas relativas. El programa debe crear dicho directorio si no existe.

Para cumplir con los requerimientos, se implementaron las funciones que se muestra a continuación:

Primero, se implementó la función **parseCsv**, que lee el archivo **csv** y retorna una tupla que contiene dos listas: la primera contiene los nombres de las columnas, y la segunda contiene listas que contienen los elementos presentes en cada fila. Nótese que se transforma el elemento presente en el índice **3** a un dato de tipo entero, ya que éste representa la edad y se utiliza posteriormente.

```
def parseCsv():
    with open("covid-19-mexico-01122020.csv", encoding='utf-8') as covidFile:
        lines = covidFile.read().splitlines()
        columnNames = lines[0].split(',')
        rows = []
        for i in range(1, len(lines)):
            entry = lines[i].split(',')
            rows.append((entry[0], entry[1], entry[2], int(entry[3]), entry[4], entry[5]))

    return columnNames, rows
```

Función parseCsv

Además, se implementó la función **createDir**, que crea el directorio requerido “*PracticaResumen*” en caso de no existir, y, de lo contrario, imprime un mensaje indicando que éste ya existe. Esta función se ejecuta al inicio del programa.

```
def createDir():
    # Crear directorio si no existe
    try:
        os.mkdir("PracticaResumen")
    except FileExistsError:
        print("El directorio PracticaResumen ya existe.")
```

Función createDir

De igual forma, se implementó la función auxiliar **printBothFactory**, que retorna otra función que imprime los argumentos recibidos en consola, y, además, los escribe en el archivo abierto proporcionado. Nótese que, una vez proporcionado el argumento **openedFile** en la función exterior, éste se “almacena” en el contexto léxico de la función generada.

```
def printBothFactory(openedFile: TextIO):
    def printBoth(*toPrint):
        formattedStr = " ".join([str(element) for element in toPrint])
        print(formattedStr)
        openedFile.write(formattedStr + "\n")
    return printBoth
```

Función printBothFactory

Finalmente, se implementó la función **analyzeData**, que obtiene los datos requeridos y, utilizando la función **printBoth** generada, los imprime a consola y los escribe en el archivo

resumenCovid.eda2. Para facilitar la tarea anterior, se decidió utilizar la librería **pandas**, ya que no se interfiere con el objetivo de la práctica.

```
def analyzeData():
    columns, entries = parseCsv()
    dataframe = pandas.DataFrame(entries, columns=columns)

    file = open("PracticaResumen/resumenCovid.eda2", "w")
    printBoth = printBothFactory(file)

    printBoth("\nTotal de casos:", len(dataframe))

    printBoth("\n          Casos por estado")
    casesPerState = dataframe["Nom_Ent"].value_counts()
    printBoth(casesPerState.to_string())

    printBoth("\nEstado con más casos:", casesPerState.idxmax())
    printBoth("Estado con menos casos:", casesPerState.idxmin())

    printBoth("\nCasos por género")
    casesPerGender = dataframe["SEX0"].value_counts()
    printBoth(casesPerGender.to_string())

    printBoth("\nPromedio de edad:", dataframe["EDAD"].mean())
    printBoth("\nEdad máxima:", dataframe["EDAD"].max())
    printBoth("Edad mínima:", dataframe["EDAD"].min())

    file.close()
```

Función analyzeData

Por último, en la función principal se crea el directorio correspondiente y se realiza en análisis del archivo:

```
if __name__ == '__main__':
    createDir()
    analyzeData()
```

Función driver

A continuación, se muestra la salida del programa:

```
El directorio PracticaResumen ya existe.
```

```
Total de casos: 1048575
```

```

                Casos por estado
Ciudad de México      223778
México                 73772
Nuevo León            63293
Guanajuato             59203
Sonora                 40984
Puebla                 39852
Jalisco                39027
Coahuila de Zaragoza  38987
Veracruz de Ignacio de la Llave 37975
Tabasco               36761
Tamaulipas            34430
San Luis Potosí       33017
Michoacán de Ocampo   27317
Chihuahua             26838
Baja California       25369
Sinaloa               23592
Yucatán               23098
Oaxaca                22986
Guerrero              22699
Querétaro             20084
Durango               19141
Hidalgo               17458
Zacatecas             15144
Quintana Roo          13960
Baja California Sur   13721
Aguascalientes        12820
Tlaxcala              7991
Chiapas              7268
```

Salida en consola

Baja California	25569
Sinaloa	23592
Yucatán	23098
Oaxaca	22986
Guerrero	22699
Querétaro	20084
Durango	19141
Hidalgo	17458
Zacatecas	15144
Quintana Roo	13960
Baja California Sur	13721
Aguascalientes	12820
Tlaxcala	7991
Chiapas	7268
Morelos	7171
Colima	7027
Nayarit	6927
Campeche	6885

Estado con más casos: Ciudad de México

Estado con menos casos: Campeche

Casos por género

M 535053

F 513522

Promedio de edad: 44.299566554609825

Edad máxima: 120

Edad mínima: 0

Salida en consola

2	Total de casos: 1048575	
3		
4	Casos por estado	
5	Ciudad de México	223778
6	México	73772
7	Nuevo León	63293
8	Guanajuato	59203
9	Sonora	40984
10	Puebla	39852
11	Jalisco	39027
12	Coahuila de Zaragoza	38987
13	Veracruz de Ignacio de la Llave	37975
14	Tabasco	36761
15	Tamaulipas	34430
16	San Luis Potosí	33017
17	Michoacán de Ocampo	27317
18	Chihuahua	26838
19	Baja California	25369
20	Sinaloa	23592
21	Yucatán	23098
22	Oaxaca	22986
23	Guerrero	22699
24	Querétaro	20084
25	Durango	19141
26	Hidalgo	17458
27	Zacatecas	15144
28	Quintana Roo	13960
29	Baja California Sur	13721
30	Aguascalientes	12820
31	Tlaxcala	7991
32	Chiapas	7268
33	Morelos	7171
34	Colima	7027

Archivo generado

CONCLUSIONES

Los archivos son una herramienta fundamental y ubicua en el funcionamiento de las computadoras, ya que nos permiten representar eficientemente datos almacenados de forma persistente. Gracias a lo anterior, se utilizan en el desarrollo de una gran cantidad de programas. Es importante conocer su funcionamiento interno, para así prevenir la aparición de errores en nuestros programas y aprovechar las funcionalidades que proporcionan.