

Unidad de Competencia 2

Nombre: Huancas Leuyacc Anselmo Junior

- La entrega fuera de hora representa 13 puntos menos
- Se debe visualizar el proceso a travez de commit en git
- Compartir el link de github
- Entregar el documento al ula virtual en pdf

```
USUARIO@DESKTOP-5DG7UNO MINGW64 ~/2022-LP3-S2 (master)
$ git add -A

USUARIO@DESKTOP-5DG7UNO MINGW64 ~/2022-LP3-S2 (master)
$ git commit -m "UC2-Huancas-Anselmo"
[master 3113c81] UC2-Huancas-Anselmo
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Semana 07/UC2/UnidaddeCompetencia2.docx

USUARIO@DESKTOP-5DG7UNO MINGW64 ~/2022-LP3-S2 (master)
$
```

1. Crear un módulo llamado: **gestion_archivos.py**, que considere los métodos:
 - a. crear_archivo, que reciba como parámetro el nombre del archivo y el contenido.
 - b. eliminar_archivo, que considere como parámetro el nombre del archivo.
 - c. agregar_contenido_archivo, que considere como parámetro el nombre del archivo y el contenido.
 - d. leer_archivo, que considere como parámetro el nombre del archivo.

```
temp.py x gestion_archivos.py x
1  #-*- coding: utf-8 -*-
2  ***
3  Created on Wed Jun  1 11:27:50 2022
4
5  @author: USUARIO
6  ***
7
8  import os
9
10 # Para crear archivo recibe como parametro el nombre del archivo y el contenido
11 def crear_archivo(nombre, contenido):
12     archivo = open(nombre, "wt")
13     archivo.write(contenido)
14     archivo.close()
15
16 #Para eliminar recibe como parametro el nombre del archivo a eliminar
17 def eliminar_archivo(nombre):
18     os.remove(nombre)
19
20 #Para agregar contenido a un archivo plano, debe existir un archivo
21 #Se envia como parametro el nombre y el contenido del archivo
22 def agregar_contenido_archivo(nombre, contenido):
23     archivo = open(nombre, "at")
24     archivo.write("\n" + contenido)
25     archivo.close()
26
27 # Para leer un archivo, ya debe existir un archivo plano
28 # Ejemplo: txt, py, java, ...
29 # Recibe como parametro el nombre del archivo a leer
30 # y devuelve el contenido del archivo
31 def leer_archivo(nombre):
32     archivo = open(nombre, "rt", encoding='utf8')
33     contenido = archivo.read()
34     return contenido
```

```

USUARIO@DESKTOP-5DG7UNO MINGW64 ~/2022-LP3-S2 (master)
$ git commit -m "gestion_archivos Huancas_Anselmo"
[master d8607ad] gestion_archivos Huancas_Anselmo
1 file changed, 34 insertions(+)
create mode 100644 Semana 07/UC2/gestion_archivos.py

USUARIO@DESKTOP-5DG7UNO MINGW64 ~/2022-LP3-S2 (master)
$ git push https://github.com/JuniorHL/LP03.git
Enumerating objects: 12, done.
Counting objects: 100% (12/12), done.
Delta compression using up to 4 threads
Compressing objects: 100% (10/10), done.
Writing objects: 100% (10/10), 13.02 KiB | 4.34 MiB/s, done.
Total 10 (delta 5), reused 0 (delta 0), pack-reused 0

remote: Resolving deltas: 100% (5/5), completed with 2 local objects.
To https://github.com/JuniorHL/LP03.git
40e6b7b..d8607ad master -> master

```

En otro archivo llamado: **aplicacion.py**, diseñar:

a. Un menú de opciones:

Menú
1. Crear archivo
2. Eliminar archivo
3. Agregar contenido
4. Mostrar contenido de archivo
5. Salir

b. Considere los métodos en **aplicacion.py**

- i.menu()
- ii.crear()
- iii.eliminar()
- iv.agregar()
- v.listar()
- vi.salir()
- vii.error()

C:\Users\USUARIO\2022-LP3-S2\Semana 07\JC2\Aplicacion.py

```
temp.py × gestion_archivos.py × Aplicacion.py ×
8 import gestion_archivos
9
10 def menu():
11     print("*** MENU PRINCIPAL ***")
12     print("1. Crear Archivo")
13     print("2. Eliminar Archivo")
14     print("3. Agregar Contenido")
15     print("4. Mostrar Contenido de Archivo")
16     print("5. Salir")
17
18 def crear():
19     print("*** CREAM ARCHIVO ***")
20     archivo = input("Crear archivo: ")
21     contenido = input("Contenido del archivo: ")
22     gestion_archivos.crear_archivo(archivo, contenido)
23
24 def eliminar():
25     print("*** ELIMINAR ARCHIVO ***")
26     archivo = input("Quiero eliminar el archivo: ")
27     gestion_archivos.eliminar_archivo(archivo)
28
29 def agregar():
30     print("*** AGREGAR CONTENIDO A UN ARCHIVO ***")
31     archivo = input("Quiero agregar contenido al archivo: ")
32     contenido = input("El contenido adicional del archivo sera: ")
33     gestion_archivos.agregar_contenido_archivo(archivo, contenido)
34
35 def listar():
36     print("*** ELIMINAR/MOSTRAR CONTENIDO DE UN ARCHIVO ***")
37     archivo = input("Quiero mostrar el contenido del archivo: ")
38     print(gestion_archivos.leer_archivo(archivo))
39
40 def salir():
41     print("Gracias... Vuelva pronto")
42
43 def error():
44     print("Opcion incorrecta")
45
46 # La logica para el menu de opciones
47 opcion = 1
48 while opcion != 5:
49     menu()
50     opcion = int(input("Seleccione una opcion (1 - 5): "))
51     if opcion == 1:
52         crear()
53     elif opcion == 2:
54         eliminar()
55     elif opcion == 3:
56         agregar()
57     elif opcion == 4:
58         listar()
59     elif opcion == 5:
60         salir()
61     else:
62         error()
63
64
```

```
USUARIO@DESKTOP-5DG7UNO MINGW64 ~/2022-LP3-S2 (master)
$ git add -A

USUARIO@DESKTOP-5DG7UNO MINGW64 ~/2022-LP3-S2 (master)
$ git commit -m "UC2 - Aplicacion.py de Anselmo Huancas"
[master b048a83] UC2 - Aplicacion.py de Anselmo Huancas
 2 files changed, 68 insertions(+)
 create mode 100644 Semana 07/UC2/Aplicacion.py
 create mode 100644 Semana 07/UC2/__pycache__/gestion_archivos.cpython-39.pyc

USUARIO@DESKTOP-5DG7UNO MINGW64 ~/2022-LP3-S2 (master)
$ git push https://github.com/JuniorHL/LP03.git
Enumerating objects: 10, done.
Counting objects: 100% (10/10), done.
Delta compression using up to 4 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 1.66 KiB | 849.00 KiB/s, done.
Total 7 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/JuniorHL/LP03.git
   d8607ad..b048a83  master -> master

USUARIO@DESKTOP-5DG7UNO MINGW64 ~/2022-LP3-S2 (master)
$ |
```

<https://github.com/JuniorHL/LP03.git>