



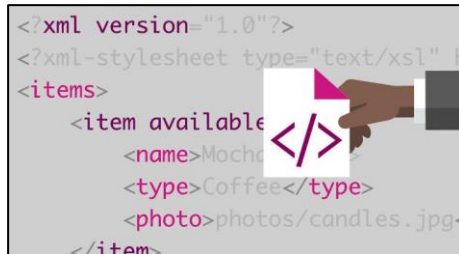
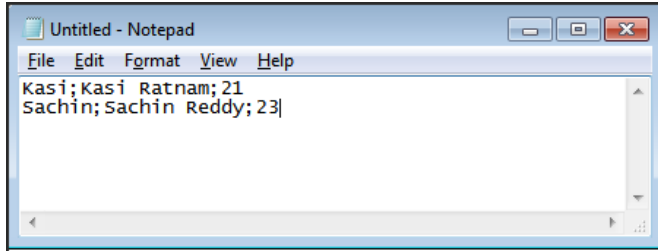
**M6 - Acceso a Datos**



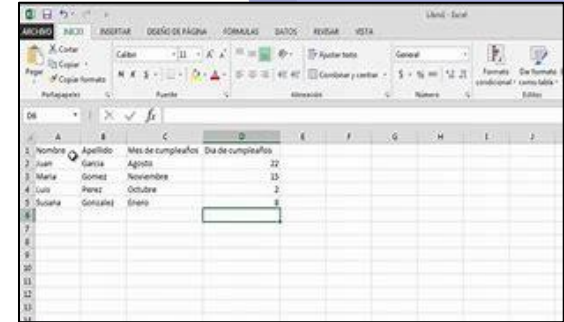
# **UF1 – Trabajar con diferentes formatos: CSV**

# Trabajar con archivos de diferentes formatos

- En Python, trabajar con archivos de diferentes formatos es una tarea común y la biblioteca estándar proporciona módulos específicos para manejar diferentes tipos de archivos. A continuación, se presenta una guía básica sobre cómo trabajar con archivos CSV, XML, JSON, y XLS



```
{
  "employees": [
    {
      "name": {
        "first_name": "Pedro",
        "last_name": "Gómez"
      },
      "age": 53,
      "car": "Audi A5",
      "country": "Mexico"
    },
    {
      "name": {
        "first_name": "Marina",
        "last_name": "Fernández"
      },
      "age": 42,
      "car": "Seat Tarraco",
      "country": "Colombia"
    }
  ]
}
```



# Trabajar con Archivos CSV en Python

- Un fichero CSV (Valores Separados por Comas o Comma Separated Values) es un archivo de texto en el cual, como su nombre indica, los caracteres están separados por comas, formando una especie de tabla en filas y columnas.
- Es fácilmente entendible tanto por personas, como por máquinas
- Típicamente se usa para importación/exportación de bases de datos sencillas u hojas de cálculo.
- Para trabajar con ficheros de este tipo desde un programa Python debemos importar el **módulo csv**

# Lectura de un fichero CSV

	M6_Alumnos.csv
1	Apellido(s);Nombre;Dirección de correo
2	Bretones López;David;davidbrelap@campus.monlau.com
3	Calonge Sendra;Ferran Josep;ferrancalsen@campus.monlau.com
4	Castillejo García;Raül;raulcasgar@campus.monlau.com
5	Crespo Cambón;Marc;marccrecam@campus.monlau.com
6	Farré Ruiz;Pau;paufarrui@campus.monlau.com
7	Farreny Zaragoza;Roger;rogerfarzar@campus.monlau.com
8	García de los Santos;Hugo;hugogarsan@campus.monlau.com
9	Jodar Ramos;Martí;martijodram@campus.monlau.com
10	Lin;Feihong;feihonglin@campus.monlau.com
11	Martínez Zahíño;Aleix;aleixmarzah@campus.monlau.com
12	Mas De Xaxars;Martí Antoni;martimasxax@campus.monlau.com
13	Masó Boltà;Victor;victormasbol@campus.monlau.com
14	Molina Pomeyrol;Jordi;jordimolpom@campus.monlau.com
15	Monroy Fernandez;Victor;victormonfer@campus.monlau.com
16	Morera Navarro;Oscar;oscarmonnav@campus.monlau.com
17	Ortiz Auge;Marti;martiortaug@campus.monlau.com
18	Ortiz Roque;Marc;marcortroq@campus.monlau.com
19	Palacios Fabregas;Jonathan;jonathanpalfab@campus.monlau.com
20	Parralejo Capellera;Joel;joelparcap@campus.monlau.com
21	Pleguezuelos Cabrera;Axel;axelplecab@campus.monlau.com
22	Ramos Guixà;Gerard;gerardramgui@campus.monlau.com
23	Rubel Díaz;Paola;paolarubdia@campus.monlau.com
24	Salgado Blanco;Victor;victorsalbla@campus.monlau.com
25	Sanchez Espona;Ivan;ivansanesp@campus.monlau.com
26	Soria Alvarez;Adan;adansoralv@campus.monlau.com
27	Tierno Fernandez;Hernan;hernantiefer@campus.monlau.com
28	Vilanova Paredes;Marc;marcvilpar@campus.monlau.com
29	Vifals Morales;Oriol;oriolvinmor@campus.monlau.com
30	

	A	B	C	D	E
1	Apellido(s)	Nombre	Dirección de correo		
2	Bretones Ló	David	davidbrelap@campus.monlau.com		
3	Calonge Sen	Ferran Josep	ferrancalsen@campus.monlau.com		
4	Castillejo Ga	Raül	raulcasgar@campus.monlau.com		
5	Crespo Cam	Marc	marccrecam@campus.monlau.com		
6	Farré Ruiz	Pau	paufarrui@campus.monlau.com		
7	Farreny Zara	Roger	rogerfarzar@campus.monlau.com		
8	García de los	Hugo	hugogarsan@campus.monlau.com		
9	Jodar Ramos	Martí	martijodram@campus.monlau.com		
10	Lin	Feihong	feihonglin@campus.monlau.com		
11	Martínez Zah	Aleix	aleixmarzah@campus.monlau.com		
12	Mas De Xaxa	Martí Antoni	martimasxax@campus.monlau.com		
13	Masó Boltà	Victor	victormasbol@campus.monlau.com		
14	Molina Pome	Jordi	jordimolpom@campus.monlau.com		
15	Monroy Fern	Victor	victormonfer@campus.monlau.com		
16	Morera Nava	Oscar	oscarmonnav@campus.monlau.com		
17	Ortiz Auge	Marti	martiortaug@campus.monlau.com		
18	Ortiz Roque	Marc	marcortroq@campus.monlau.com		
19	Palacios Fabi	Jonathan	jonathanpalfab@campus.monlau.com		
20	Parralejo Ca	Joel	joelparcap@campus.monlau.com		
21	Pleguezuelo	Axel	axelplecab@campus.monlau.com		
22	Ramos Guixà	Gerard	gerardramgui@campus.monlau.com		

Delimitadores: , : ; | ...

# Lectura de un fichero CSV

- Vamos a leer M6\_Alumnos.csv
- Pasos:
  1. **Importar el módulo csv** de Python
  2. **Abrir el fichero** en modo lectura
  3. **Usar** la función **csv.reader(...)** que retorna un objeto reader iterable (se puede iterar, se puede recorrer elemento a elemento en un bucle)
  4. **Recorrer el contenido del fichero** línea a línea con un bucle

# Lectura de un fichero CSV

Por defecto delimiter = ','

```
import csv

with open("../M6_Alumnos.csv", "r") as file:
    file_content = csv.reader(file)
    for line in file_content:
        print(line)
```

```
"C:\Users\user\OneDrive - Centre d'Estudis Monlau\2023 -2024\M6\Pytho
^ ['Apellido(s);Nombre;Dirección de correo']
^ ['Bretones López;David;davidbrelop@campus.monlau.com']
^ ['Calonge Sendra;Ferran Josep;ferrancalsen@campus.monlau.com']
^ ['Castillejo García;Raül;raulcasgar@campus.monlau.com']
^ ['Crespo Cambón;Marc;marccrecam@campus.monlau.com']
```

```
import csv

with open("../M6_Alumnos.csv", "r") as file: # En
    file_content = csv.reader(file, delimiter=';')
    for line in file_content:
        print(line)
```

- csv.reader devuelve una lista de valores (las líneas del fichero)
- En el bucle for obtenemos línea a línea.
- Cada línea es una nueva lista de valores (los que están separados por el delimitador)

```
"C:\Users\user\OneDrive - Centre d'Estudis Monlau\2023 -2024\M06\Pytho
3 ['Apellido(s)', 'Nombre', 'Dirección de correo']
3 ['Bretones López', 'David', 'davidbrelop@campus.monlau.com']
3 ['Calonge Sendra', 'Ferran Josep', 'ferrancalsen@campus.monlau.com']
```

# Lectura de un fichero CSV

- Si quiero tratar columnas:

```
import csv
# En mi caso, el fichero está un directorio por encima del proyecto
with open("../Mó_Alumnos.csv", "r") as file:
    file_content = csv.reader(file, delimiter=';')

    next(file_content) # Permite acceder al siguiente elemento de un iteardor,
                        # se avanza el lector una línea y así nos saltamos los encabezados
    for line in file_content:
        print('Nombre: ' + line[1] + ', Apellidos: ' + line[0])
```

```
"C:\Users\user\OneDrive - Centre d'Estudis Monlau\2023 -2024\
Nombre: Nombre, Apellidos: Apellido(s)
Nombre: David, Apellidos: Bretones López
Nombre: Ferran Josep, Apellidos: Calonge Sendra
Nombre: Raül, Apellidos: Castillejo García
```

# Escritura de un fichero CSV

- Vamos a escribir M6\_NotasAlumnos.csv
- Pasos:
  1. **Importar el módulo csv** de Python
  2. Obtener, crear o **definir los datos que deseas escribir** en el CSV
  3. **Abrir el fichero** en modo escritura
  4. **Usar** la función **csv.writer(...)** que retorna un objeto writer
  5. Utilizar el objeto writer para **escribir en el archivo**



# Escritura de un fichero CSV

```
import csv
import os
file_name = 'M6_NotasAlumnos.csv'
folder = 'FicherosCSV'
ruta = os.path.join(os.getcwd(), folder)
column_name = ["Apellidos", "Nombre", "Pract01", "Pract02", "Pract03", "NotaFinal"]
data = ["Quintás Ramírez", "Carmen", 6.5, 7, 9, 7.5]
try:
    if not os.path.exists(ruta):
        os.makedirs(folder)

    with open(os.path.join(ruta, file_name), "a", newline="") as file:
        writer = csv.writer(file, delimiter=';')
        writer.writerow(column_name)
        writer.writerow(data)
except FileNotFoundError:
    print("El fichero", file_name, "no existe en la ruta", ruta)
except PermissionError:
    print("No tienes permisos para acceder al archivo", os.path.join(ruta, file_name))
except Exception as e:
    print("Ha ocurrido un error inesperado: {e}")
```

# Otras formas de leer un CSV

- **Uso de csv.DictReader**

- En lugar de usar un csv.reader podemos usar **csv.DictReader**
- **csv.DictReader** permite acceder a los valores de cada fila usando el nombre de la columna como clave.
- **El archivo csv debe tener una primera fila de encabezados** (los nombres de las columnas)
- csv.DictReader retorna un objeto iterable (podemos recorrer todas las líneas del archivo). **Cada línea es un diccionario** desde el que podemos acceder a cada elemento por el nombre de la columna (previamente especificado)

# Ejemplo de uso de csv.DictReader

```
import csv

# En mi caso, el fichero está un directorio por encima del proyecto
with open("../M6_Alumnos.csv", "r") as file:
    file_content = csv.DictReader(file, delimiter=';')

    for row in file_content:
        print('Nombre: ' + row["Nombre"] + ', Apellidos: ' + row["Apellidos"])
```

```
Apellidos;Nombre;Dirección de correo
Bretones López;David;davidbrelop@campus.monlau.com
Calonge Sendra;Ferran Josep;ferrancalsen@campus.monlau.com
Castillejo García;Raül;raulcasgar@campus.monlau.com
Crespo Cambón;Marc;marccrecam@campus.monlau.com
Farré Ruiz;Pau;paufarrui@campus.monlau.com
Farreny Zaragoza;Roger;rogerfarzar@campus.monlau.com
García de los Santos;Hugo;hugogarsan@campus.monlau.com
Jodar Ramos;Martí;martijodram@campus.monlau.com
Lin;Feihong;feihonglin@campus.monlau.com
Martínez Zahiño;Aleix;aleixmarzah@campus.monlau.com
Mas De Xaxars;Martí Antoni;martimasxax@campus.monlau.com
Masó Boltà;Victor;victormasbol@campus.monlau.com
Molina Pomeyrol;Jordi;jordimolpom@campus.monlau.com
Monroy Fernandez;Victor;victormonfer@campus.monlau.com
Morera Navarro;Oscar;oscarmornav@campus.monlau.com
Ortiz Auge;Marti;martiortaug@campus.monlau.com
Ortiz Roque;Marc;marcortroq@campus.monlau.com
Palacios Fabregas;Jonathan;jonathanpalfab@campus.monlau.com
Parralejo Capellera;Joel;joelparcap@campus.monlau.com
Pleguezuelos Cabrera;Axel;axelplecab@campus.monlau.com
Ramos Guixà;Gerard;gerardramgui@campus.monlau.com
Rubel Diaz;Paola;paolarubdia@campus.monlau.com
Salgado Blanco;Victor;victorsalbla@campus.monlau.com
Sanchez Espona;Ivan;ivansanesp@campus.monlau.com
Soria Alvarez;Adan;adansoralv@campus.monlau.com
Tierno Fernandez;Hernan;hernantiefer@campus.monlau.com
Vilanova Paredes;Marc;marcvilpar@campus.monlau.com
Viñals Morales;Oriol;oriolvinmor@campus.monlau.com
```

# csv.DictReader vs csv.reader

- **Ventajas de usar csv.DictReader:**

- Acceder a los valores por el nombre de la columna hace que el **código** sea **más legible**.
- **Si el orden de las columnas cambia**, un código que use DictReader **seguirá funcionando**

- **Inconvenientes de usar csv.DictReader::**

- **Para archivos muy grandes, csv.reader** puede ser algo **más rápido** ya que no tiene que construir un diccionario para cada fila

# Otras formas de escribir un CSV

- Uso de `csv.DictWriter`
  - En lugar de usar un `csv.writer` podemos usar **`csv.DictWriter`**
  - `csv.DictWriter` permite escribir filas en un archivo CSV utilizando diccionarios. Cada clave del diccionario representa el nombre de una columna, y el valor asociado representa el dato de dicha columna
  - Para usar `csv.DictWriter` primero hay que proporcionar una lista de los nombres de las columnas ("fieldnames"). Esta lista determinará el orden en que se escriben las columnas en el archivo CSV
  - Para escribir una fila de encabezados utilizaremos el método **`writeheader()`**

# Otras formas de escribir un CSV

- Uso de `csv.DictWriter`
  - Para añadir datos al archivo CSV, utilizaremos el método **`writerow()`** al que le pasaremos un diccionario con los datos que queremos escribir. Es importante asegurarse de que las claves del diccionario coinciden con los `fieldnames`

# Ejemplo de uso de csv.DictWriter

```
import csv
import os

file_name = 'M6_NotasAlumnos.csv'
folder = 'FicherosCSV'
ruta = os.path.join(os.getcwd(), folder)

# Definiendo los encabezados (nombres de las columnas)
column_name = ["Apellidos", "Nombre", "Pract01", "Pract02", "Pract03", "NotaFinal"]

# Representando los datos como un diccionario
data_dict = {
    "Apellidos": "Quintás Ramírez",
    "Nombre": "Carmen",
    "Pract01": 6.5,
    "Pract02": 7,
    "Pract03": 9,
    "NotaFinal": 7.5
}
```

# Ejemplo de uso de csv.DictWriter

```
try:
    # Crear directorio si no existe
    os.makedirs(ruta, exist_ok=True)

    # Escribir datos en el archivo CSV usando DictWriter
    with open(os.path.join(ruta, file_name), "a", newline="") as file:
        writer = csv.DictWriter(file, fieldnames=column_name, delimiter=';')

        # Escribir encabezados sólo si el archivo está vacío
        if file.tell() == 0:
            writer.writeheader()

        writer.writerow(data_dict)

except FileNotFoundError:
    print("El fichero", file_name, "no existe en la ruta", ruta)
except PermissionError:
    print("No tienes permisos para acceder al archivo", os.path.join(ruta, file_name))
except Exception as e:
    print(f"Ha ocurrido un error inesperado: {e}")
```

exist\_ok por defecto está a False. Si lo ponemos a True conseguimos que en caso de que la carpeta a crear ya exista no se dé un error.

La función tell() devuelve un número entero que representa la posición actual del cursor en el archivo.



# csv.DictReader vs csv.reader

- **Ventajas de usar csv.DictWriter:**

- Usar csv.DictWriter proporciona claridad al código y ayuda a entender la estructura del archivo CSV
- Además, como el orden de escritura se especifica en el campo "fieldnames", nosotros podemos escribir los datos en el orden que queramos especificando las claves del diccionario

- **Inconvenientes de usar csv.DictReader::**

- **Para archivos muy grandes y la estructura de datos es sencilla, csv.writer puede ser algo más eficiente** ya que no tiene que construir un diccionario para cada fila

# PRACTICA

```
Id;Apellidos;Nombre;UF1
1;Bretones López;David;6
2;Calonge Sendra;Ferran Josep;10
3;Castillejo García;Raül;4
4;Crespo Cambón;Marc;7
5;Farré Ruiz;Pau;8
6;Farreny Zaragoza;Roger;5
7;García de los Santos;Hugo;2
8;Jodar Ramos;Martí;7
9;Lin;Feihong;6
10;Martínez Zahiño;Aleix;3
11;Mas De Xaxars;Martí Antoni;1
12;Masó Boltà;Victor;5
13;Molina Pomeyrol;Jordi;0
14;Monroy Fernandez;Victor;3
15;Morera Navarro;Oscar;0
16;Ortiz Auge;Martí;9
17;Ortiz Roque;Marc;7
18;Palacios Fabregas;Jonathan;9
19;Parralejo Capellera;Joel;2
20;Pleguezuelos Cabrera;Axel;0
21;Ramos Guixà;Gerard;0
22;Rubel Diaz;Paola;9
23;Salgado Blanco;Victor;4
24;Sanchez Espona;Ivan;2
25;Soria Alvarez;Adan;4
26;Tierno Fernandez;Hernan;8
27;Vilanova Paredes;Marc;1
28;Viñals Morales;Oriol;6
```



```
Id;Apellidos;Nombre;UF2
1;Bretones López;David;8
2;Calonge Sendra;Ferran Josep;1
3;Castillejo García;Raül;7
4;Crespo Cambón;Marc;4
5;Farré Ruiz;Pau;4
6;Farreny Zaragoza;Roger;3
7;García de los Santos;Hugo;5
8;Jodar Ramos;Martí;2
9;Lin;Feihong;5
10;Martínez Zahiño;Aleix;8
11;Mas De Xaxars;Martí Antoni;10
12;Masó Boltà;Victor;6
13;Molina Pomeyrol;Jordi;6
14;Monroy Fernandez;Victor;8
15;Morera Navarro;Oscar;1
16;Ortiz Auge;Martí;6
17;Ortiz Roque;Marc;0
18;Palacios Fabregas;Jonathan;2
19;Parralejo Capellera;Joel;4
20;Pleguezuelos Cabrera;Axel;0
21;Ramos Guixà;Gerard;4
22;Rubel Diaz;Paola;1
23;Salgado Blanco;Victor;6
24;Sanchez Espona;Ivan;10
25;Soria Alvarez;Adan;1
26;Tierno Fernandez;Hernan;2
27;Vilanova Paredes;Marc;4
28;Viñals Morales;Oriol;4
```



```
Id;Apellidos;Nombre;UF1;UF2
1;Bretones López;David;9;5
2;Calonge Sendra;Ferran Josep;0;6
3;Castillejo García;Raül;5;4
4;Crespo Cambón;Marc;2;1
5;Farré Ruiz;Pau;7;5
6;Farreny Zaragoza;Roger;5;5
7;García de los Santos;Hugo;7;2
8;Jodar Ramos;Martí;0;9
9;Lin;Feihong;5;9
10;Martínez Zahiño;Aleix;10;3
11;Mas De Xaxars;Martí Antoni;7;7
12;Masó Boltà;Victor;6;0
13;Molina Pomeyrol;Jordi;9;8
14;Monroy Fernandez;Victor;10;9
15;Morera Navarro;Oscar;2;6
16;Ortiz Auge;Martí;10;7
17;Ortiz Roque;Marc;5;0
18;Palacios Fabregas;Jonathan;6;4
19;Parralejo Capellera;Joel;1;3
20;Pleguezuelos Cabrera;Axel;2;3
21;Ramos Guixà;Gerard;4;6
22;Rubel Diaz;Paola;1;1
23;Salgado Blanco;Victor;7;0
24;Sanchez Espona;Ivan;2;7
25;Soria Alvarez;Adan;7;9
26;Tierno Fernandez;Hernan;7;6
27;Vilanova Paredes;Marc;6;4
28;Viñals Morales;Oriol;2;1
```

notas\_alumnos\_UF1

notas\_alumnos\_UF2

notas\_alumnos

# PRACTICA

- Crea un programa que lea con `csv.reader/csv.DictReader` los 2 ficheros de entrada (`notas_alumnos_UF1.csv` y `notas_alumnos_UF2.csv`)
- Con la información de cada uno de los ficheros hay que montar un nuevo fichero de salida `notas_alumnos.csv` que contendrá la lista de los alumnos (Id, nombre y apellidos) y las notas tanto de la UF1 como de la UF2.
- Para escribir usa `csv.witer/csv.DictWriter`



Archivo de  
; separados por comas



Archivo de  
; separados por comas