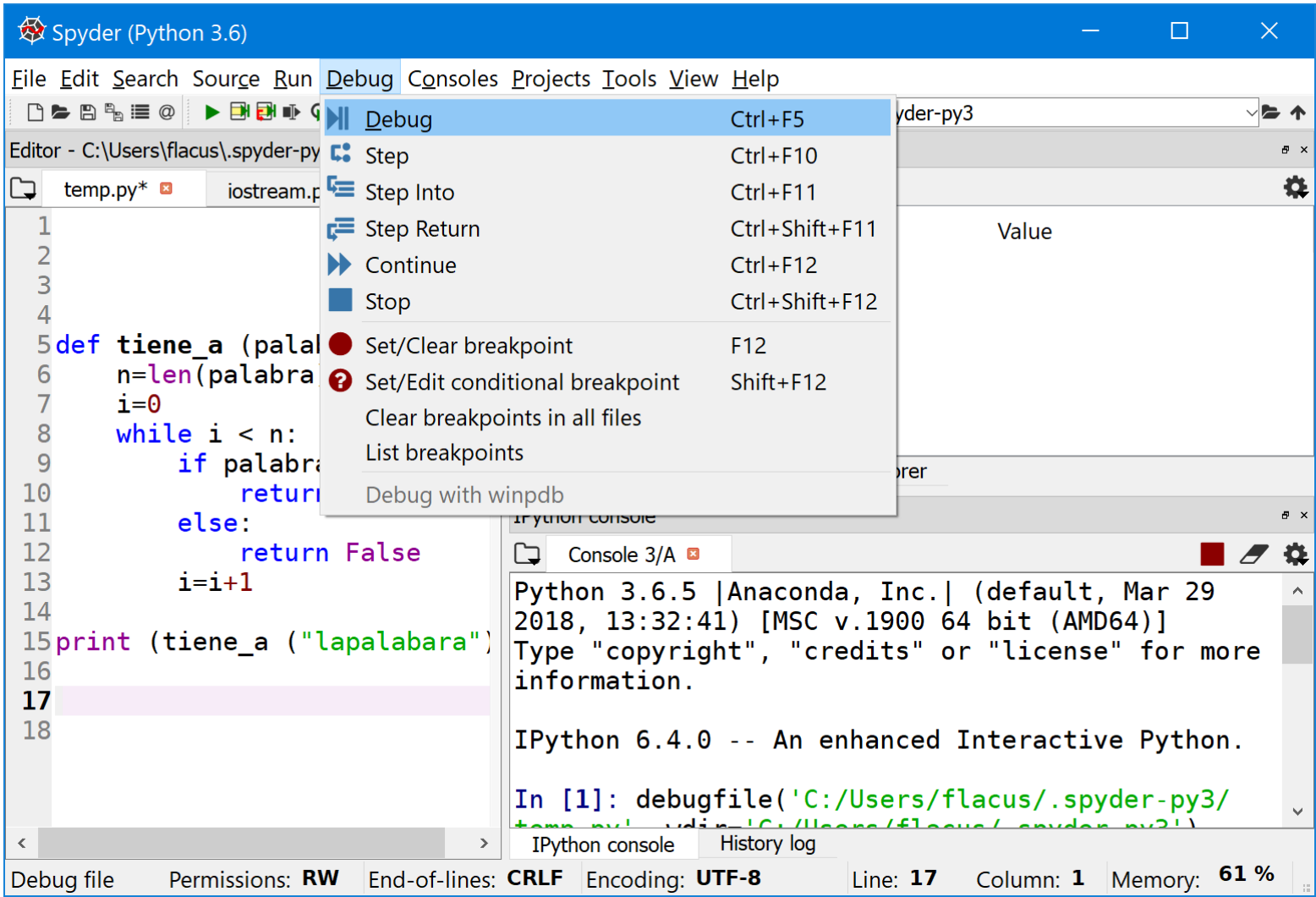
Este es un breve "manual" sobre cómo usar Spyder para hacer debugging de un programa.

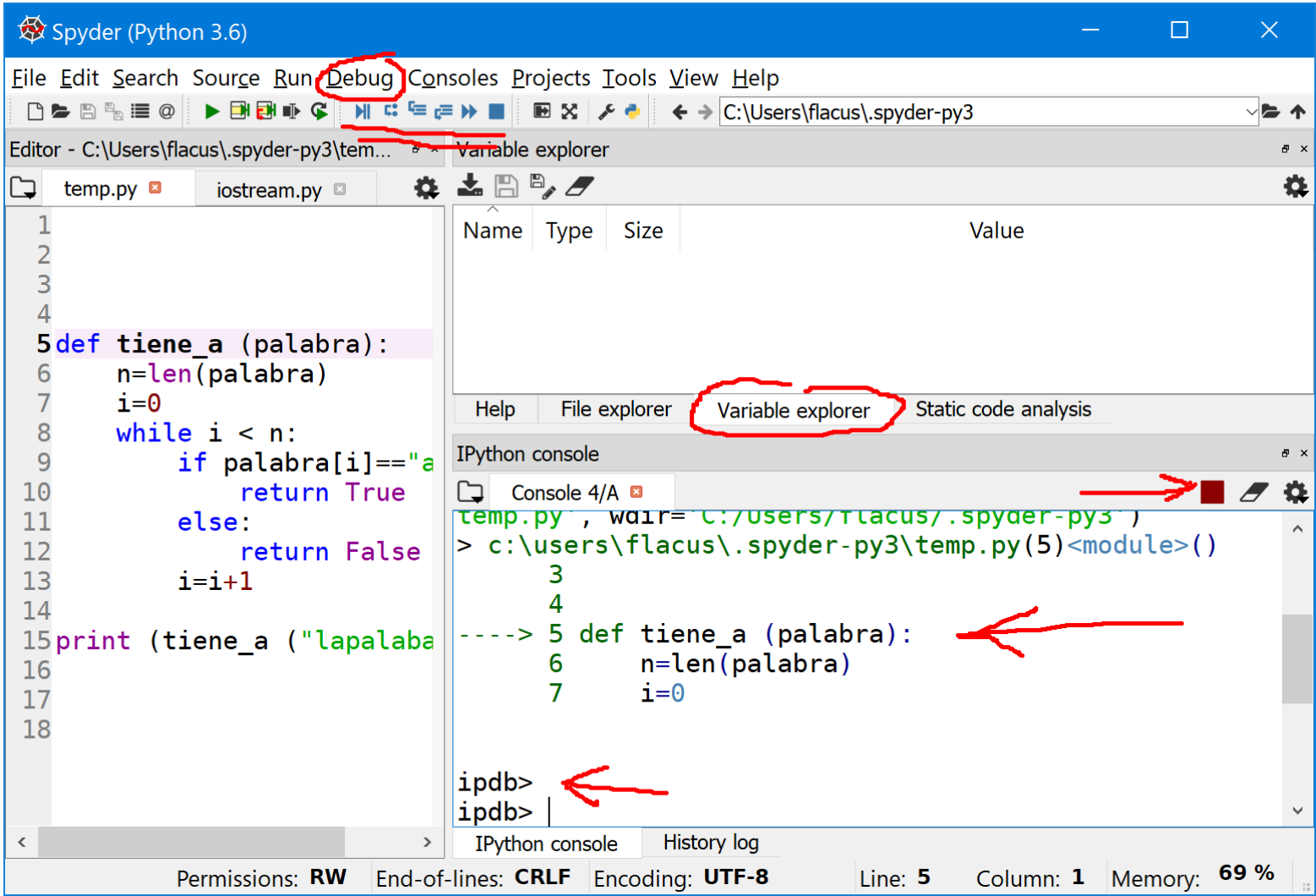
Primero entramos al modo DEBUG: (Ctrl+F5) El programa queda pausado antes de comenzar. Luego pedimos Step (ejecutar línea)/ Step into (entrar) / Step return (ejecutar hasta salir), comparando el estado esperado de las variables con su estado real, PREDICIENDO lo que ocurrirá.



Queremos ver la solapa Variable Explorer (arriba derecho). Sabemos que el programa está en ejecución pero pausado por el "Stop" rojo de la derecho. Sabemos que estamos en modo DEBUG por el prompt "ipdb" abajo

Step ejecuta una línea. Step Into entra en la función referida. Step return ejecuta hasta salir de la función en la que está.

Vamos a debuggear el Código del ejercicio 11 de la clase 1 (el primero de los tres).

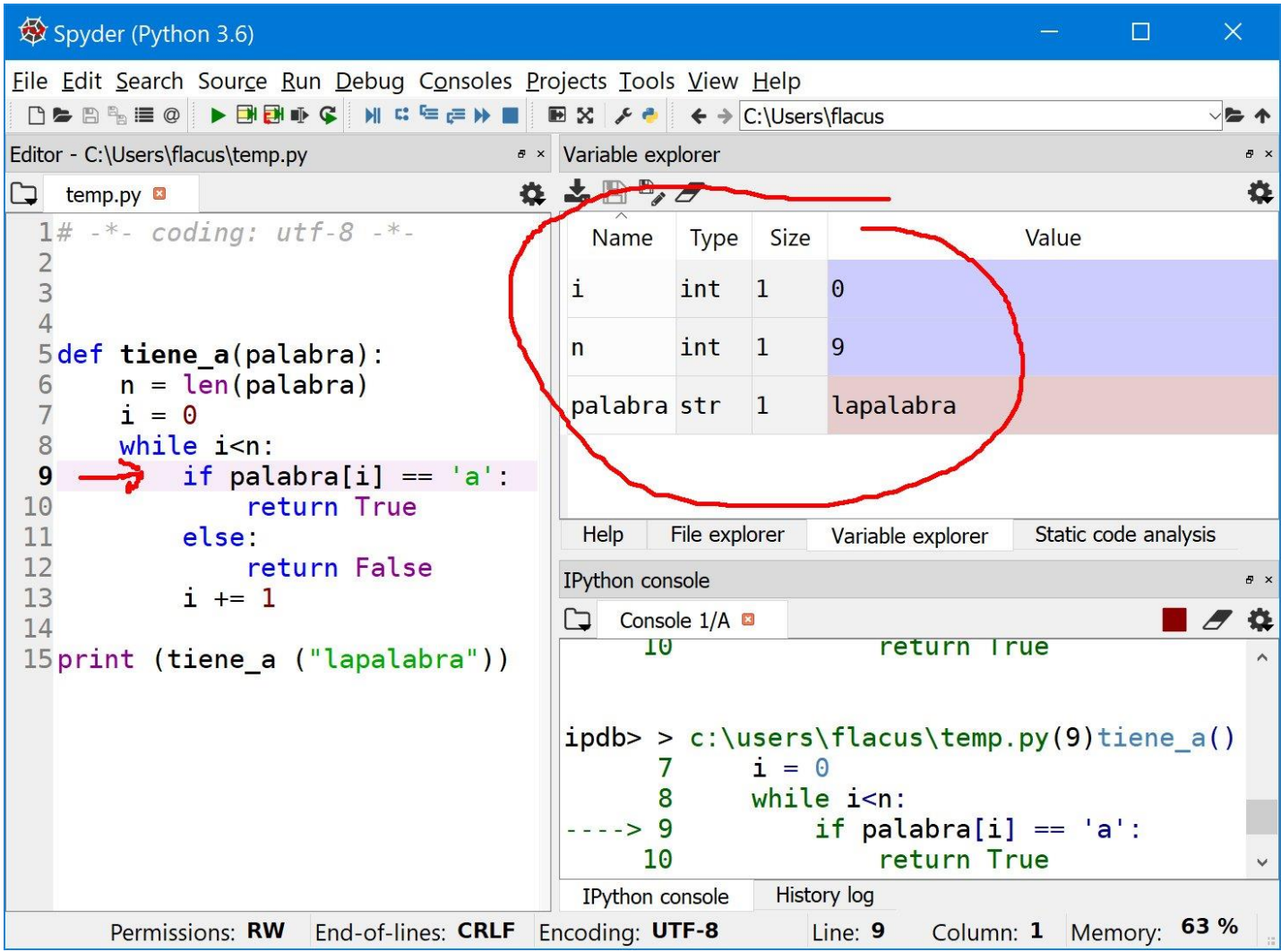


Pedimos algunos STEP Into hasta llegar a la línea 9. Vemos que todas las variable internas de la función están definidas y con sus valores.

Como i=0 sabemos que es la primera iteración. Corroboramos que n=9 (“lapalabra” tiene 9 letras)

En este punto se evalúa el IF, y saltaremos a alguna de las dos ramas de ejecución.

Pedimos un Step.

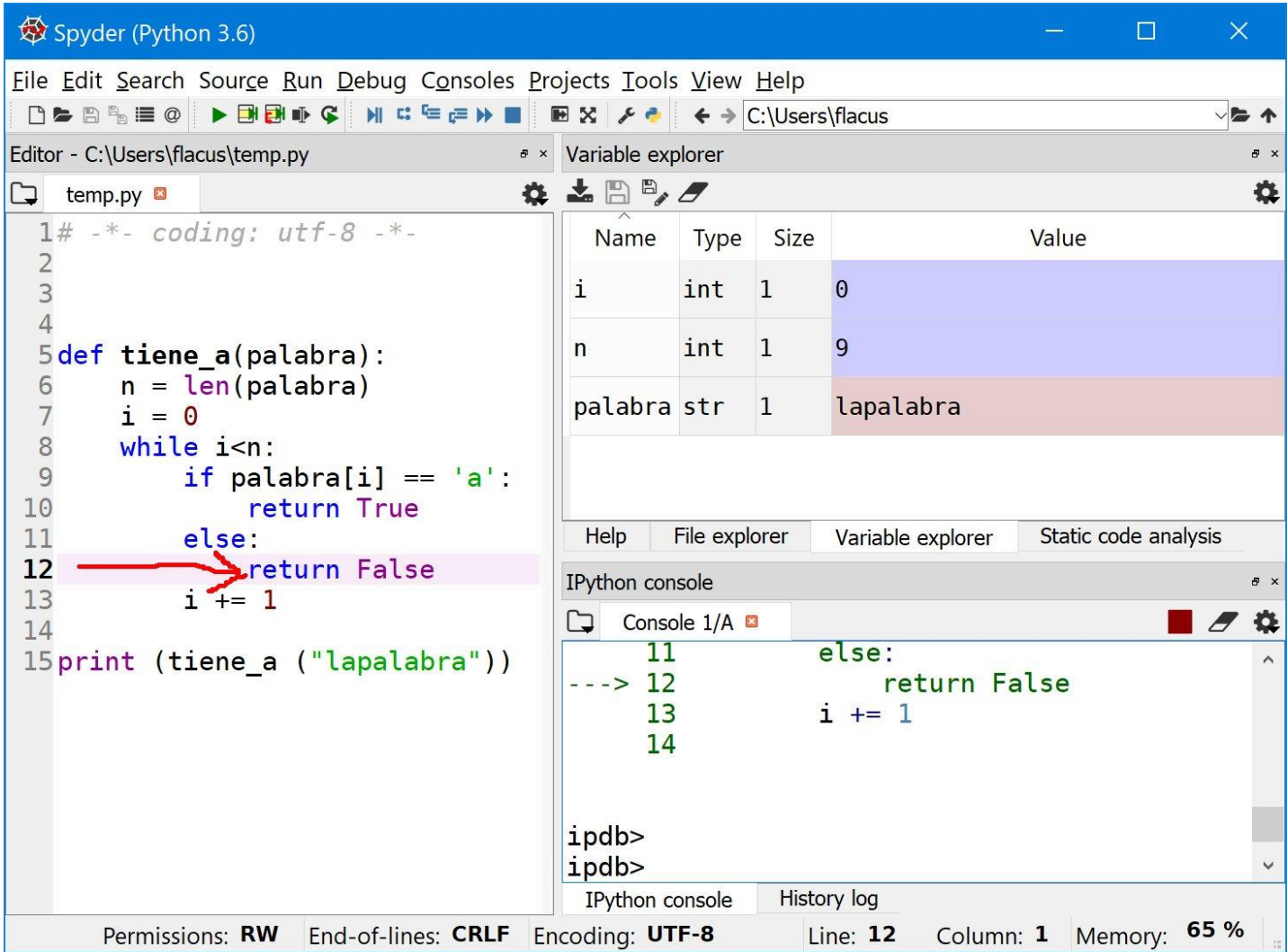


El IF result FALSO, lo que concuerda con lo esperado porque palabra[1]="l" != "a" (la primera letra) no es "a".

La próxima instrucción será RETURN !!!

Pero si es Return saldremos de la function y aún no evaluamos mas que la primera posición. Esto es lo que deseamos ?

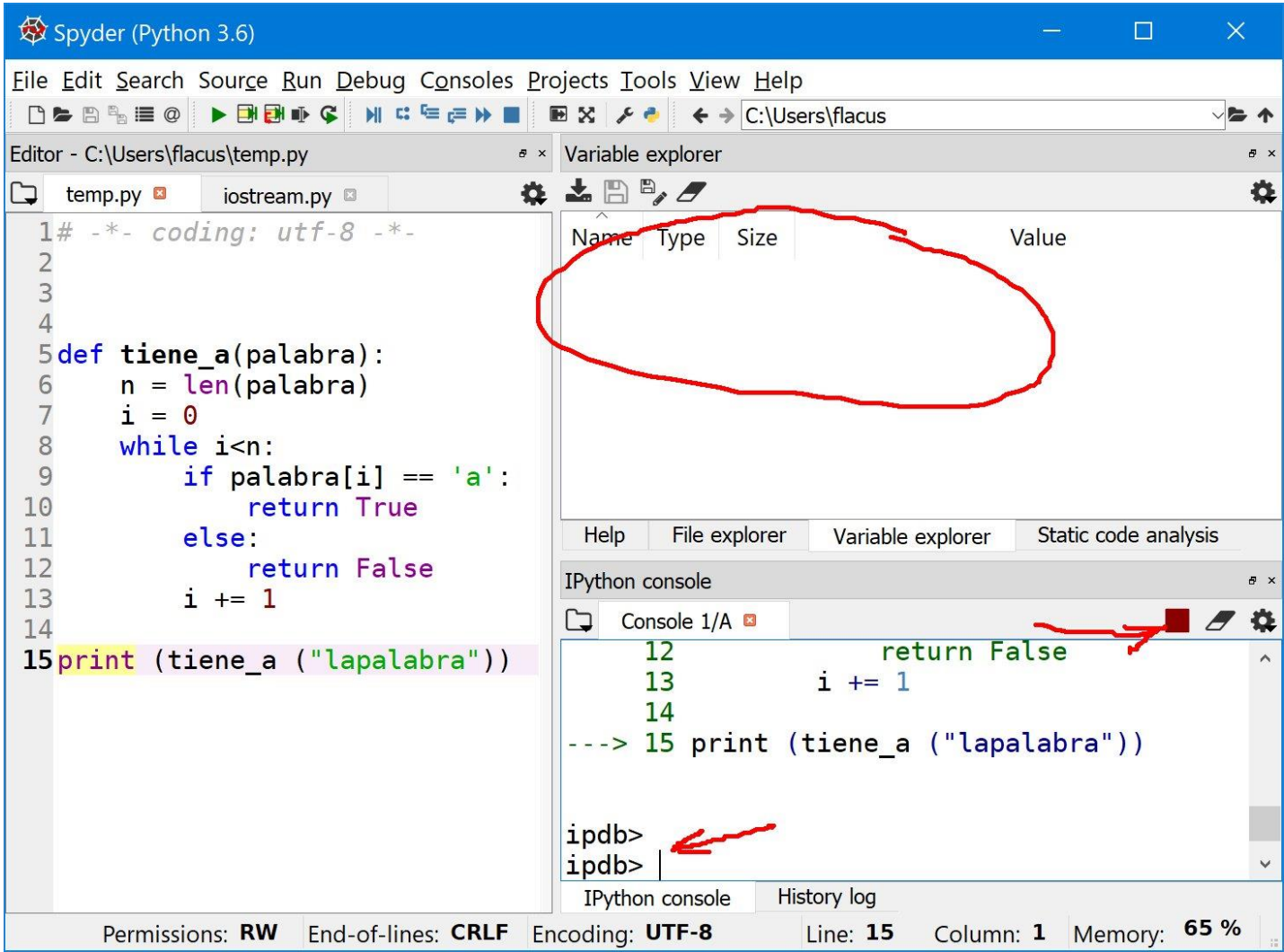
Para no entrar dentro del return (que es complicada) pedimos un Step (NO un Step Into)



En este punto acabamos de volver de la function. Las variables internas ya no están visibles (salimos de su "scope"). El programa sigue en ejecución, (flechas)

Salir de la función después de haber analizado solo la primera letra no era lo deseado. Que pasó ? A pensar !

Pedimos un Step …



El programa terminó. Las flechas indicant el STOP apagado y el prompt normal. Tenemos control del IDE y la tarea ahora es pensar porqué la función terminó antes de lo deseado.