

```
In [31]: from IPython.display import Image
```

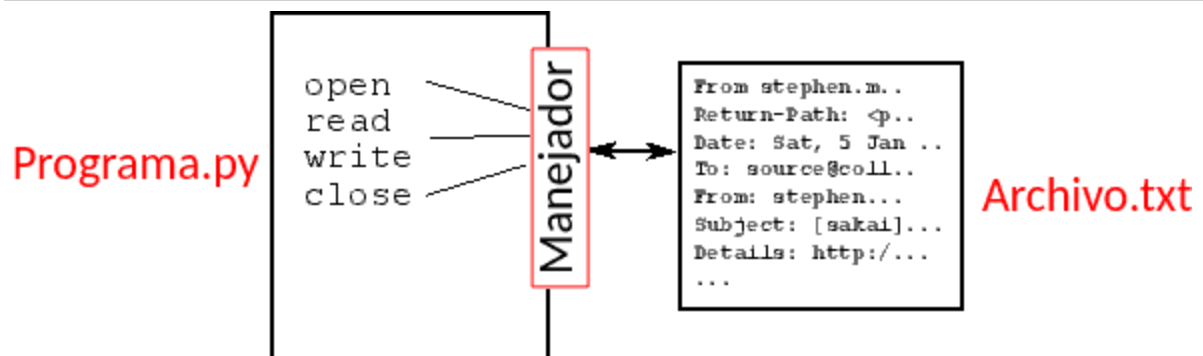
## Archivos

- Que son los archivos
- Tipos de archivos
- Operaciones principales
  - Abrir un archivo
  - Lectura de un archivo
  - Escritura en un archivo
  - Cerrar un archivo
- Iteración sobre las líneas de un archivo
- Otras funciones integradas.
  - `file.readlines`
  - `file.next`

## Conceptos

```
In [32]: Image(filename='images/files.png')
```

Out[32]:



## Qué son

- Almacenan información - datos **persistentes**
  - Almacenamiento en diferentes dispositivos:
    - Disco duro
    - CDs / DVs
    - Internet
    - Memorias USBs
- 
- El programa interactúa con el archivo a través de un **manejador**

---

# Tipos de Archivos

- Archivos tipo Texto:
    - Editables (leíbles)
    - Líneas de texto
      - Cada línea tiene al final un carácter escondido indicando salto de líneas
    - Fácilmente procesables
    - Utilizados para guardar datos de secuencias biológicas:
      - ADN
      - ARN
      - Proteínas
- 
- Binario:
    - No directamente editables o leíbles
    - Estructura solo la entiende el creador del archivo
    - Difíciles de procesar (bloques de bytes)
- 

## Algunos ejemplos de tipos de archivos:

### Tipo Texto:

- .txt: Archivos texto
  - .py : Archivos de python
  - .fasta: Archivos de secuencias en formato FASTA
  - .pdb : Archivos de estructuras de proteínas 3D
- 

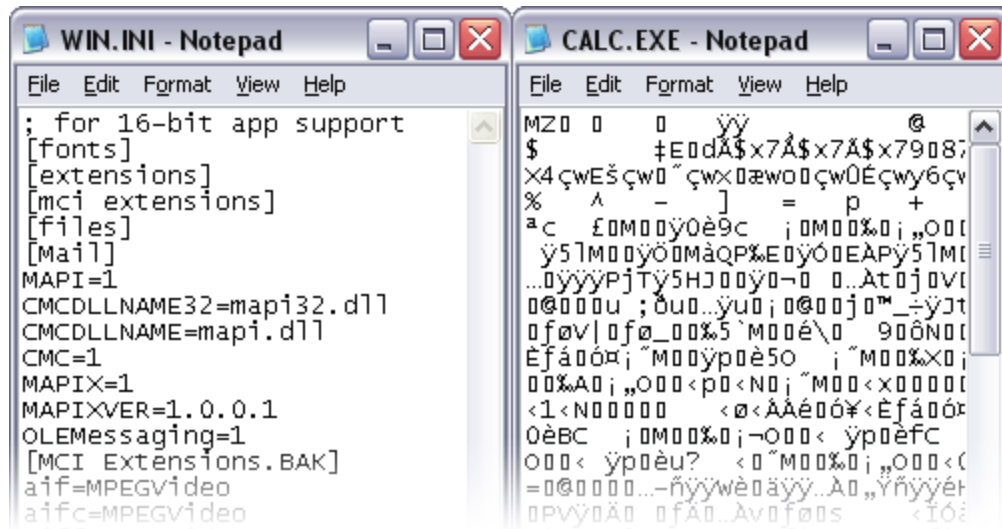
### Binarios

- .mp3: música
- .doc: documentos word
- .png: imágenes
- .exe: archivos ejecutables windos

## Ejemplo archivo texto y archivo binarios

In [33]: `Image(filename='images/file-types.png')`

Out[33]:



## Archivos en Python (file)

- Tipo de datos (o clase): **file**
- Funciones principales:
  - **Abrir:**
    - manejador = open (...)
  - **Leer:**
    - manejador.read (...)
    - manejador.readlines (...)
    - manejador.next (...)
  - **Escribir:**
    - manejador.write (...)
  - **Cerrar:**
    - manejador.close (...)

## Abrir un archivo

manejador = **open** (nombre archivo, **modo**)

- **nombre archivo:**
  - Es una cadena o nombre de un archivo
  - Ejemplo: "secuencias.fasta", "nombres.txt", "resultados.out", "proteina.pdb"
- **Principales modos de apertura de un archivo:**
  - **"r":** (Lectura): Se abre para leer lo que contiene (No se puede modificar)
  - **"w":** (Escritura) Se abre para crear un nuevo archivo (Si existe lo reescribe)
  - **"a":** (Adición) Se abre para adicionar información al final del archivo.

## Ejemplo1: Abrir un archivo para escribir (modo escritura)

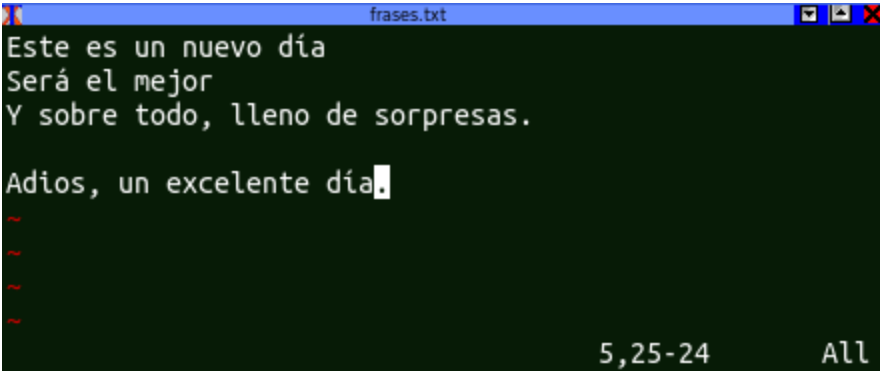
```
In [34]: # Se abre el archivo "hola.txt" para escribir en el
# Retorna una variable o manejador para manipular el archivo.
manejadorHola = open ("hola.txt", "w")
# Ahora se escribe en el archivo una frase
manejadorHola.write ("Hola Mundo")
# Finalmente, se cierra el archivo y termina su uso
manejadorHola.close ()
## Ahora si, revise si se creó su archivo en su carpeta de archivos
```

## Lectura e iteración de un archivo¶

- Se usa la instrucción *for*
- Sintaxis:
  - **for "línea" in "manejador":**

```
In [35]: ### Ejemplo: archivo "frases.txt"
```

```
In [36]: Image(filename='images/frases.png')
```

```
Out[36]: 
```

```
In [37]: # Se abre el archivo
manejador = open ("frases.txt", "r")
# Se itera sobre línea y se la imprime
for línea in manejador:
    print(">>>", línea)
# Se cierra el archivo
manejador.close()
```

```
>>> Este es un nuevo día

>>> Será el mejor

>>> Y sobre todo, lleno de sorpresas.

>>>

>>> Adios, un excelente día.
```

**Ejemplo: Procesamiento del archivo de metagenomas  
"ecoli-metagenome.ptt"**

```

In [38]: listaOperon = []
manejador = open ("ecoli-metagenome.ptt", "r")
for linea in manejador:
    if "operon" in linea:
        listaOperon.append (linea)

manejador.close()
print listaOperon [0]
s = listaOperon [0]
ls = s.split()
print ls
p0 = ls [0]
print p0
nls = p0.split (".")

print nls

mnj2 = open ("ecoli-metagenome.fna", "r")
print mnj2.next ()
adn = mnj2.next()
print adn
ini = int (nls [0])
fin = int (nls [1])
print adn [ini:fin]
print "TGAAACGCATTAGCACCACCATTACCACCACCATCACCATTACCACAGGTAACGGTGCGGGCTGX"

```

```

File "<ipython-input-38-b3292a9889fc>", line 8
    print listaOperon [0]
          ^

```

SyntaxError: Missing parentheses in call to 'print'. Did you mean print(listaOperon [0])?

## Escritura de archivos

- open (nombre archivo, "w")
- write (cadena)

```

In [ ]: mnjW = open ("ejemplo-escritura.txt", "a")
mnjW.write ("PPrimera linea")
mnjW.write ("\n")
mnjW.write ("SSegunda linea\n")
mnjW.write ("SSegunda linea")

mnjW.close ()

```

```

In [ ]: ## Escribe solo las líneas que tenga la palabra "operon"
mjEscritura = open ("operon.data", "w")
for linea in listaOperon:
    mjEscritura.write (linea)

mjEscritura.close()

```

# Ejemplos

## Procesar un archivo de estructura de proteína (.pdb)

### Inicio

```
In [ ]: mnjIn = open ("insulin-atoms.pdb")

listaLineas = mnjIn.readlines()

s0 = listaLineas[0]
print l0
l0 = s0.split ()
x = l0 [6]
y = l0 [7]
z = l0 [8]

print x, y, z

l0 [6] = round (float (l0 [6]) - 10, 2):q
l0 [7] = round (float (l0 [7]) - 10, 2)
l0 [8] = round (float (l0 [8]) - 10, 2)

print s0
print l0
mnjIn.close ()
```

```
In [ ]: def procesarLineas (linea):
    l0 = linea.split ()
    l0 [6] = str (round (float (l0 [6]) - 10, 2))
    l0 [7] = str (round (float (l0 [7]) - 10, 2))
    l0 [8] = str(round (float (l0 [8]) - 10, 2))

    return l0

mnjIn = open ("insulin-atoms.pdb")
mnjOut = open ("insulin-nueva.pdb", "w")
listaLineas = mnjIn.readlines()

for linea in listaLineas:
    nlinea = procesarLineas (linea)
    lineaFormato = "\t".join (nlinea)
    mnjOut.write (lineaFormato)
    mnjOut.write ("\n")

mnjIn.close ()
mnjOut.close()
```