**FUNDAMENTOS DE**

**PROGRAMACIÓN**

**LABORATORIO N° 14**

**ARCHIVOS EN PYTHON**

**CODIGO DEL CURSO:**



|  |  |  |
| --- | --- | --- |
| ***Alumno(s)*** | | ***Nota*** |
|  | |  |
|  | |  |
|  | |  |
| ***Grupo*** |  | |
| ***Ciclo*** |  | |
| ***Fecha de entrega*** |  | |

**I.- OBJETIVOS:**

* Elaborar y diseñar programas en Python que permitan leer y escribir datos en archivos

**II.- SEGURIDAD:**

|  |  |
| --- | --- |
|  | **Advertencia:**  **En este laboratorio está prohibida la manipulación del hardware, conexiones eléctricas o de red; así como la ingestión de alimentos o bebidas.** |

**III.- FUNDAMENTO TEÓRICO:**

1. Revisar el texto guía que está en el campus Virtual.

**IV.- NORMAS EMPLEADAS:**

1. No aplica

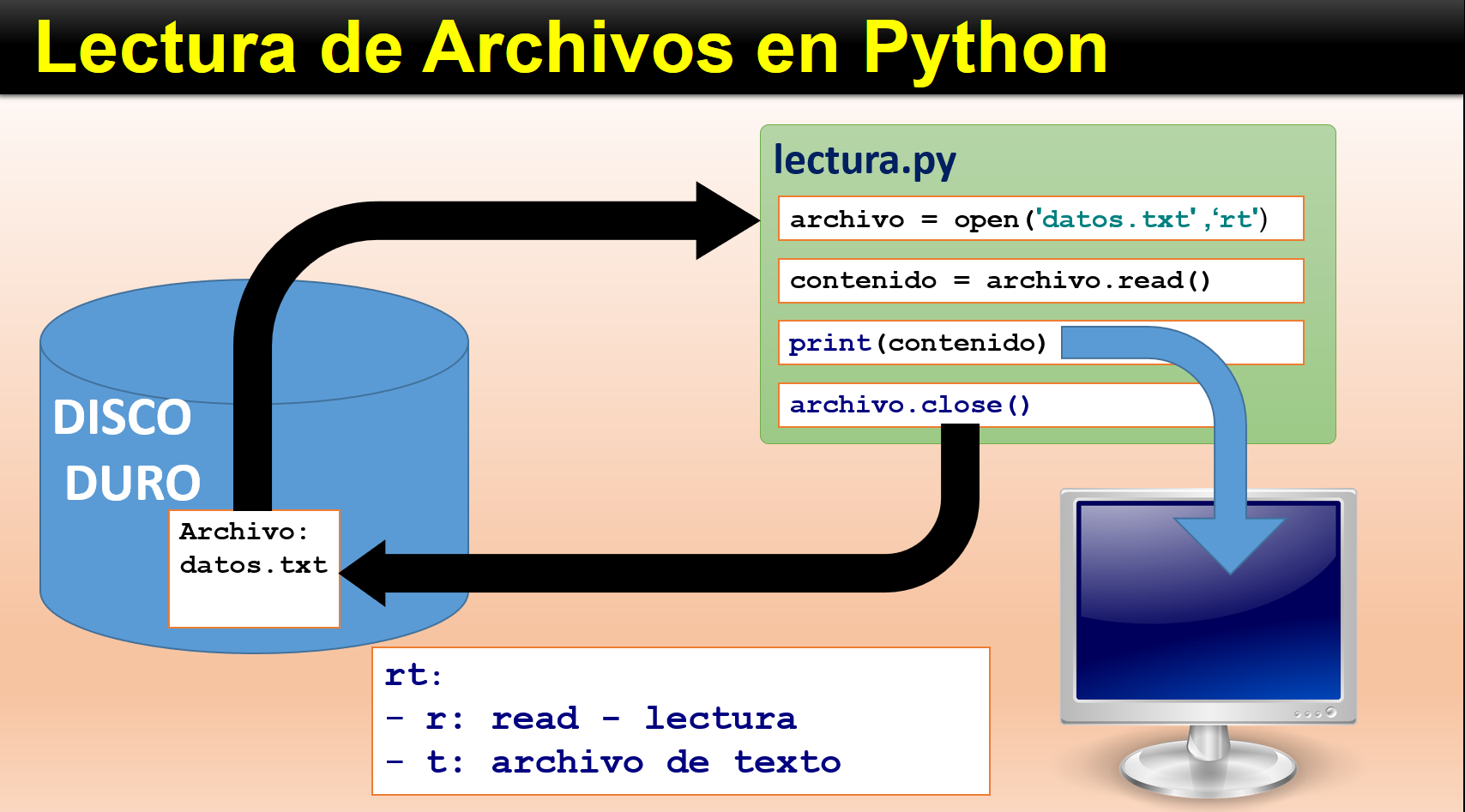
**V.- RECURSOS:**

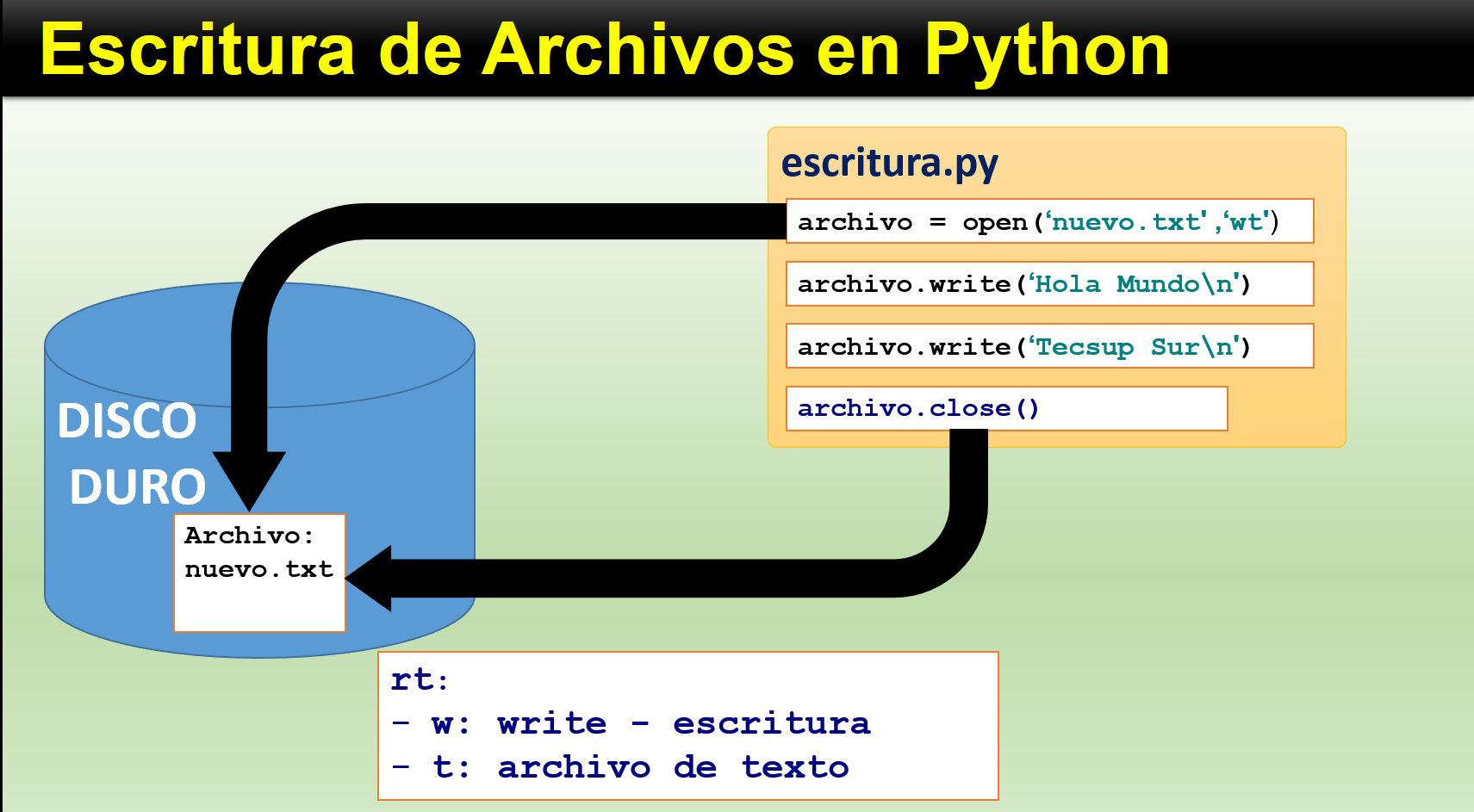
* En este laboratorio cada alumno trabajará con un equipo con Windows 10.

**VI.- METODOLOGÍA PARA EL DESARROLLO DE LA TAREA:**

* El desarrollo del laboratorio es individual.

**VII.- MARCO TEÓRICO:**



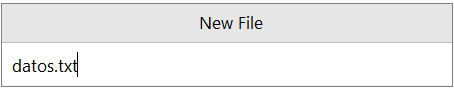


**VIII.- PROCEDIMIENTO:**

Crear un nuevo proyecto llamado **“Lab14”**. Crear, dentro de este proyecto, un programa **Python** para cada ejercicio.

**EJERCICIOS DE APLICACIÓN:**

**LECTURA DE ARCHIVOS**

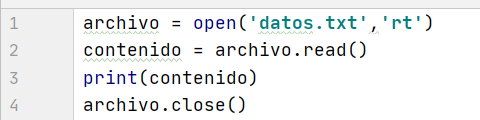
1. **Lectura simple de archivos en python**
   1. En el nuevo proyecto. Crear un archivo de texto con el nombre “**datos.txt**”:

(Utilice la opción: **File** 🡪 **New** 🡪 **File**: y asigne el nombre **datos.txt**

* 1. Adicione el siguiente contenido al nuevo archivo desde el editor de **PyCharm**:

|  |
| --- |
| Este es un archivo de texto  lo utilizaremos como ejemplo  para realizar pruebas de lectura  desde un programa en Python |

* 1. Crear un nuevo programa denominado **archivos\_01.py** con el siguiente contenido:

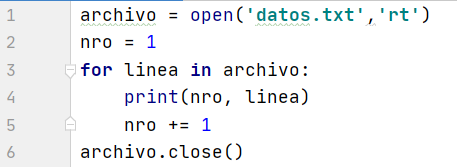


* 1. Indique qué es lo que hace cada una de las líneas numeradas del programa anterior:

|  |  |
| --- | --- |
| **No Linea** | **Descripción** |
| 1 | Se crea la variable archivo luego se se abre el archivo con la funcion open() ponemos en comillas el nombre del archivo y ademas le damos la orden rt que significa que lea el archivo txt. |
| 2 | Crea otra variable con el contenido y le asigna lo que esta en el archivo, antes se llama a la funcion read() que gaurda el contenido en esa variable. |
| 3 | Imprime el contenido del archivo txt mendiante lo que se guardo justamente en la variable contenido. |
| 4 | Cierra el archivo abierto con la funcion close(). |

* 1. Ejecute el programa y verifique que se muestre en la consola de Python el contenido del archivo “**datos.txt**” que acaba de ser leido por este programa. Adjunte la captura de la salida del programa:

1. **Lectura de archivo por líneas**
   1. Crear un nuevo programa denominado **archivos\_02.py** con el siguiente contenido:

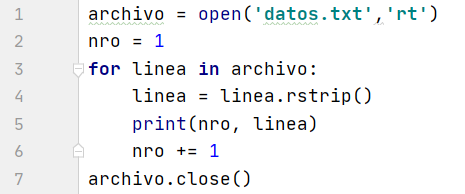


* 1. Indique qué es lo que hace cada una de las líneas numeradas del programa anterior:

|  |  |
| --- | --- |
| **No Linea** | **Descripción** |
| 1 | Se crea la variable archivo, luego de damos su valor o contenido con le funcion open() con su parametro rt que es para la lectura del archivo. |
| 2 | Creamos una variable nro y le asiganamos el valor de 1. |
| 3 | Iniciamos un for con la variable linea para que itere en el contenido de la variable archivo. |
| 4 | Se usa la funcion print() para imprimir la variable nro, junto con linea que seria cada linea en el contenido del archivo txt. |
| 5 | Se hace un incremento de la variable nro para que con cada iteracion se sume hasta la ultima linea. |
| 6 | Cierra el archivo abiertop anteriormente con la funcion open(), cerramos con la funcion close(). |

* 1. Ejecute el programa y verifique que se muestre en la consola de Python el contenido del archivo “**datos.txt**” que acaba de ser leido por este programa. Adjunte la captura de la salida del programa: (Indique si hay algo distinto en la salida generada por el programa)

1. **Lectura de archivo por líneas eliminando el “salto de línea” de cada linea del archivo de texto**
   1. Crear un nuevo programa denominado **archivos\_03.py** con el siguiente contenido:

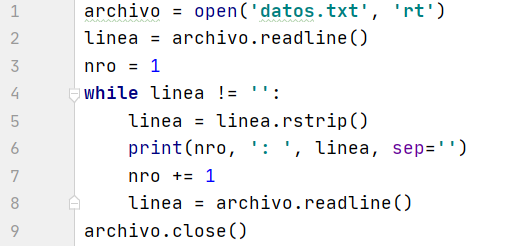


* 1. Indique qué es lo que hace cada una de las líneas numeradas del programa anterior:

|  |  |
| --- | --- |
| **No Linea** | **Descripción** |
| 1 | Primero se crea la variable archivo para luego abrir un archivo txt de nombre datos con la funcion open y ademas su parametro rt. |
| 2 | Se crea la variable nro y se le asigna el valor de 1. |
| 3 | Inicamos un for con la variable linea para que itere sobre cada linea en la variable que creamos archivo que tiene el contenido del archivo txt datos. |
| 4 | Luego le decimos con la funcion rstrip que elimine los espacion vacios vacios que puedan existir al inicio y al final de cada linea. |
| 5 | Se usa la funcion print() y se imprime la variabloe nro y las lineas que pasen de la anterior linea de condigo, se imprime cada linea. |
| 6 | Aumentamos la variable nro en cada iteracion con mas + 1 |
| 7 | Cerramos el archivo abierto con la funcion close() |

* 1. Ejecute el programa y verifique que se muestre en la consola de Python el contenido del archivo “**datos.txt**” que acaba de ser leido por este programa. Adjunte la captura de la salida del programa: (Verifique que ya no aparezcan lineas en blanco en la salida generada)

1. **Lectura de archivo por líneas usando bucle while**
   1. Crear un nuevo programa denominado **archivos\_04.py** con el siguiente contenido:

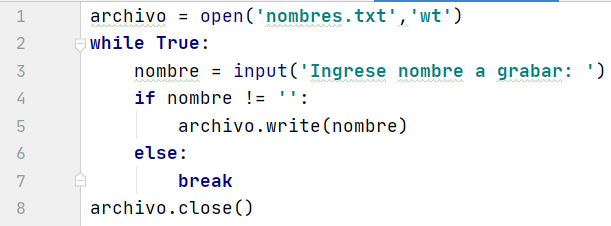


* 1. Indique qué es lo que hace cada una de las líneas numeradas del programa anterior:

|  |  |
| --- | --- |
| **No Linea** | **Descripción** |
| 1 | Primero se crea la variable archivo para luego abrir un archivo txt de nombre datos con la funcion open y ademas su parametro rt. |
| 2 | Se crea una variable linea que se le asiga el llamado al archivo y la funcion readline (). Esta funcion lee una sola linea del archivo. |
| 3 | Se crea la variable nro y se le asigna el valor de 1. |
| 4 | Creamos un while que se ejecutara hasta que la variable linea no tenga ningun valor |
| 5 | Luego le decimos con la funcion rstrip que elimine los espacion vacios vacios que puedan existir al inicio y al final de cada linea. |
| 6 | Se usa la funcion print para imprimir primero el numero luego concatenamos dos puntos luego imprime la linea, usamos el parametro de la funcion print sep, la funcion print tiene por default sep “ ” osea que cuando contaneamos automaticamente le da un espaci, cambiamos esto para que despues del 1 no haya espacio y despues de los dos puntos tampoco y este los dos puntos y la linea. |
| 7 | Aumentamos la variable nro en cada iteracion con mas + 1 |
| 8 | En esta linea se llama otra vez a la funcion readline() esto para que siga leyendo la siguiente linea hasta que no encuentre nada donde terminara. |
| 9 | Cerramos el archivo abierto con la funcion close() |

* 1. Ejecute el programa y verifique que se muestre en la consola de Python el contenido del archivo “**datos.txt**” que acaba de ser leido por este programa. Adjunte la captura de la salida del programa:

1. **Escritura de archivo en Python**
   1. Crear un nuevo programa denominado **archivos\_05.py** con el siguiente contenido:

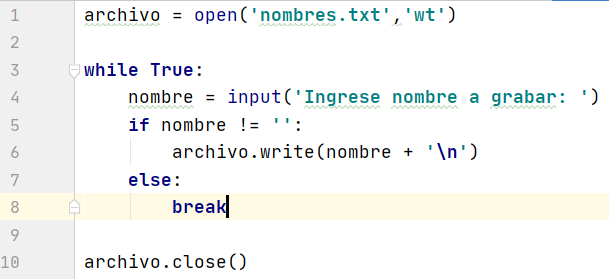


* 1. Indique qué es lo que hace cada una de las líneas numeradas del programa anterior:

|  |  |
| --- | --- |
| **No Linea** | **Descripción** |
| 1 | Creamos la variable archivo, luego llamos a la funcion opn() para abrir el achivo txt y ademas le decimos el parametro wt que nos sirve para grabar en ese archivo txt |
| 2 | Creamos un bucle while y ademas le decimos que inicie en True, esto hace que el bucle se ajecute hasta que se use el break. |
| 3 | Creamos una variable nombre y le asignamos el valor que se ingrese por teclado esto con la funcion input |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |

* 1. Ejecute el programa y verifique que se haya creado en la carpeta de su proyecto un archivo de nombre “**nombres.txt**” Adjunte la imagen que muestre la existencia del archivo creado. Abra el archivo y muestre su contenido: (¿Los nombres se han almacenado en lineas distintas? Explique

1. **Escritura de archivo en Python – Incluir salto de linea**
   1. Crear un nuevo programa denominado **archivos\_06.py** con el siguiente contenido:

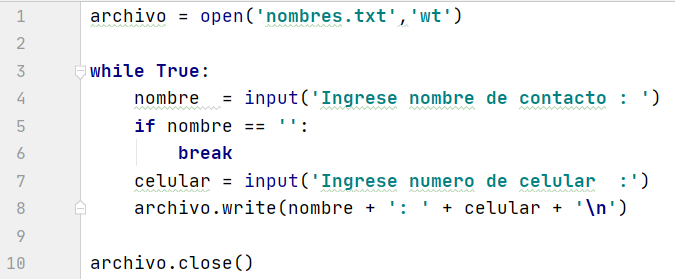


* 1. Indique qué es lo que hace cada una de las líneas numeradas del programa anterior:

|  |  |
| --- | --- |
| **No Linea** | **Descripción** |
| 1 |  |
| 3 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |
| 10 |  |

* 1. Ejecute el programa y verifique que se haya creado/reemplazado en la carpeta de su proyecto el archivo de nombre “**nombres.txt**” Adjunte la imagen que muestre la existencia del archivo creado. Abra el archivo y muestre su contenido: (¿Los nombres se han almacenado en lineas distintas? Explique

1. **Escritura de archivo en Python – Registro de contactos**
   1. Crear un nuevo programa denominado **archivos\_07.py** con el siguiente contenido:

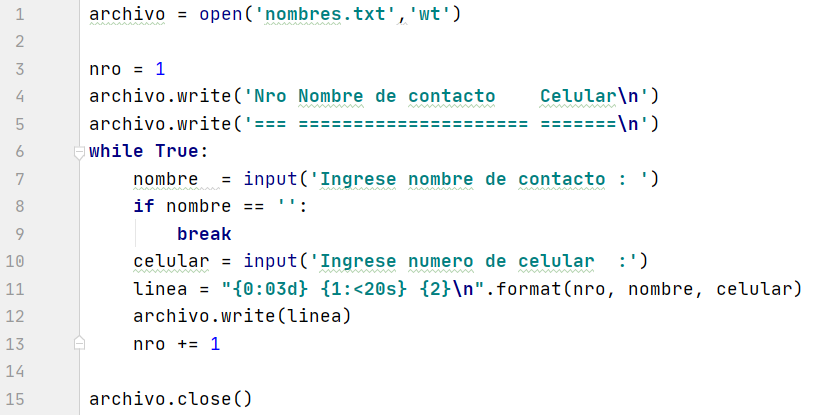


* 1. Indique qué es lo que hace cada una de las líneas numeradas del programa anterior:

|  |  |
| --- | --- |
| **No Linea** | **Descripción** |
| 1 |  |
| 3 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |
| 10 |  |

* 1. Ejecute el programa y verifique que se haya creado/reemplazado en la carpeta de su proyecto el archivo de nombre “**nombres.txt**” Adjunte la imagen donde se muestre su contenido:

1. **Escritura de archivo en Python – Registro de contactos usando formato de alineación**
   1. Crear un nuevo programa denominado **archivos\_08.py** con el siguiente contenido:



* 1. Indique qué es lo que hace cada una de las líneas numeradas del programa anterior:

|  |  |
| --- | --- |
| **No Linea** | **Descripción** |
| 1 |  |
| 3 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |
| 9 |  |
| 10 |  |
| 11 |  |
| 12 |  |
| 13 |  |
| 15 |  |

* 1. Ejecute el programa y verifique que se haya creado/reemplazado en la carpeta de su proyecto el archivo de nombre “**nombres.txt**” Adjunte la imagen donde se muestre su contenido:

**EJERCICIOS PROPUESTOS**

1. Crear un programa en Pyhon que lea, por teclado, tu nombre, curso, 2 notas, calcule el promedio y la condicion *(aprobado / desaprobado)*. Posteriomente, que guarde en un archivo de texto la fecha actual, el nombre, curso, promedio y condición. Finalmente, que muestre en pantalla el contenido del archivo.

|  |
| --- |
| **Programa en Python** |
|  |

1. Elabore una aplicación Python que registre ventas. La información a registrar ser la siguiente:
   * Información ingresada por teclado 🡪Nombre de producto, precio, cantidad
   * Información que se debe obtener o calcular 🡪 Fecha de venta *(fecha actual)*, subtotal, monto de igv *(18%)* y total

Utilice el modo **‘at**’ para el trabajo con el archivo de registro de ventas. Se puede emplear un bucle para ingresar varias ventas en la ejecución de la aplicación.

|  |
| --- |
| **Programa en Python** |
|  |

**OBSERVACIONES:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**CONCLUSIONES:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**"Dime y lo olvido, enséñame y lo recuerdo, involúcrame y lo aprendo”. Benjamín Franklin (1706-1790)**