

Unidad de Competencia 2

Nombre: Motta Mendoza Miguel Angel

- 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

```
Miguel@Poyo MINGW64 ~/2022 LP3-S1/Semana07 (master)
$ git add -A

Miguel@Poyo MINGW64 ~/2022 LP3-S1/Semana07 (master)
$ git commit -m "UC 2 - Inicio de Motta Mendoza"
[master (root-commit) 92715a6] UC 2 - Inicio de Motta Mendoza
1 file changed, 37 insertions(+)
create mode 100644 UC2/Gestion_Archivos.py

Miguel@Poyo MINGW64 ~/2022 LP3-S1/Semana07 (master)
$ git branch -M main

Miguel@Poyo MINGW64 ~/2022 LP3-S1/Semana07 (main)
$ git remote add origin https://github.com/Miguel2897/Semana07.git

Miguel@Poyo MINGW64 ~/2022 LP3-S1/Semana07 (main)
$ git push -u origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 678 bytes | 678.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Miguel2897/Semana07.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

1. Crear un módulo llamado: **gestion_archivos.py**, que considere los métodos:
 1. crear_archivo, que reciba como parámetro el nombre del archivo y el contenido.
 2. eliminar_archivo, que considere como parámetro el nombre del archivo.
 3. agregar_contenido_archivo, que considere como parámetro el nombre del archivo y el contenido.
 4. leer_archivo, que considere como parámetro el nombre del archivo.

```

5  @author: Miguel
6  """
7
8  import os
9  #Para crear archivo, recibe como parametro el nombre del archivo
10 # y e l contenido del archivo
11
12 def crear_archivos(nombre, contenido):
13     archivo = open(nombre, "wt")
14     archivo.write(contenido)
15     archivo.close()
16
17 # para eliminar recibe como parametro el nombre del archivo a eiminar
18 def eliminar_archivo(nombre):
19     os.remove(nombre)
20
21 #para gregar contenido a un archivo plano, debe existir un archivo
22 # se envia como parametro el nombre y el contenido del archivo
23
24 def agregar_contenido_archivo(nombre, contenido):
25     archivo = open(nombre, "at")
26     archivo.write("\n " + contenido)
27     archivo.close()
28
29 #para leer un archivo, ya debe existir un rarchivo plano
30 #ejmplo: txt, py, java ..
31 #Recibe como parametro el nombre del archivo a leer
32 #y devuelve el contenido del archivo
33
34 def leer_archivo(nombre):

```

```

Miguel@Poyo MINGW64 ~/2022 LP3-S1/Semana07 (main)
$ git push -u origin main
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (4/4), 678 bytes | 678.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Miguel2897/Semana07.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

Miguel@Poyo MINGW64 ~/2022 LP3-S1/Semana07 (main)
$ git push
Everything up-to-date

```

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**

1. menu()
2. crear()
3. eliminar()
4. agregar()
5. listar()
6. salir()
7. error()

```
5  @author: Miguel
6  """
7
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("* CREAR ARCHIVO *")
20     archivo = input("Crear archivo: ")
21     contenido = input("Contenido del archivo: ")
22     Gestion_Archivos.crear_archivos(archivo, contenido)
23
24 def eliminar():
25     print("* ELIMINAR ARCHIVO *")
26     archivo = input("Quiero eliminar 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 será: ")
33     Gestion_Archivos.agregar_contenido_archivo(archivo, contenido)
34
35 def listar():
36     print("* 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 lógica 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()

```

```

Miguel@Poyo MINGW64 ~/2022 LP3-S1/Semana07 (main)
$ git add -A

Miguel@Poyo MINGW64 ~/2022 LP3-S1/Semana07 (main)
$ git commit -m "UC 2 - Aplicacion.py de Motta Mendoza"
[main d6b0213] UC 2 - Aplicacion.py de Motta Mendoza
1 file changed, 62 insertions(+)
create mode 100644 UC2/Aplicacion.py

Miguel@Poyo MINGW64 ~/2022 LP3-S1/Semana07 (main)
$ git push
Enumerating objects: 6, done.
Counting objects: 100% (6/6), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (4/4), 918 bytes | 918.00 KiB/s, done.
Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/Miguel2897/Semana07.git
 92715a6..d6b0213  main -> main

```

Git: <https://github.com/Miguel2897/Semana07.git>