**EJERCICIO TIPO EXAMEN**

Esta es una evaluación que corresponde a una ejecución práctica y tiene un 40% de ponderación sobre la nota final de la asignatura. El tiempo para desarrollar esta evaluación es de 5 horas pedagicas y se realiza de manera individual en laboratorio La evaluación consiste en: Desarrollar una aplicación en Python utilizando los conceptos de programación desarrollados durante la asignatura:

* Estructuras de entrada y salida
* Estructuras de decisión
* Estructuras de repetición
* Colecciones
* Funciones
* Manejo de archivos
* Uso de github como repositorio de código

Preste atención en la rúbrica, donde se señala explícitamente que de haber errores de sintaxis no obtendrá puntaje en ningún ítem

Una empresa llamada **Planing Solutions** necesita analizar los datos de sus trabajadores para generar algunos reportes. Usted ha sido contratado para desarrollar un prototipo en Python que cumpla con los siguientes requerimientos:

La aplicación debe permitir analizar los sueldos de 10 empleados, cuyos nombres y cargos están previamente definidos y cuyos sueldos se generarán aleatoriamente entre $300.000 y $2.500.000. La lista de empleados es la siguiente:

trabajadores = [

{"nombre": "Juan Pérez", "cargo": "Consultor TI"},

{"nombre": "María García", "cargo": "Analista"},

{"nombre": "Carlos López", "cargo": "Programador"},

{"nombre": "Ana Martínez", "cargo": "Jefe de Proyecto"},

{"nombre": "Pedro Rodríguez", "cargo": "Consultor TI"},

{"nombre": "Laura Hernández", "cargo": "Analista"},

{"nombre": "Miguel Sánchez", "cargo": "Programador"},

{"nombre": "Isabel Gómez", "cargo": "Jefe de Proyecto"},

{"nombre": "Francisco Díaz", "cargo": "Consultor TI"},

{"nombre": "Elena Fernández", "cargo": "Analista"}

]

**Requerimientos del Prototipo**

La aplicación deberá tener un menú con las siguientes funcionalidades:

1. **Asignar sueldos aleatorios**
2. **Clasificar sueldos**
3. **Ver estadísticas**
4. **Generar reporte de sueldos**
5. **Salir del programa**

**1. Asignar sueldos aleatorios**

Desarrolle una función que genere sueldos aleatorios para los 10 empleados. Esta función será utilizada para inicializar los datos necesarios para la ejecución del programa.

**2. Clasificar sueldos**

Desarrolle una función que muestre la lista de empleados con su respectivo sueldo y clasificación según el siguiente esquema:

* **Sueldos menores a $800.000**
* **Sueldos entre $800.000 y $2.000.000**
* **Sueldos superiores a $2.000.000**

El formato de salida debe ser similar al siguiente ejemplo:

Pantalla de computadora con letras

Descripción generada automáticamente con confianza baja

**3. Ver estadísticas**

Desarrolle una función que muestre por pantalla los siguientes datos con respecto a los sueldos:

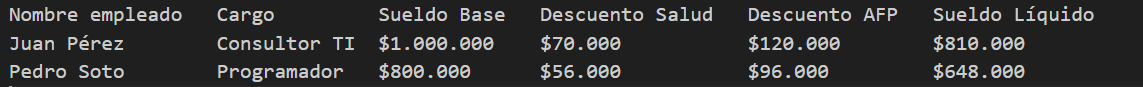
* **Sueldo más alto**
* **Sueldo más bajo**
* **Promedio de sueldos**
* **Media geométrica de sueldos**

**4. Generar reporte de sueldos**

Desarrolle una función que muestre el detalle de los sueldos de los trabajadores, aplicando las siguientes reglas de negocio:

* **Descuento salud**: 7%
* **Descuento AFP**: 12%
* **Sueldo líquido**: calculado en base al sueldo base menos el descuento en salud y menos el descuento AFP.

El formato de salida debe ser similar al siguiente ejemplo y debe ser exportado a un archivo CSV:



**5. Salir del programa**

La aplicación deberá finalizar mostrando un mensaje de salida con sus datos. El mensaje debe ser similar al siguiente:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Solución:

import random

import csv

import math

# Lista de empleados con nombre y cargo

trabajadores = [

{"nombre": "Juan Pérez", "cargo": "Consultor TI"},

{"nombre": "María García", "cargo": "Analista"},

{"nombre": "Carlos López", "cargo": "Programador"},

{"nombre": "Ana Martínez", "cargo": "Jefe de Proyecto"},

{"nombre": "Pedro Rodríguez", "cargo": "Consultor TI"},

{"nombre": "Laura Hernández", "cargo": "Analista"},

{"nombre": "Miguel Sánchez", "cargo": "Programador"},

{"nombre": "Isabel Gómez", "cargo": "Jefe de Proyecto"},

{"nombre": "Francisco Díaz", "cargo": "Consultor TI"},

{"nombre": "Elena Fernández", "cargo": "Analista"}

]

# Asignación de sueldos aleatorios

sueldos = []

def asignar\_sueldos\_aleatorios():

global sueldos

sueldos = [random.randint(300000, 2500000) for \_ in range(10)]

print("Sueldos asignados aleatoriamente.")

def clasificar\_sueldos():

print("Sueldos menores a $800.000 TOTAL:", len([s for s in sueldos if s < 800000]))

for trabajador, sueldo in zip(trabajadores, sueldos):

if sueldo < 800000:

print(f"Nombre empleado: {trabajador['nombre']} Cargo: {trabajador['cargo']} Sueldo: ${sueldo}")

print("\nSueldos entre $800.000 y $2.000.000 TOTAL:", len([s for s in sueldos if 800000 <= s <= 2000000]))

for trabajador, sueldo in zip(trabajadores, sueldos):

if 800000 <= sueldo <= 2000000:

print(f"Nombre empleado: {trabajador['nombre']} Cargo: {trabajador['cargo']} Sueldo: ${sueldo}")

print("\nSueldos superiores a $2.000.000 TOTAL:", len([s for s in sueldos if s > 2000000]))

for trabajador, sueldo in zip(trabajadores, sueldos):

if sueldo > 2000000:

print(f"Nombre empleado: {trabajador['nombre']} Cargo: {trabajador['cargo']} Sueldo: ${sueldo}")

print("\nTOTAL SUELDOS:", sum(sueldos))

def ver\_estadisticas():

sueldo\_max = max(sueldos)

sueldo\_min = min(sueldos)

sueldo\_promedio = sum(sueldos) / len(sueldos)

sueldo\_geom = math.exp(sum(math.log(sueldo) for sueldo in sueldos) / len(sueldos))

print(f"Sueldo más alto: ${sueldo\_max}")

print(f"Sueldo más bajo: ${sueldo\_min}")

print(f"Promedio de sueldos: ${sueldo\_promedio:.2f}")

print(f"Media geométrica de sueldos: ${sueldo\_geom:.2f}")

def reporte\_sueldos():

with open('reporte\_sueldos.csv', 'w', newline='') as file:

writer = csv.writer(file)

writer.writerow(["Nombre empleado", "Cargo", "Sueldo Base", "Descuento Salud", "Descuento AFP", "Sueldo Líquido"])

for trabajador, sueldo in zip(trabajadores, sueldos):

descuento\_salud = sueldo \* 0.07

descuento\_afp = sueldo \* 0.12

sueldo\_liquido = sueldo - descuento\_salud - descuento\_afp

writer.writerow([trabajador["nombre"], trabajador["cargo"], sueldo, descuento\_salud, descuento\_afp, sueldo\_liquido])

print(f"Nombre empleado: {trabajador['nombre']} Cargo: {trabajador['cargo']} Sueldo Base: ${sueldo} Descuento Salud: ${descuento\_salud:.2f} Descuento AFP: ${descuento\_afp:.2f} Sueldo Líquido: ${sueldo\_liquido:.2f}")

def salir\_programa():

print("Finalizando programa…")

print("Desarrollado por [Su Nombre]")

print("RUT [Su RUT]")

def menu():

while True:

print("\nMenu:")

print("1. Asignar sueldos aleatorios")

print("2. Clasificar sueldos")

print("3. Ver estadísticas")

print("4. Generar reporte de sueldos")

print("5. Salir del programa")

opcion = int(input("Seleccione una opción: "))

if opcion == 1:

asignar\_sueldos\_aleatorios()

elif opcion == 2:

clasificar\_sueldos()

elif opcion == 3:

ver\_estadisticas()

elif opcion == 4:

reporte\_sueldos()

elif opcion == 5:

salir\_programa()

break

else:

print("Opción no válida. Intente de nuevo.")

if \_\_name\_\_ == "\_\_main\_\_":

menu()