Justo Mota Márquez

**Lección 4: Buenas Prácticas de Programación en Python**

**Apartado 1:**

1. Creo una matriz de números enteros (en mi caso he usado la del ejemplo):



1. Creo una compresión de listas en la que voy comprobando en cada lista que contiene la matriz, si el valor de cada iteración de la lista es igual al valor máximo de esta misma. Si el valor de la iteración es igual al valor máximo de la lista, le concatenamos a este número los caracteres ANSI correspondientes al texto en negrita, en el caso de que no sea igual el valor máximo que al valor de la iteración dejamos el número tal y como está:



1. Recorro con bucles la matriz que contiene la matriz inicial modificada con los valores máximos de cada lista en negrita, y voy pintando cada valor de esta misma:

Texto

Descripción generada automáticamente

1. El programa muestra por pantalla:



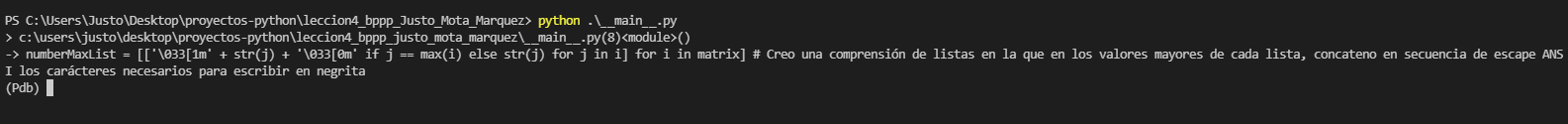
1. Conclusión:

La compresión de listas me ha hecho reducir líneas de código y a la vez aumentar la eficiencia del programa al guardar una lista filtrada directamente y no tener que hacer bucles anidados en mas de una línea.

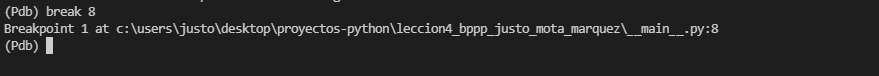
**Apartado 2:**

1. Especifíco al programa la línea en la que quiero que comience a depurar, con la función “pdb.set\_trace()”:





1. Creo un punto de parada en la línea anterior en donde empleo la compresión de listas y otro en la línea posterior que no esté en blanco, con el comando pdb “break (número de línea)”:





1. Compruebo que exista el punto de parada:

Texto

Descripción generada automáticamente

1. Compruebo el valor de la variable que contiene la compresión, con el comando “p (nombre de la variable)”:



1. Ejecuto linea el programa hasta el final:

Texto

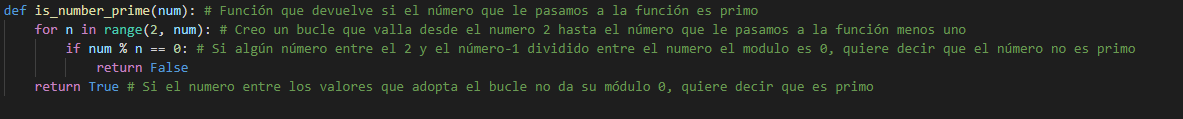
Descripción generada automáticamente

1. Conclusión:

La depuración me ha servido para ver que esta pasando en mi código en cada línea e ir viendo los valores que van adoptando las variables

**Apartado 3:**

1. Creo una función que devuelva “True” si el número que le pasamos es primo y “False” si no lo es:



1. Creo una lista con valores desde el 1 al 20:



1. Creo una lista filtrada que me guarda los números primos que hay entre el 1 y el 20, con la función “filter(función, lista)”:



1. El programa muestra por pantalla:



1. Conclusión:

El uso de la función “filter” agiliza el proceso de filtración de listas para eliminar los datos que no necesitamos, ahorrando en código y tiempo.