15/05/2023 Davis Steck Landazuri

Resumen Estructuras de datos en Python. Uso, manejo y ventajas. Listas, tablas y diccionarios

La estructura de datos en Python se puede entender como un tipo de datos complejo porque podemos almacenar una estructura completa con información en la misma variable. Estas estructuras pueden tener diferentes características y funciones. En realidad, hay varios tipos de estructuras de datos en Python.

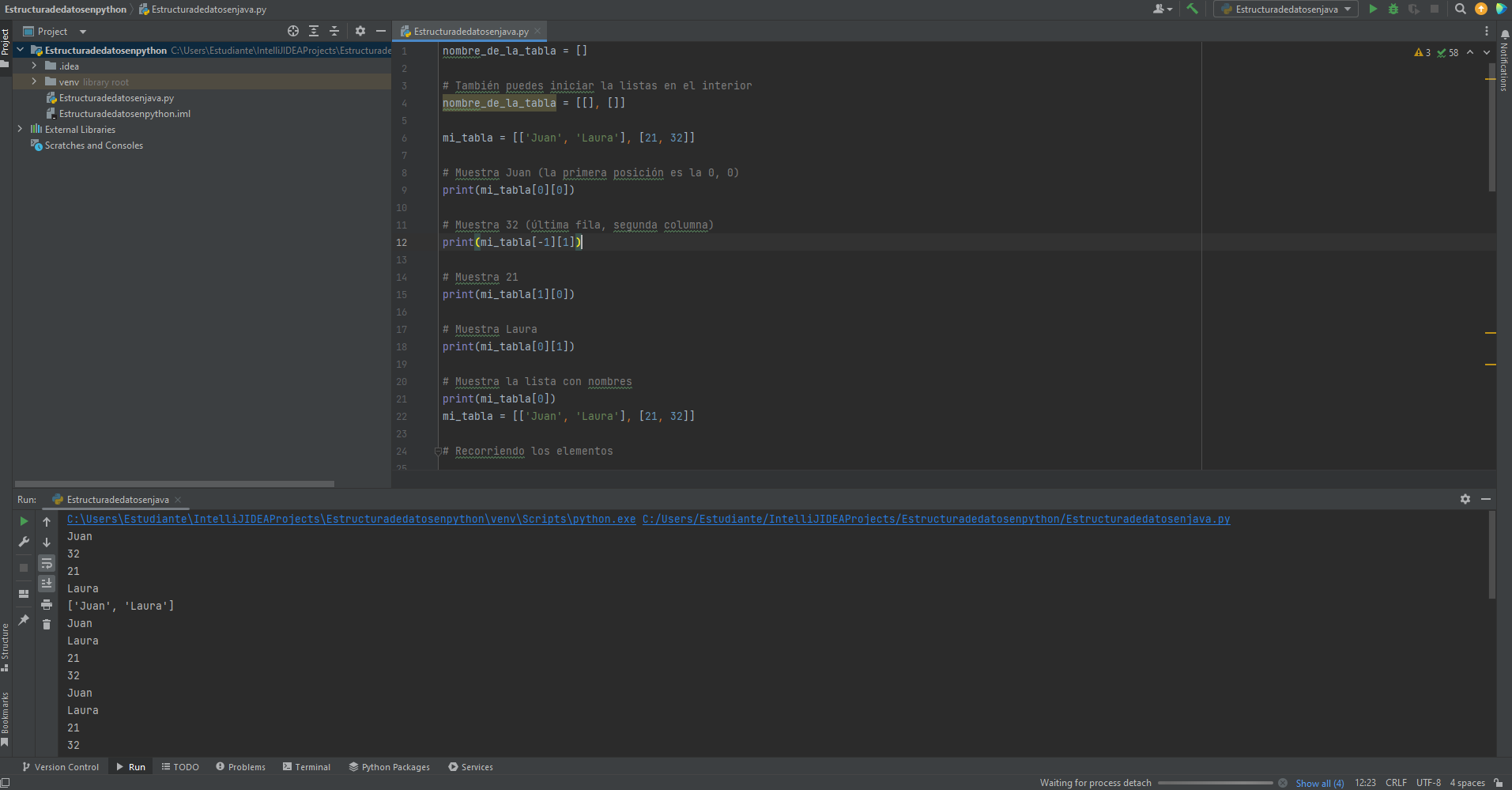
Las estructuras de datos más comunes en Python son listas, tablas y diccionarios. Aunque tienen un nombre diferente, en otros lenguajes son arreglos o vectores, matrices y arreglos indexados, respectivamente. Son básicamente los mismos que los habituales en Python, con varias funciones ya incluidas.

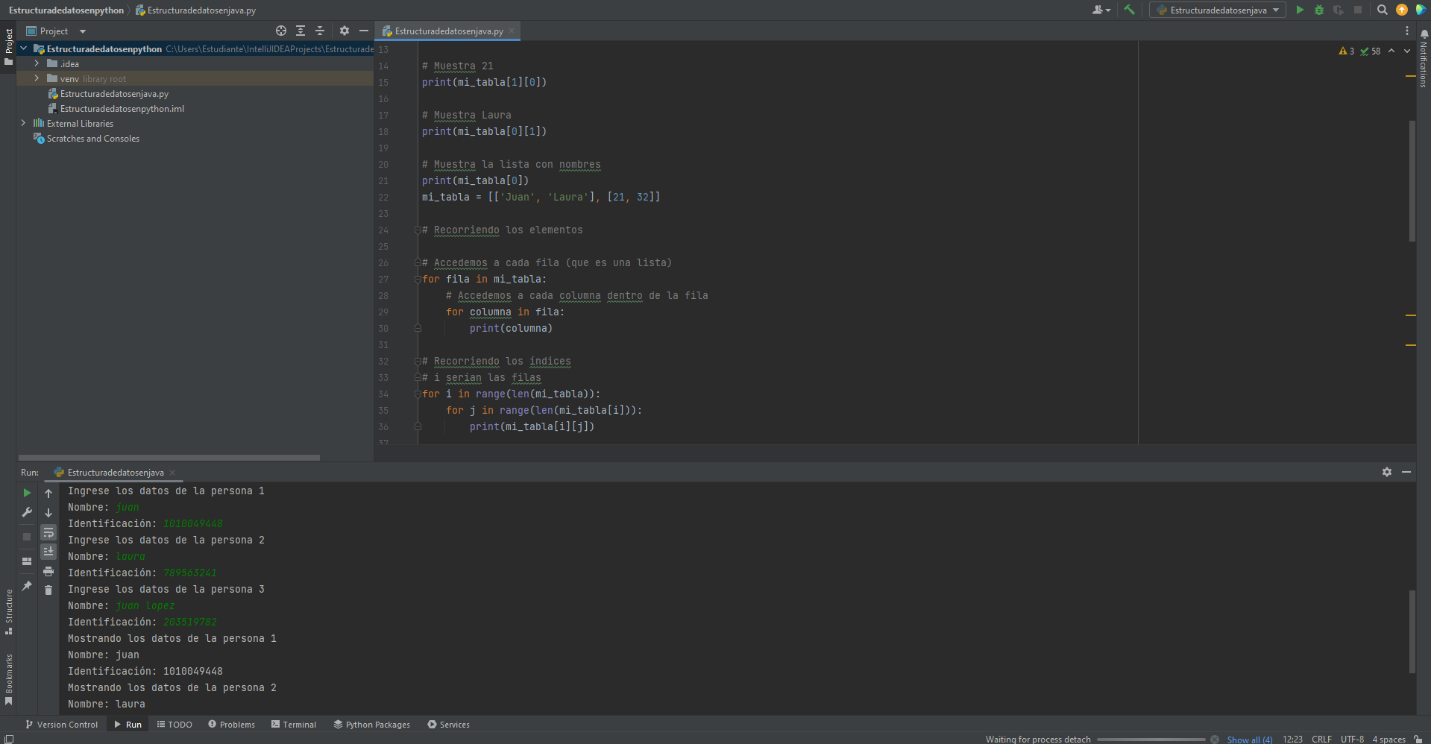
En esta serie de contenidos vamos a explorar cada una de esas estructuras de datos en Python (listas, tablas, diccionarios y tuplas).

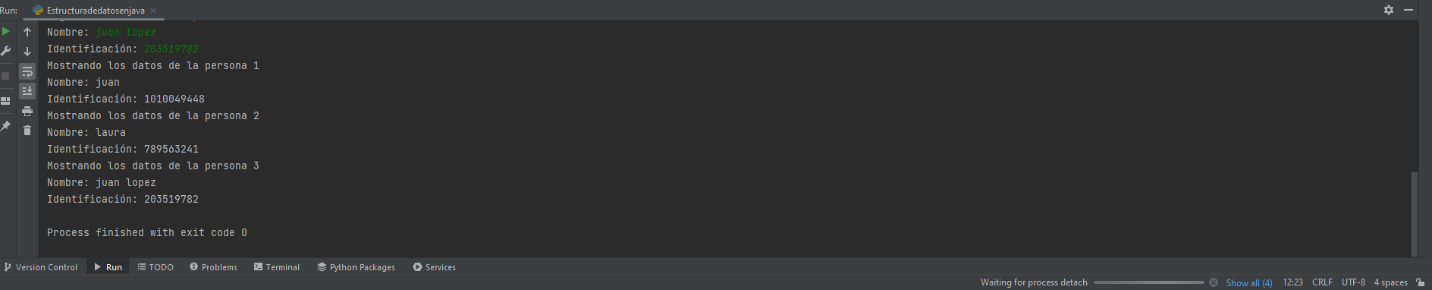
Las estructuras de datos se crearon para resolver una variedad de problemas que sabemos que hasta ahora no podían, porque le permiten agrupar fácilmente un grupo de datos (a menudo relacionados) para que sean fáciles de operar. Cosas como ordenar, agregar, eliminar, mostrar, mover y otras operaciones son posibles y fáciles en Python.

Por ejemplo, sin la existencia de una estructura de datos sería imposible o muy difícil conocer y procesar todos los números de identificación, nombres y direcciones de todos los usuarios del sistema (generalmente muchos), pero ahora con la estructura de datos es muy fácil definir un grupo de elementos para cada valor que queremos almacenar y acceder a ellos con una sola variable. No tendría sentido crear 500 variables para contener los nombres de 500 usuarios.

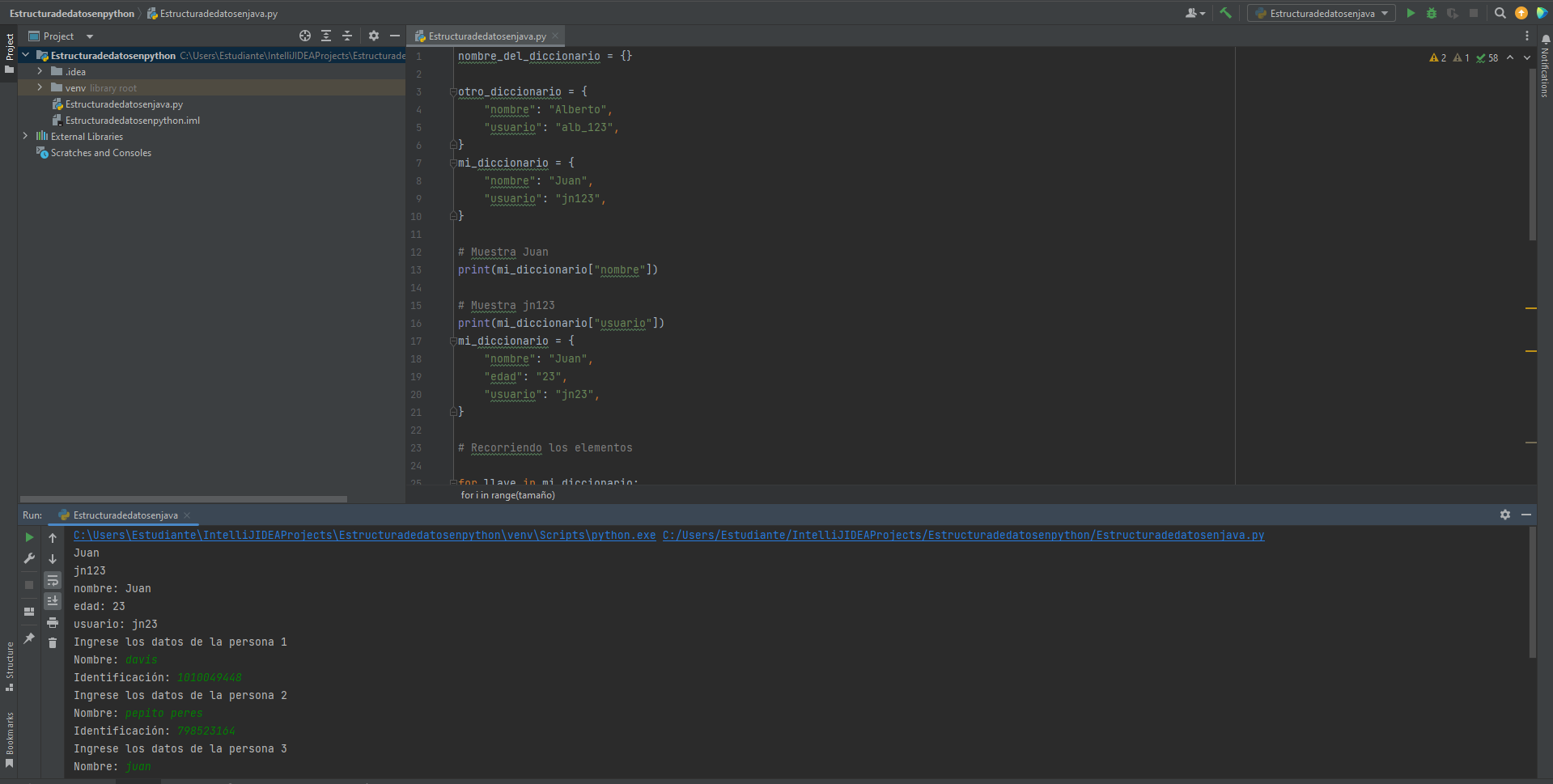
Cabe aclarar que en cada una de las capturas de pantalla de evidencia de su correcta función se dan las descripciones detalladas de los pasos.

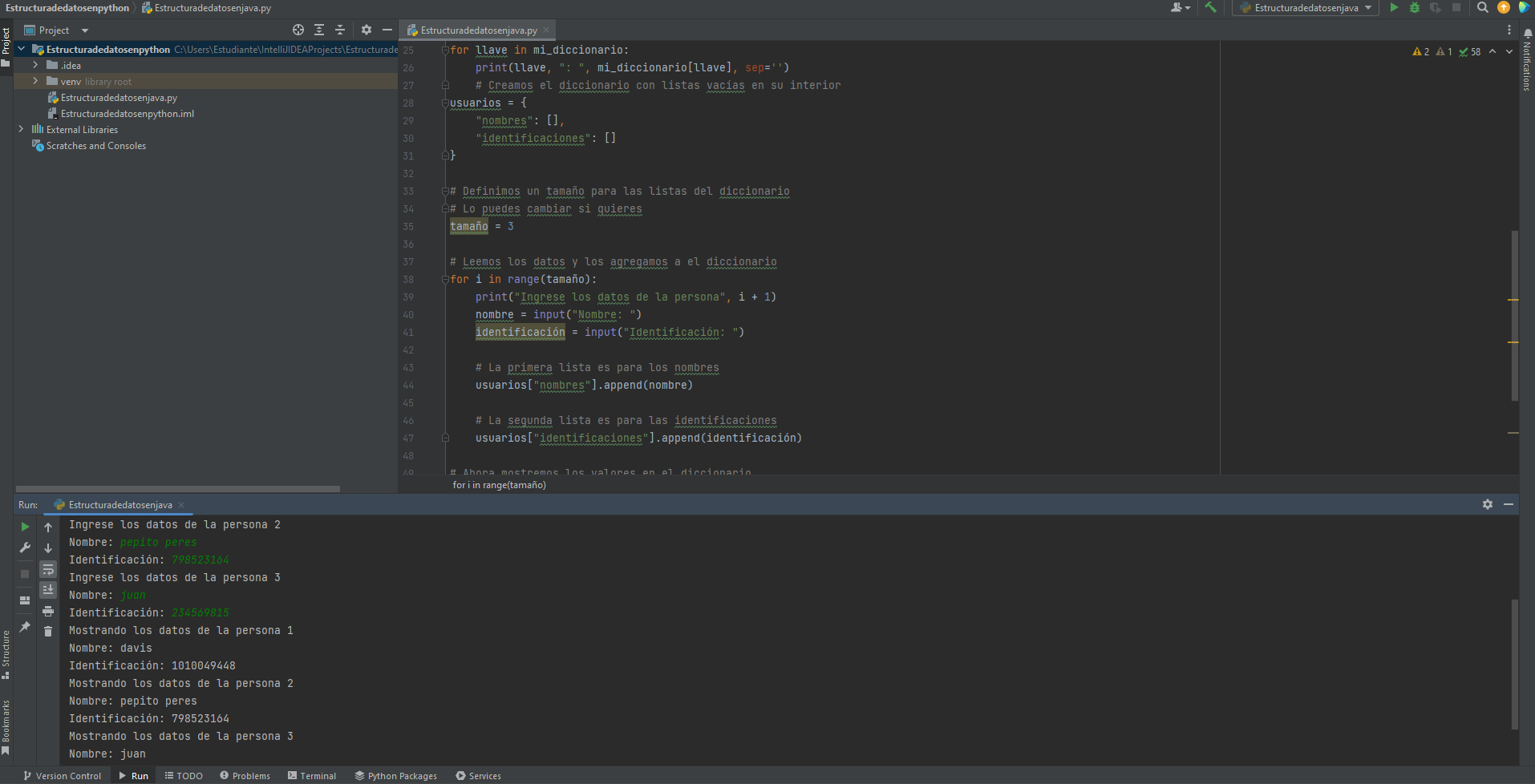
Tablas:

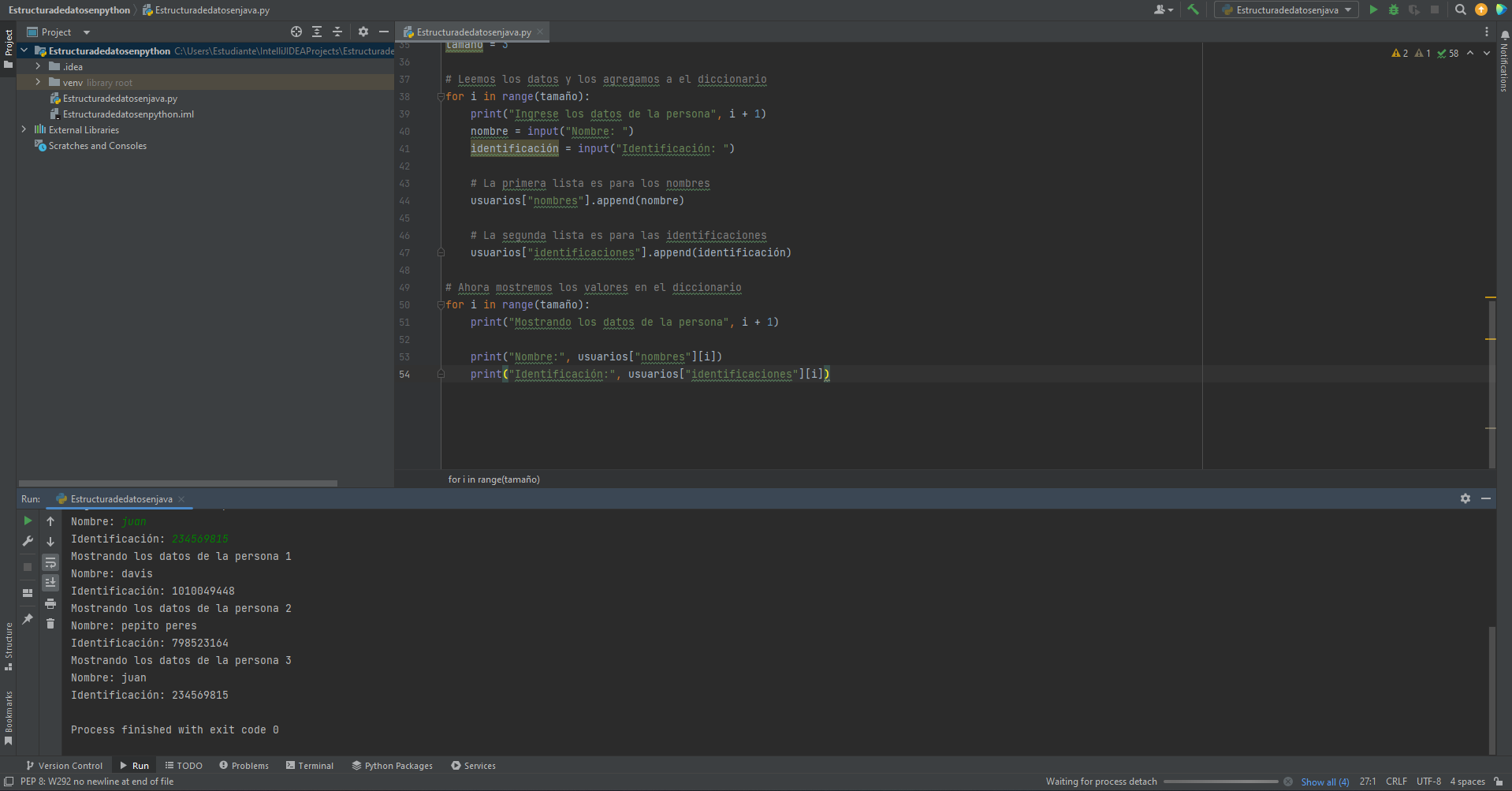


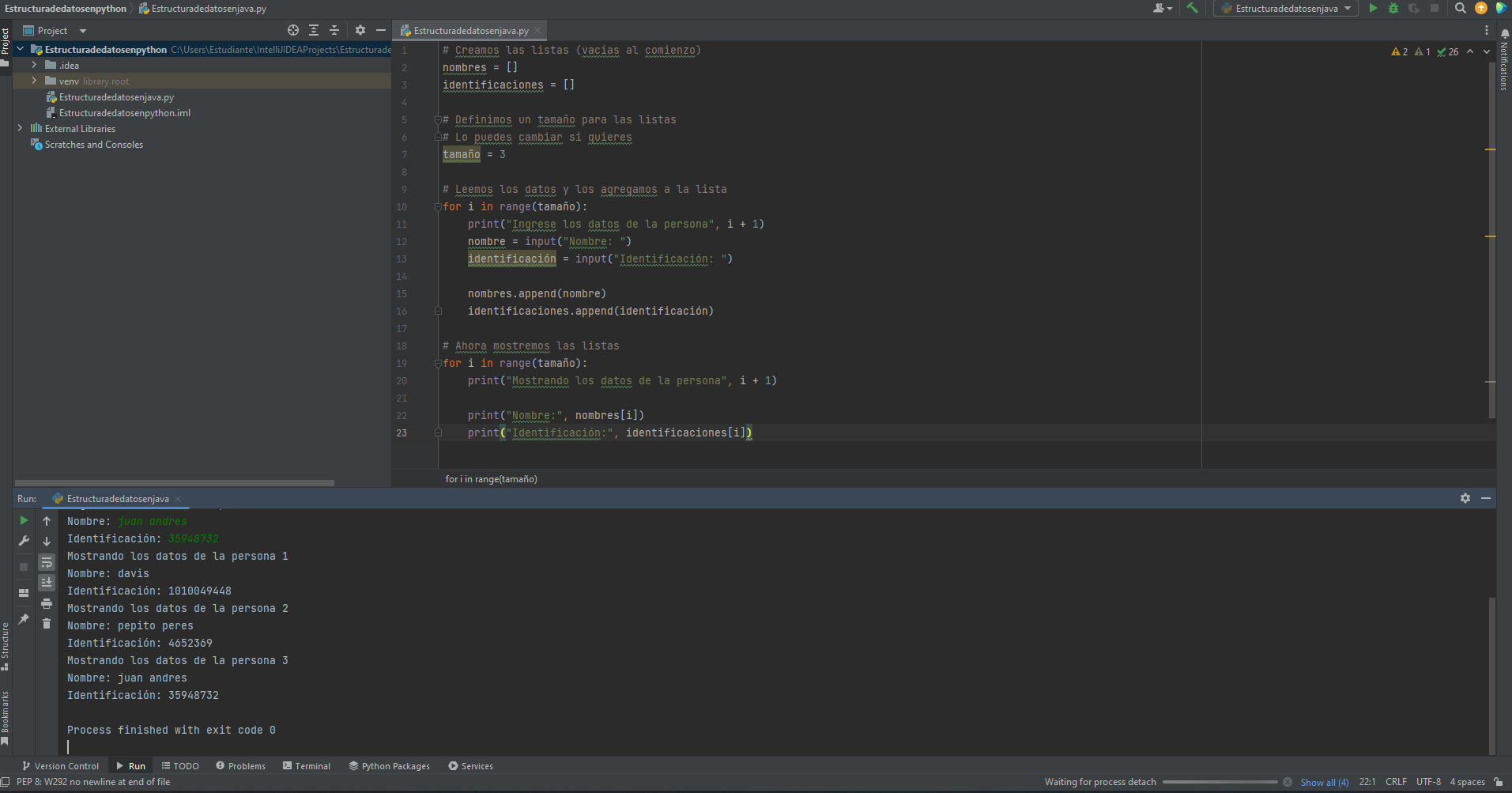


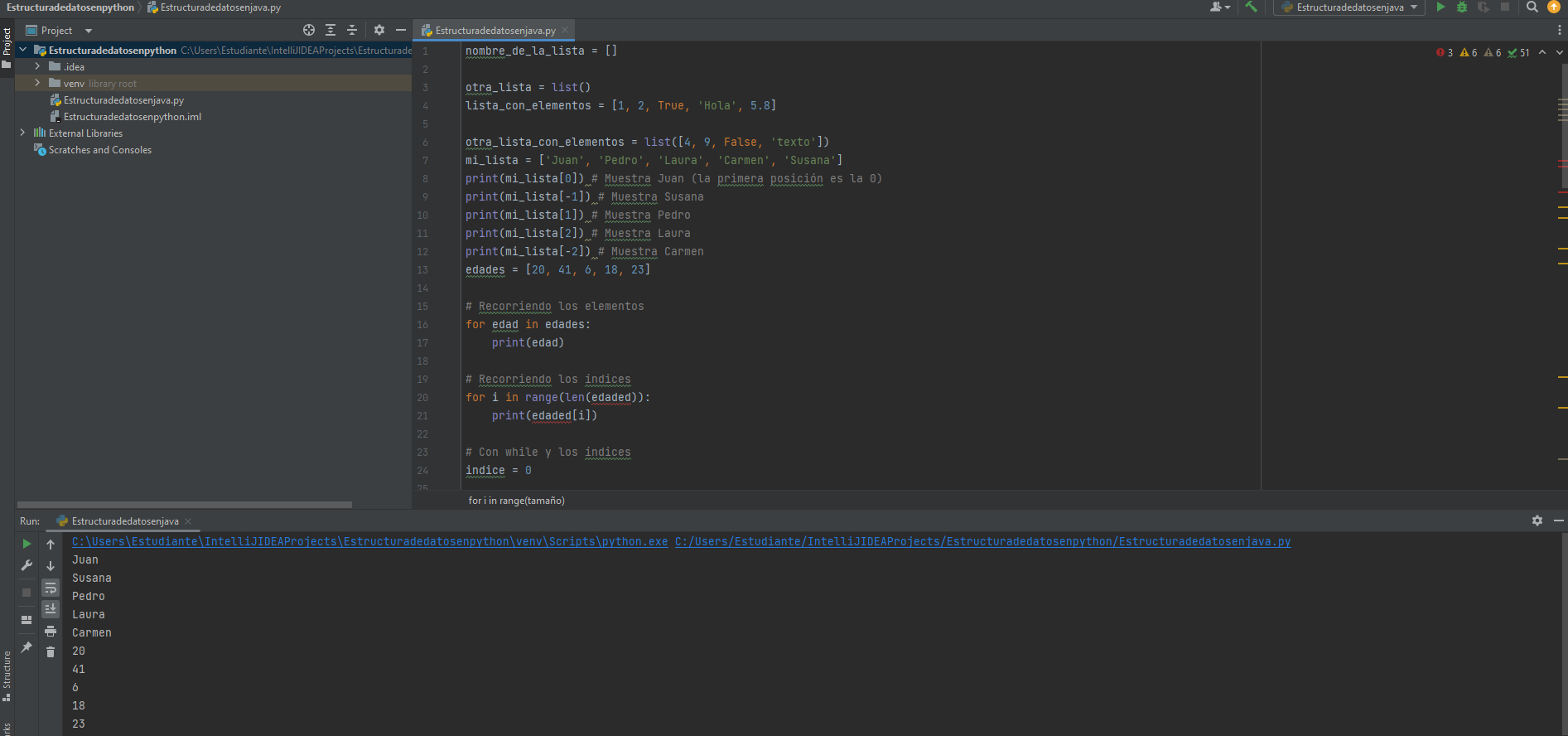
Diccionario:

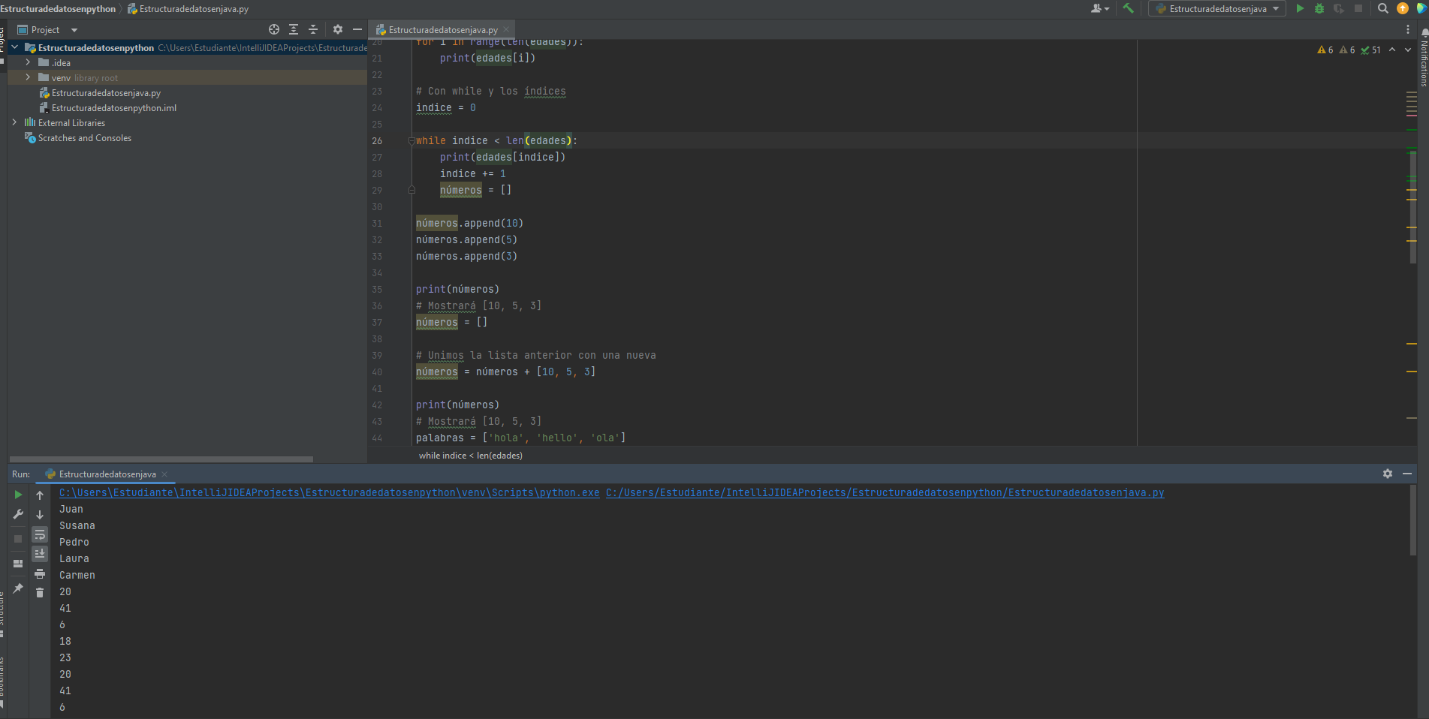


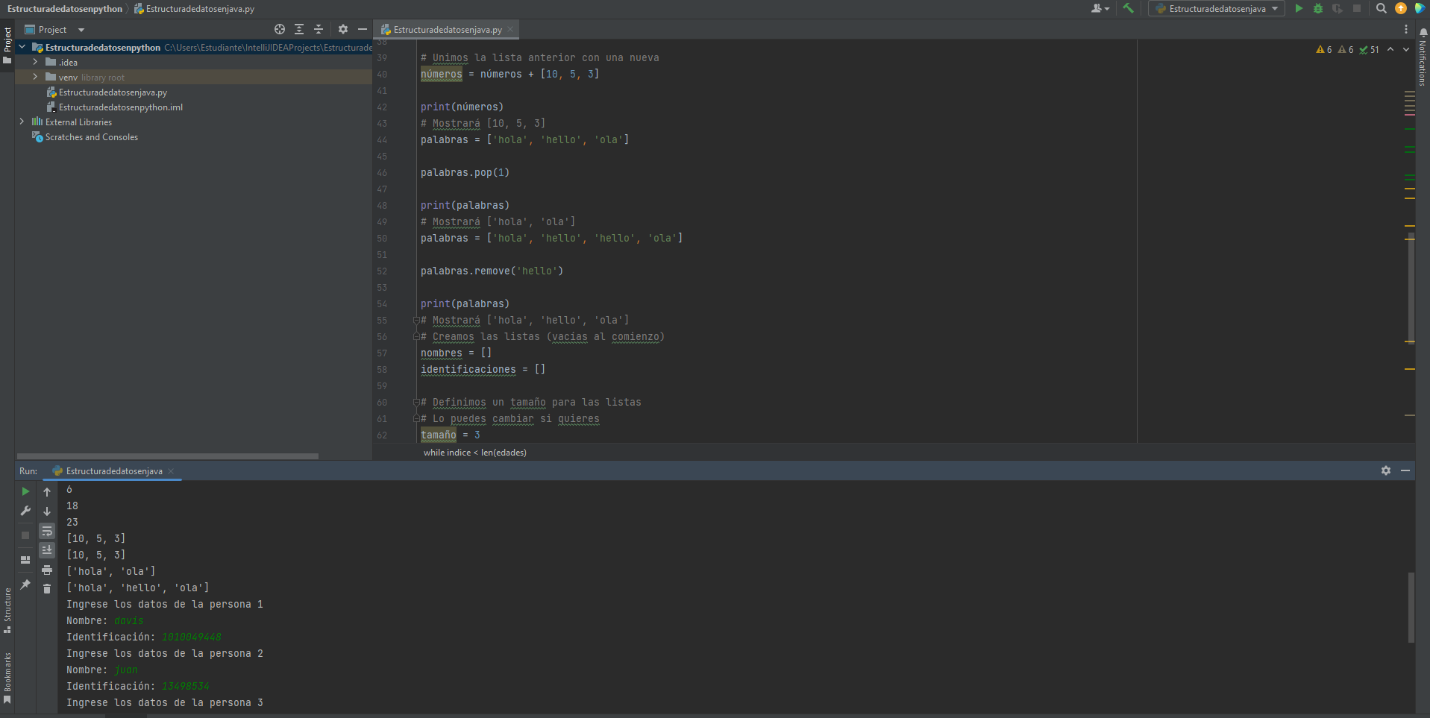


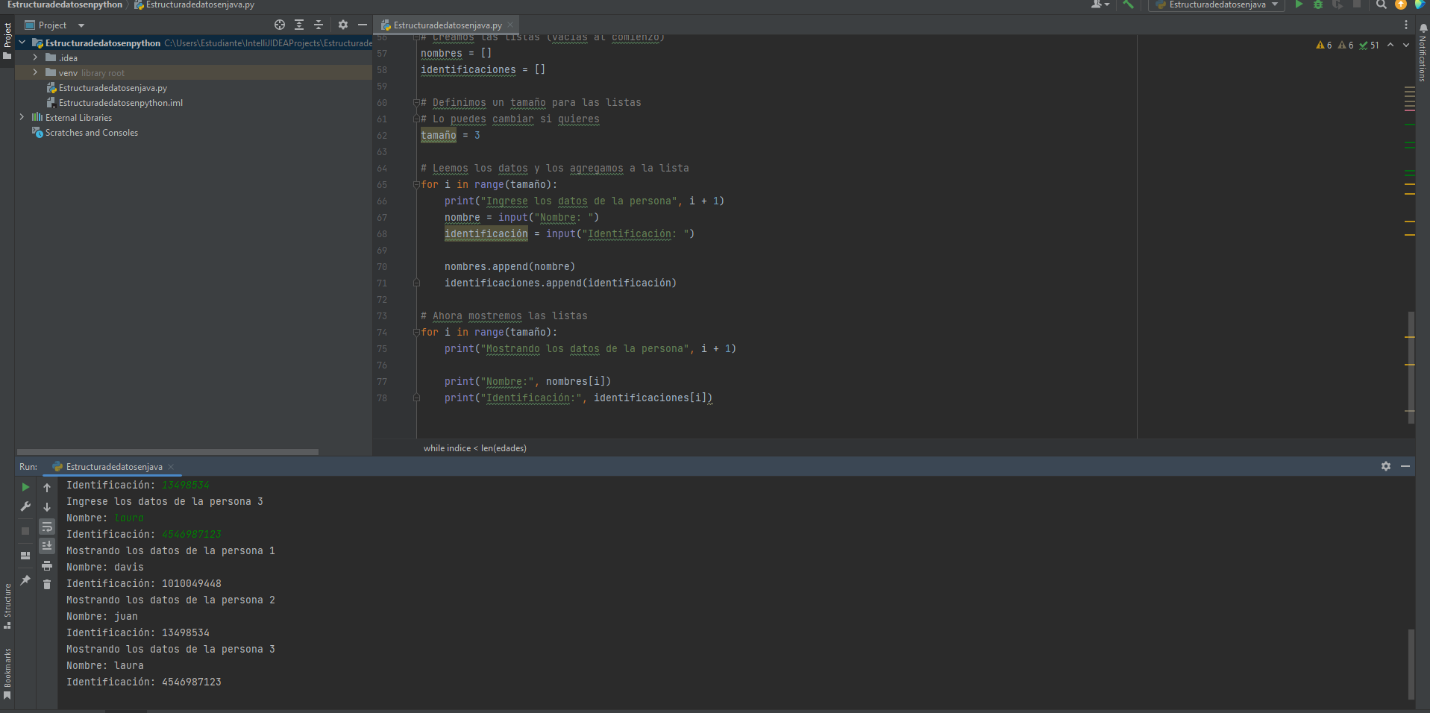


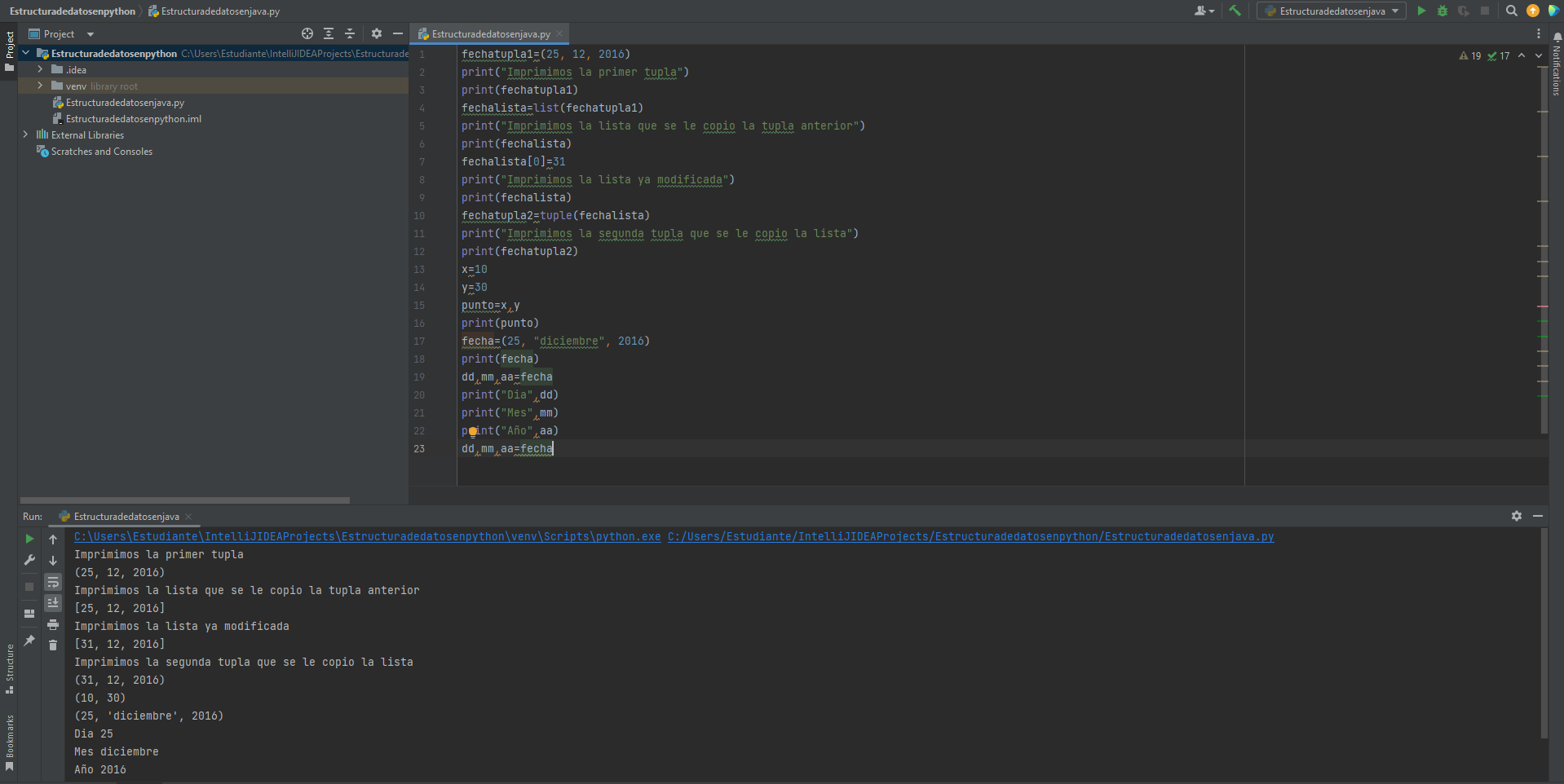
Listas:







Tupla:



Empaquetado y desempaquetado de tuplas.

Podemos generar una tupla asignando a una variable un conjunto de variables o valores separados por coma:

x=10

y=30

punto=x, y

print(punto)

tenemos dos variables enteras x e y. Luego se genera una tupla llamada punto con dos elementos.

fecha= (25, "diciembre", 2016)

print(fecha)

dd, mm, aa=fecha

print("Dia",dd)

print("Mes",mm)

print("Año",aa)

El desempaquetado de la tupla "fecha" se produce cuando definimos tres variables separadas por coma y le asignamos una tupla:

dd,mm,aa=fecha

Es importante tener en cuenta de definir el mismo número de variables que la cantidad de elementos de la tupla.