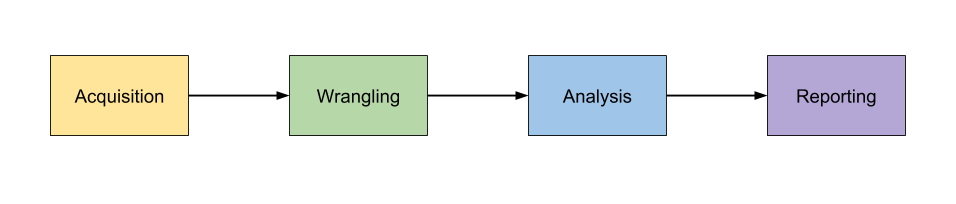
**Pipeline**



Hay que comenzar con el final en mente ¿que entregable tenemos que desarrollar?

* **Acquisition:** necesita **adquirir** los datos de una **fuente de datos**.
* **Wrangling:** **limpiar** y **preparar** los datos.
* **Analysis:** aplicar **métodos analíticos** y realizar las **agregaciones necesarias**.
* **Reporting:** comunicar los hallazgos a las partes interesadas adecuadas a través de informes y visualizaciones (csv, BI, …)

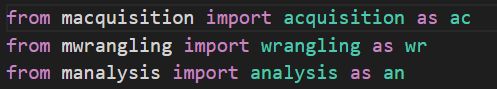
→ Cada etapa va a tener un “file.py” compuesto generalmente de funciones.

***CARPETAS:***

Podemos tener un único main.py o diferentes módulos que serán importados en el main.py

→ jupyter o **visual studio** para crearlos (lo importantes esq sean .py)

* notebooks (para almacenar notebooks de trabajo)
* data (para almacenar csv, excels, etc)
* modules
  + **acquisition** (con su file.py)
  + **wrangling** (con su file.py)
  + **analysis** (con su file.py)
* **file.py** (main.py), en el que **importaremos los otros files.py** y los ejecutaremos



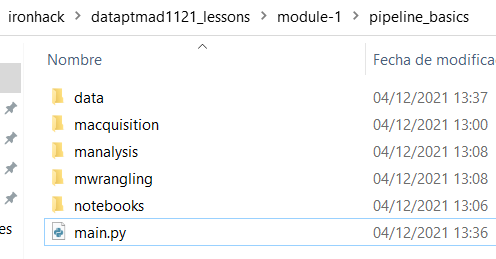
if \_\_name\_\_ == '\_\_main\_\_' después de los imports. Es la declaración en la parte inferior **donde realmente se ejecutan las funciones**. Esta declaración debe agregarse cuando hay código que queremos que el intérprete de Python ejecute solo cuando **se ejecuta un script como programa principal** (no se importa a otro archivo de Python).

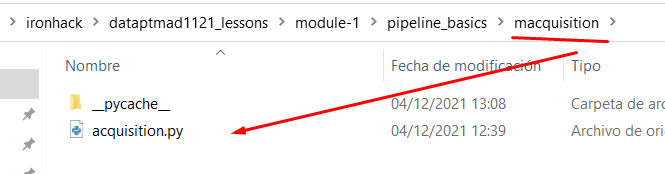
Lo podremos ejecutar en la terminal (cuidado con estar en la librería adecuada)



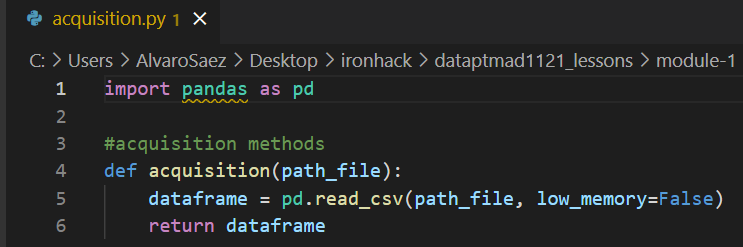
**EJEMPLO con DIFERENTES módulos + main.py:**

1. *estructura de carpetas y archivos*

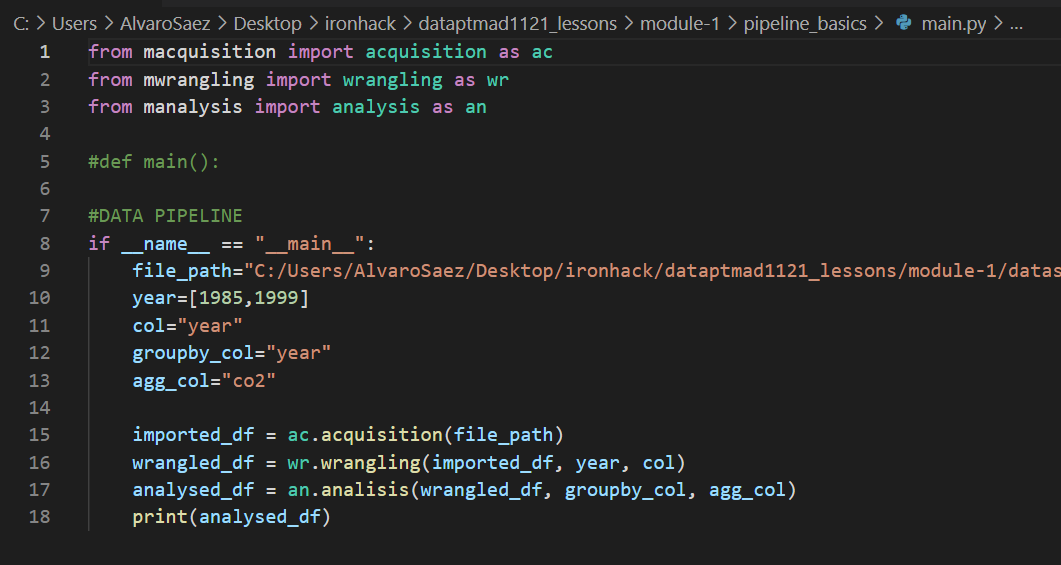




1. *cada archivo tendrá su función*



1. *main.py:* en el que importaremos los otros files.py y los ejecutaremos



**EJEMPLO con un ÚNICO main.py:**