

## Ejercicios de listas y tuplas

1. Escribir un programa que guarde en una lista los nombres de las materias de primer año y las muestre por pantalla.
2. En el programa anterior definir la variable nombre y usando for imprimir tantos mensajes como materias haya en la lista: "Mi nombre es <nombre> y curso <matéria>".
3. Escribir un programa que guarde una lista de materias rendidas con su nota. Cada elemento de la lista es una tupla de dos elementos (nombre, nota). Ejemplo: ("LEDD", 10).
4. En el programa anterior definir la variable nombre y usando for imprimir tantos mensajes como materias haya en la lista: "Mi nombre es <nombre> y aprobé <matéria> con <nota>".
5. Escribir un programa que le pida al usuario por teclado las materias que ya rindió y la nota obtenida. Guardar la información en una estructura como en la del punto anterior y mostrar los mismos mensajes. Tip: usar `while` e `input`.

## Ejercicios de conjuntos

1. Definir un conjunto con los primeros diez enteros divisibles por cinco. Definir un conjunto con los primeros quince enteros divisibles por siete. Realizar las operaciones de unión, intersección y diferencia simétrica (en ambos sentidos) entre ambos conjuntos.
2. Dadas la listas:

```
autores_argentinos = ["Borges", "Saer", "Cortazar",  
"Piglia", "Kohan", "Fogwill", "Casas"]  
autores_extranjeros = ["Joyce", "Poe", "Child", "Carver",  
"Chandler", "Coetze", "Auster"]  
autores_vivos = ["Casas", "Kohan", "Child", "Coetze",  
"Auster"]
```

Encontrar la lista autores argentinos vivos. ¿Qué operación usó?

Encontrar la lista de autores muertos. ¿Qué operación usó?

## Ejercicios de diccionarios

1. Definir un diccionario de materias de primer año en el que la clave sea el nombre y el valor la nota obtenida. Si la materia no se rindió, no guardarla.
2. Escribir un programa que utilice el diccionario anterior y que le permite al usuario consultar por teclado la nota de una materia. Si la materia no existe, mostrar un mensaje indicando que aún no se rindió.
3. Modificar el programa anterior para que al inicio le pregunte al usuario qué desea hacer, si consultar notas o cargar notas. Si elige consultar, el programa debe funcionar como en el punto anterior. Si elige cargar, debe permitirle al usuario ingresar una materia y a continuación una nota. Permitir que el usuario cargue notas para tantas materias como quiera hasta que ingrese una cadena vacía como nombre de materia. Si el usuario ingresa una materia que ya existe en el diccionario, la nota se debe actualizar.