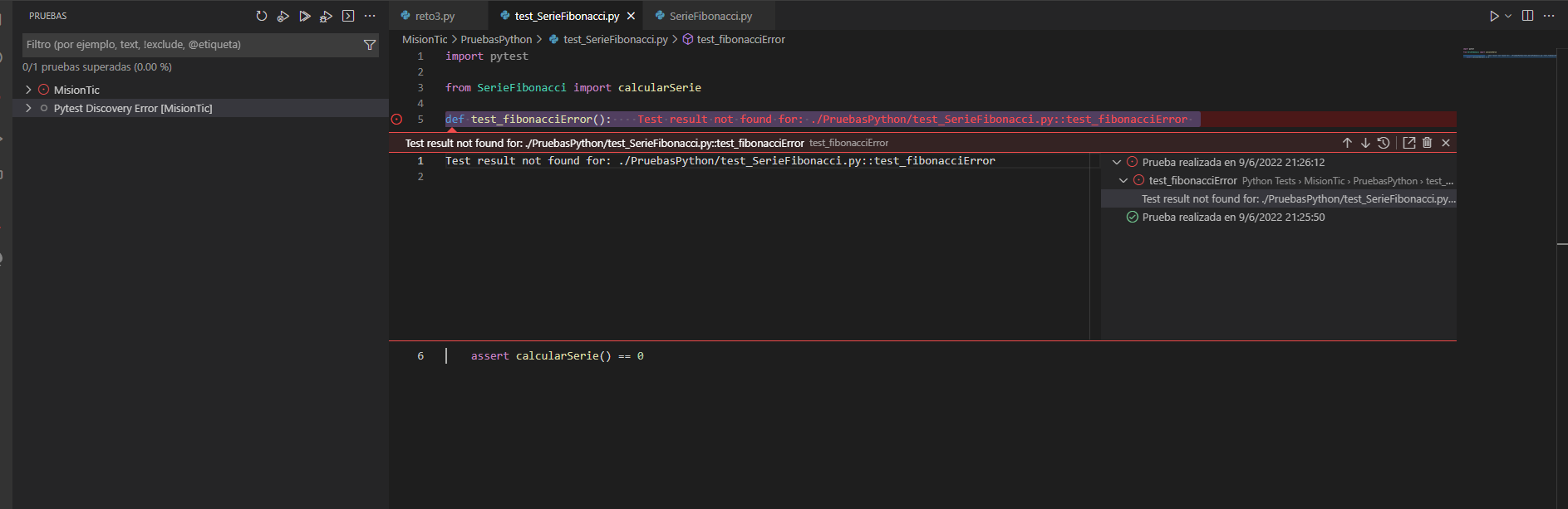
Aplicando algoritmo TDD en lenguaje no conocido.

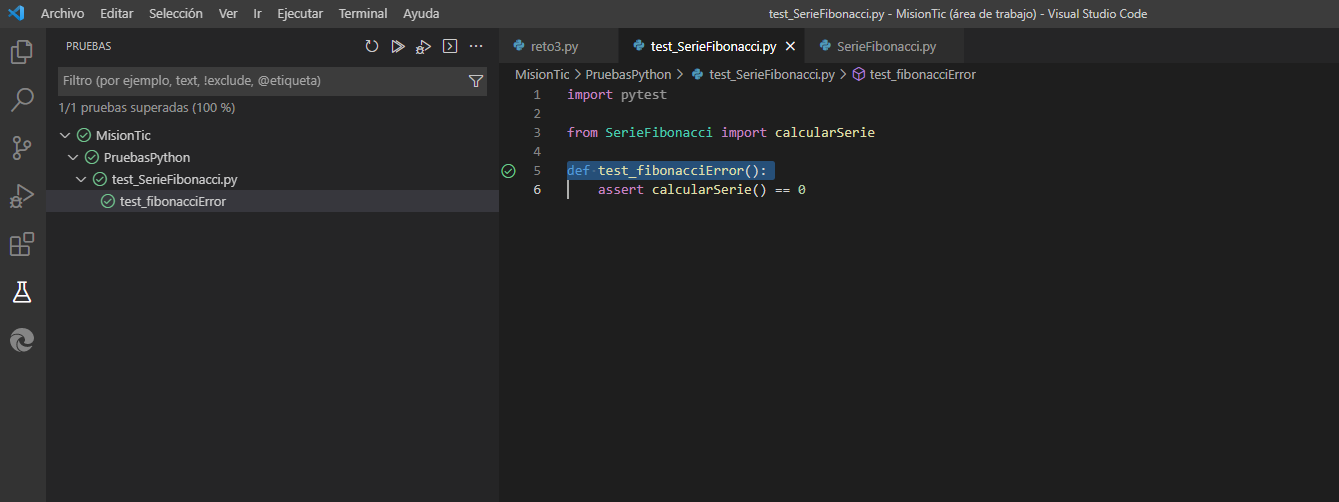
En este caso ulitice python con pytest para realizar el desarrollo de las pruebas

En el primer caso, se realizó la prueba con una funcion que no existia, se obtubieron los siguientes resultados.

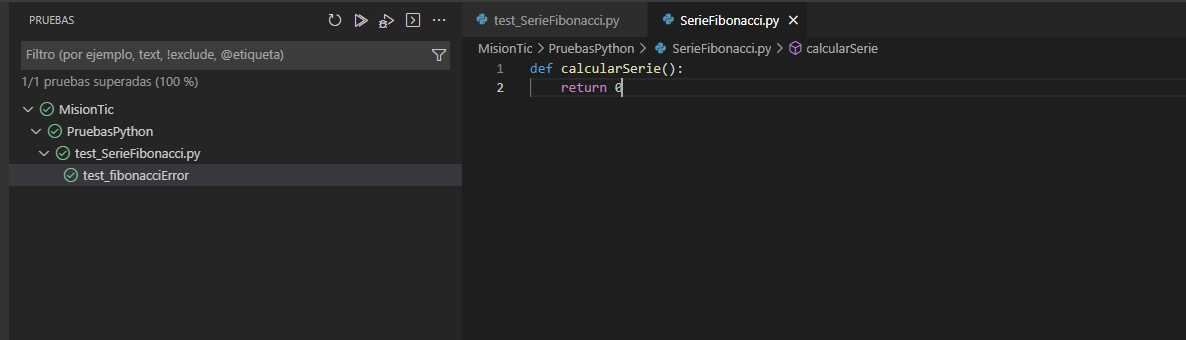


Evicentemente, nos marca un error porque no existe la funcion calcularSerie que estaba invocando.

Como siguiente paso, se procedio a realizar la prueba creando dicha funcion, pero sin ninguna implentacion de codigo, simplemente retornando un valor.

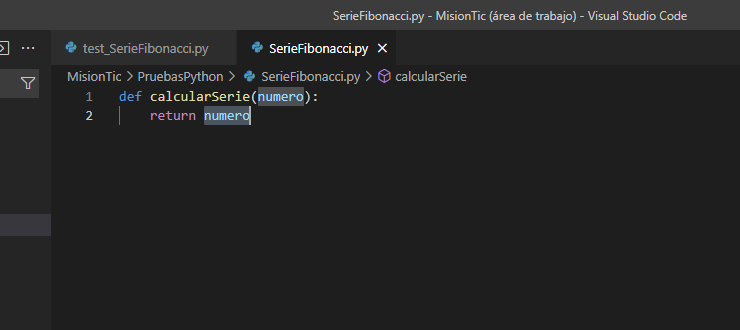


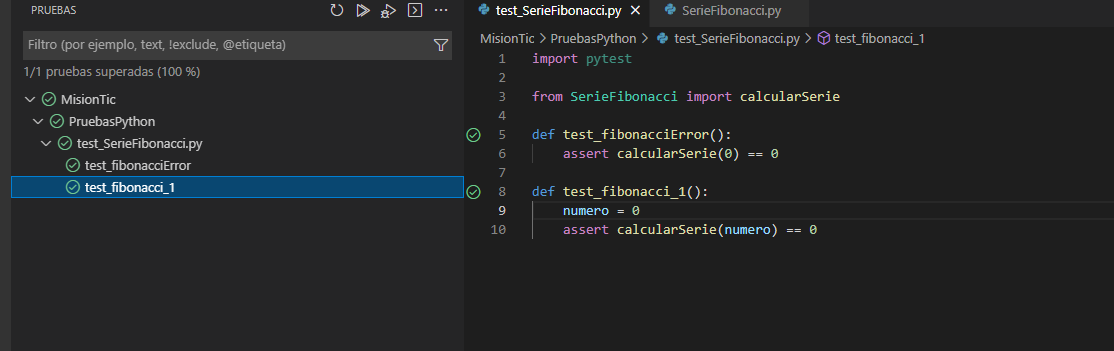
este es el archivo de pruebas, donde llamo la funcion creada, acontinuacion mostrare el archivo donde tenemos la funcion calcularSerie.



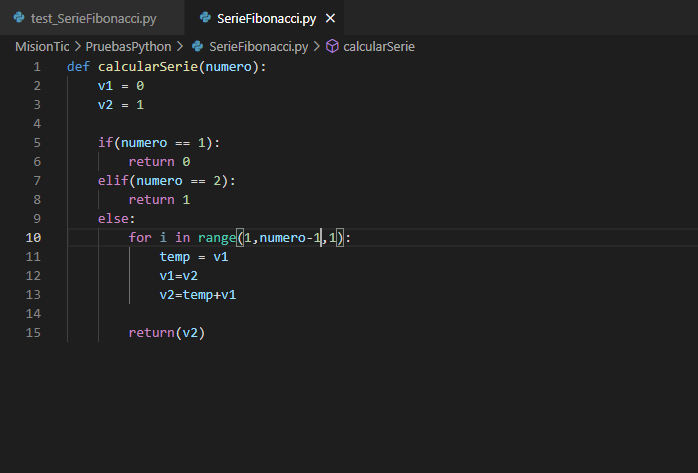
Como podemos observar, retornamos un cero, y en la funcion de pruebas, esperamos que nos devuelva un cero, y para este caso, cumpliriamos con lo relacionado a la parte de la creacion del metodo que anteriormente no existia.

Se procedio a crearla implementacion para retornar el numero que esperamos que retorne, en este caso, si se le dijita un 1 deberia devolvernos cero, ya que la serie de fibonacci empieza en cero.



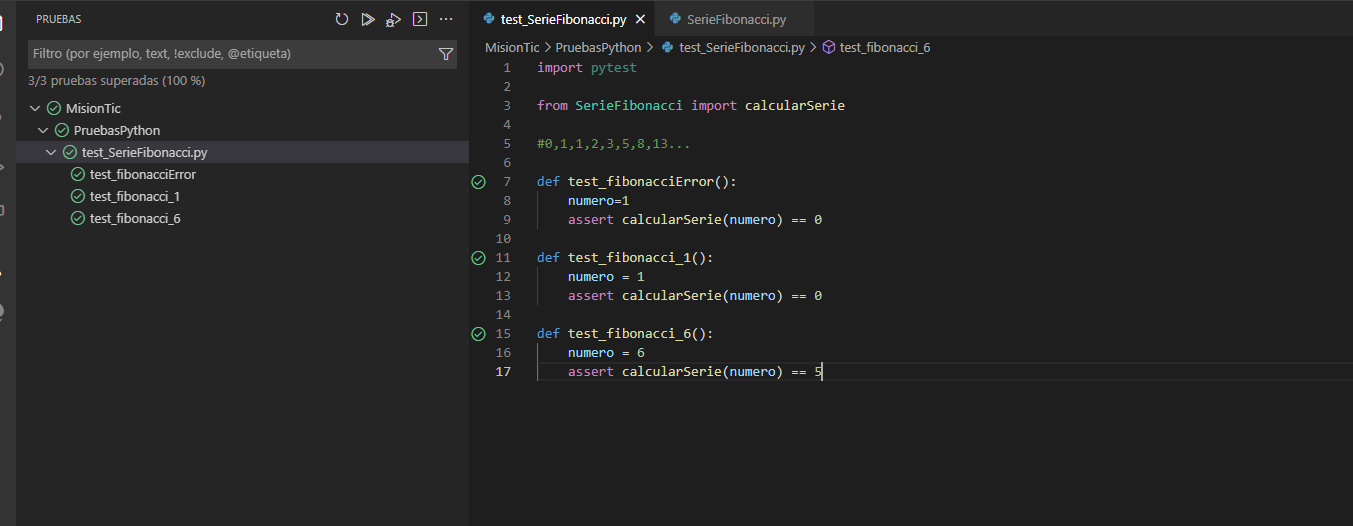


Una vez evaluado ese caso, se procedio a darle implementacion al algoritmo de la serie de fibonacci en la funcion calcular serie.



Una vez implementado, se procedio a realizar las pruebas, con el valor de 1 esperando un resultado de cero, y se obtuvo una prueba satisfactoria, por lo tanto, se procedio a realizar el testing con diferentes valores.

Ejemplo con el numero de la serie en la poscicion 6, teniendo encuenta que la serie va (0,1,1,2,3,5,8,13...) esperamos que la funcion nos devuelva un 5



Por lo tanto, procedi a agregar un decorador, para hacerlo con diferentes valores, sin tener que escribir prueba por prueba.

