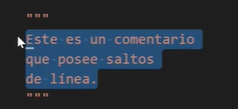
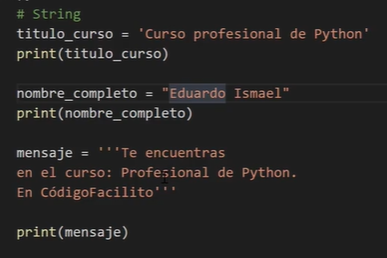
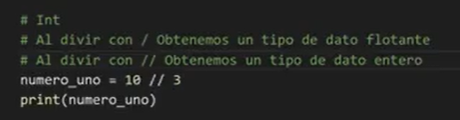
**Modulo 1**

Clase 8:

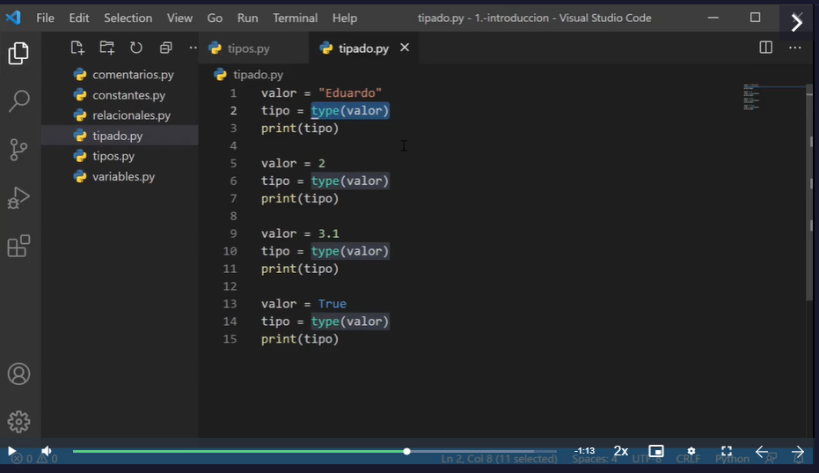
Los comentarios también se pueden colocar con triple comilla doble arriba y abajo del comentario

Clase 9:

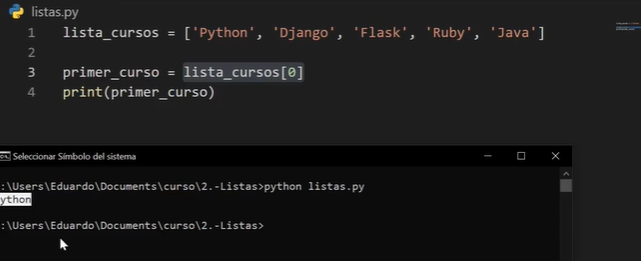
Esto nos ayuda a imprimir un texto con espacios y que aparezcan como línea abajo y no todo junto, pero esto es con las triples comillas ya sean simples o dobles.



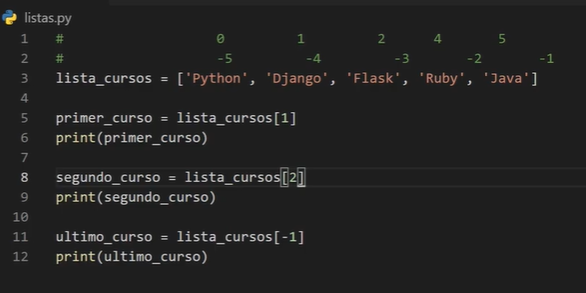
**Clase 13**

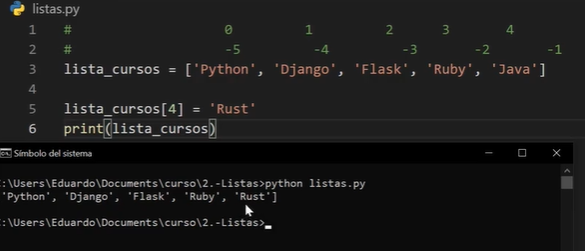
Para conocer el tipo de dato

**Modulo 2 - Listas**

**Clase 2**

Para imprimir el valor de una lista colocando la posición



**Clase 3**

Para reemplazar o actualizar un valor en la lista

**Clase 4**

Pantalla de un celular con texto e imagen

Descripción generada automáticamentepuedo crear sub listas, y en los corchetes coloco los valores que tenga, pero aquí no cuenta el ultimo valor que se le indica, en este caso solo toma hasta la posición 2, no toma en cuenta el ultimo Texto

Descripción generada automáticamentevalor del rango en la lista

sí colocamos más rango de lo que tiene, va a imprimir normal y todos los datos.

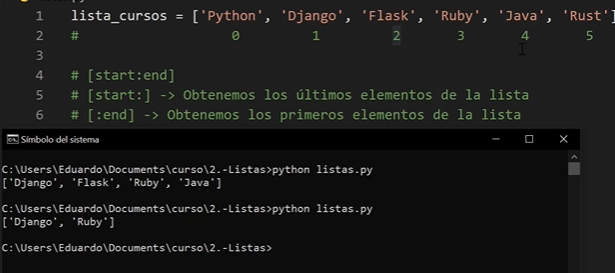
Texto

Descripción generada automáticamente

si no se coloca numero al final, entonces la lista hace que sea desde la posición que indica y automáticamente lo hasta el número final o ultimo numero

Captura de pantalla de un celular

Descripción generada automáticamentelo contrario si queremos obtener solo los primeros valores hasta la posición que se le indica

También se puede hacer generando saltos,

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamenteAquí sin indicar de donde empieza o donde termina imprime todos los elementos

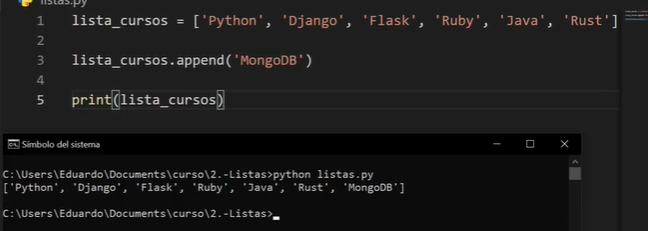
Interfaz de usuario gráfica, Texto

Descripción generada automáticamentey lo mismo hace con los saltos, que imprime todo tomando en cuenta los saltos

Interfaz de usuario gráfica, Texto

Descripción generada automáticamenteEsto invierte el orden de la lista



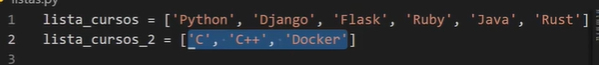
**Clase 5**

Así se agrega valores al final de la lista

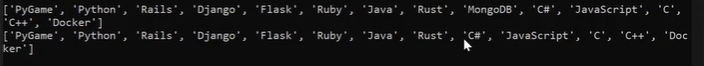
Texto

Descripción generada automáticamenteEl LEN es para medir el tamaño de una lista

esto sirve para añadir valores a la lista pero en la posición que yo le indique pero este no reemplaza, solo añade en la posición haciendo que todos los elementos se muevan a la derecha



extend permite añadir los valores de una lista a otra

esto sirve para eliminar valores en una lista, en la primera línea si aparece porque se imprimió la original, pero después de usar el remove, ya no aparece al valor que le dijimos

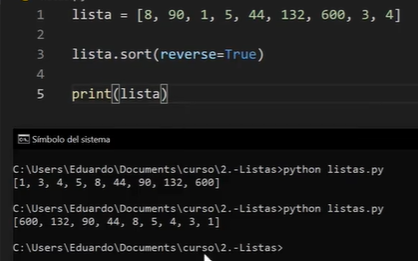
también se puede eliminar según la posición que le indiquemos, pero se debe usar “del” y corchetes

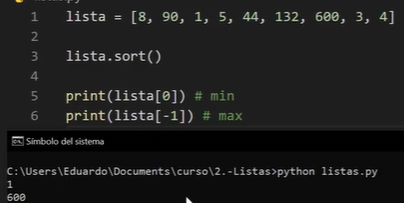
esto elimina todo los elementos de una lista

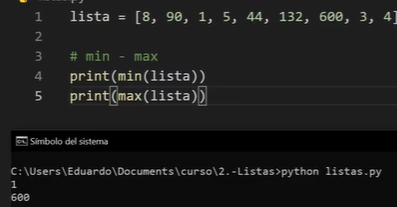
**Clase 6**

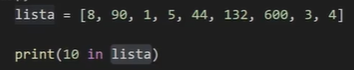
Texto

Descripción generada automáticamentesort permite ordenar los elemento de una lista de menor a mayor

y si queremos que se ordene de mayor a menor se coloca reverse=True

Una forma de encontrar el valor menor y mayor de una lista, primero ordenamos de menor a mayor y luego imprimimos el primer valor y luego el ultimo valor

otra forma sin que sea necesario ordenar, usamos mix y max entre paréntesis el nombre de la lista

este IN sirve para saber si un elemento está o no en una lista y arroja un valor booleano



Texto

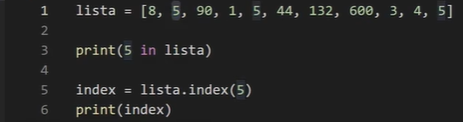
Descripción generada automáticamentecon NOT IN podemos confirmar si un elemento NO se encuentra en la lista, por eso sale True. El NOT niega el valor booleano

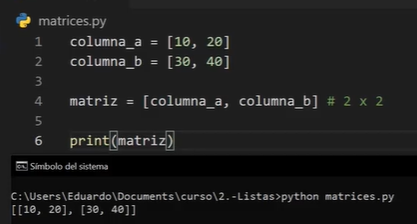
**Texto

Descripción generada automáticamenteClase 7**

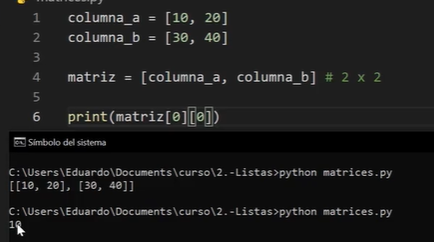
este INDEX sirve para saber el valor del índice en que se encuentra un valor, aquí no olvidemos que los índices (posiciones) comienzan desde 0

Texto

Descripción generada automáticamentepero si hay más datos repetidos, el INDEX solo imprime la primera posición, solo una, no en todas las que se encuentra ese valor

**Clase 8**

También se puede crear matrices así

esto sirve para saber la posición dentro de una matriz, de igual forma empieza a contar desde 0

Texto

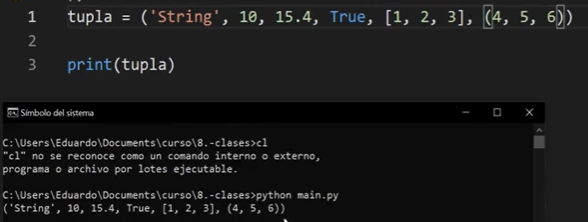
Descripción generada automáticamenteaquí las “X” son las filas y “Y” las columnas, por eso en el primer corchete colocamos 1 refiriendo a la segunda fila, y luego 0 refiriendo a la primera columna

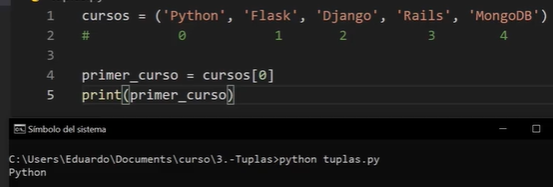
**Modulo 3 – Tuplas**

Estas tuplas son como las listas solo que no se pueden modificar, después de definir una se queda así para todo el programa. No se puede agregar datos ni quitar ni nada, pero estos pueden almacenar al mismo tiempo datos, números, decimales y booleanos. También se pueden agregar listas dentro de una tupla. Y también tuplas dentro de una tupla

Interfaz de usuario gráfica, Texto

Descripción generada automáticamenteuna tupla de crea de esta forma, con cualquier nombre le damos pero es si o si con paréntesis.

esta es una demostración de que si se puede tener todo tipo de datos dentro de una tupla

se puede saber los índices de las tuplas, ya que no se pueden hacer nada más, más que todo sirven para esto las tuplas

Interfaz de usuario gráfica, Texto

Descripción generada automáticamentetambién se pueden crear sub-tuplas

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamenteesto permite convertir una lista en una tupla, usando TUPLE

Texto

Descripción generada automáticamentey lo mismo viceversa, convertir una tupla en una lista, usando simplemente LIST

Texto

Descripción generada automáticamente

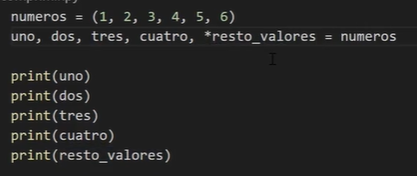
también se puede designar a cada tupla una variable, es como una forma se separarla

Texto

Descripción generada automáticamente

se hace esto para no hacer una por una como lo de arriba

Texto

Descripción generada automáticamentesi no sabemos cuántos valores tiene una Tupla y queremos asignarles a variables con ese \* antes del nombre, decimos que todas las palabras que sobran vayan a esa variable. Pero se agregaran a esa variable como lista, no como una tupla. El asterisco es importante porque hace referencia a una lista

Texto

Descripción generada automáticamentepero si no quiero asignarle algún valor, coloco \*\_ y solo crea para los valores que si se les indicó

Imagen de la pantalla de un celular con letras

Descripción generada automáticamente con confianza bajaesto así funciona

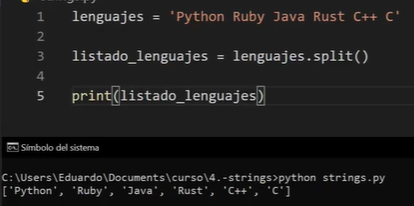


Texto

Descripción generada automáticamente con confianza mediapero si queremos nombrar a los últimos valores, solo le designamos una variable después del \*\_. Pero si queremos que no se tome en cuenta solo un valor ya no es necesario colocar el asterisco, directo el \_, porque el asterisco lo tomará en cuenta como si fuera una lista.

**Modulo 4 – strings**

**Clase 1**

Esto permite convertir de una variable todas sus palabras separadas por un espacio en una lista, usando SPLIT

Interfaz de usuario gráfica, Sitio web

Descripción generada automáticamentesi queremos separar las palabras pero no con un espacio, colocamos dentro del Split el argumento que queremos que sea el que separe

Interfaz de usuario gráfica, Texto

Descripción generada automáticamentey esto hace todo lo contrario, de una lista convertirla a una sola variable, separándolas con lo que se le indique entre las comillas, en este caso con el espacio

en conclusión el método SPLIT genera un listado y el método JOIN genera un string o variable

**Clase 2**

Texto

Descripción generada automáticamente****esto sirve para unir dos strings o concatenarlos. Pero se unen sin tener un espacio

así que le agrego el espacio para que así se separen bien

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamentey también puedo hacer esto para tener el mismo resultado. En donde el %s toma el valor del string, se acuerdo al orden que le coloque dentro de los paréntesis

**Clase 3**

esta es otra forma de hacer lo mismo

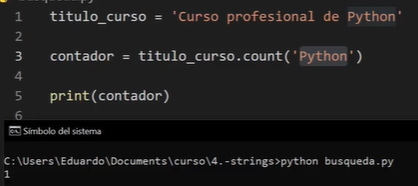
**Clase 4**

también puedo hacer de esta forma para hacer lo mismo, pero con la idea de que puedo agregar nuevas operaciones dentro del string.

**Clase 5**

También se puede imprimir varios valores al mismo tiempo, pero siempre separados por un espacio, esto por definición

pero también puedo definir con qué carácter quiero separarlos. Usando SEP

**Clase 6**

COUNT sirve para indicar cuantas veces se repite un string dentro de otro string.

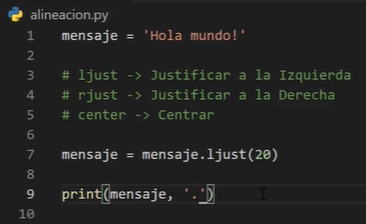
.lower() sirve para poner todo a minúsculas

.upper() sirve para poner todo en mayúsculas

este sirve para saber si el primer valor es el que se le indica en el paréntesis, y retorna un valor booleano.

también podemos saber si el ultimo es el que le indicamos

**Clase 7**

esto sirve para justificar el texto

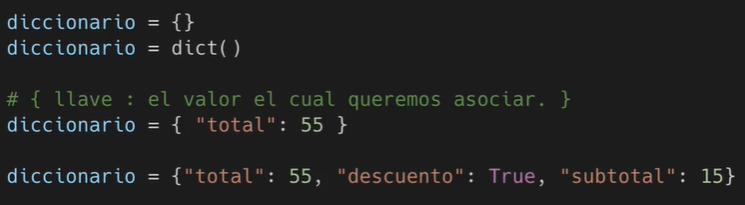
en el paréntesis se coloca la cantidad de espacios que queremos que haya

Texto

Descripción generada automáticamente con confianza baja Y con center lo que logramos es que 10 espacios se vayan para la izquierda y otros 10 a la derecha



**Modulo 5**

**Clase 1**

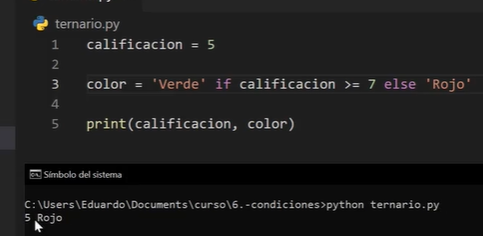
**Modulo 6**

Texto

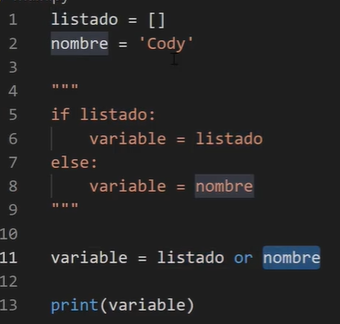
Descripción generada automáticamente**Clase 2**

Estas son las formas de Python para indicar que un valor es falso

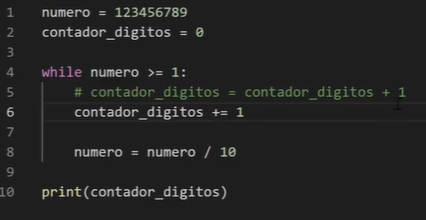
**Clase 5**

esta es otra forma de reducir el uso del IF, pero si o si debe colocarse el ELSE sino sale error

**Clase 6**

****el IF imprime solo valores que están definidos, no imprime valores falsos, lo mismo con las variables, solo tomará los valores que sean verdaderos o aceptables

**Clase 7**

el while sirve más que todo cuando no sabemos con cuantos datos vamos a trabajar. Y ese += sirve para indicar que vamos a sumarle a ese mismo valor.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamentecuando ya termine el ciclo de while, se puede usar ELSE para indicar que ya se terminó la condición

Clase 8

Texto

Descripción generada automáticamenteaquí el USUARIO vendría a ser la i que se usa, entonces esa palabra va a tomar el valor de cada carácter de la lista. Y aquí no se usa el else porque no estamos usando una condicional

Clase 9

Texto

Descripción generada automáticamenteTambien si se tiene una variable, lo que va a imprimir es cada letra de esa variable

Texto

Descripción generada automáticamenteaquí el BREAK sirve para finalizar algo, en este caso finaliza el IF por lo cual se llegará a finalizar el for

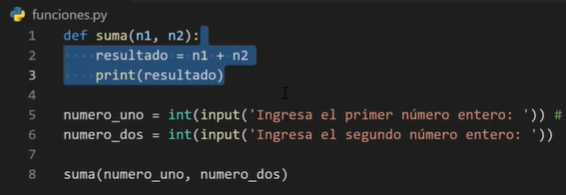
Texto

Descripción generada automáticamenteLo contrario de BREAK es CONTINUE, que cuando se de la condición lo que hará es saltearse, en este caso no toma en cuenta el espacio y se saltea a la siguiente letra o carácter en este caso. Lo que hace es saltearse la ejecución.

Aquí el ciclo pasa a la siguiente iteración (letra o string) por eso ya no se imprime el carácter.

**Break** corta la iteración y **continue** hace pasar directo a la siguiente

**Modulo 7**

**Clase 2**

Cuando usamos funciones no es necesario siempre usar RETURN si es que solo necesitamos que se imprima su valor.

Aquí tambien vemos que n1 y n2 toman el valor de numero\_uno y numero\_dos de acuerdo al orden que se le indicó cuando se llamó a la función al final.

**Texto

Descripción generada automáticamenteClase 3**

Pero si queremos que de la función salga un valor el cual lo usaremos en otras funciones o en otra parte del código si es necesario usar el RETURN

Texto

Descripción generada automáticamenteo igual se puede hacer la operación dentro del return

Texto

Descripción generada automáticamentetambién puede retornar varios valores, pero los devuelve como una tupla.

Texto

Descripción generada automáticamente

Texto

Descripción generada automáticamenteO tambien se puede hacer asi por separado la tupla como se vio antes.

Texto

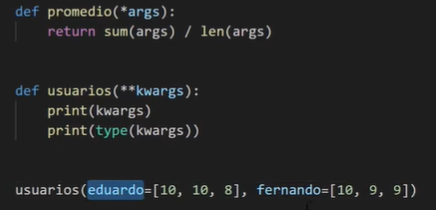
Descripción generada automáticamenteClase 4

Para elevar es el doble asterisco \*\*,

Captura de pantalla de un celular

Descripción generada automáticamentepero tambien se puede definir dentro de los argumentos el valor de una variable (pero sin espacios según la comunidad de Python, todo debe estar pegado)

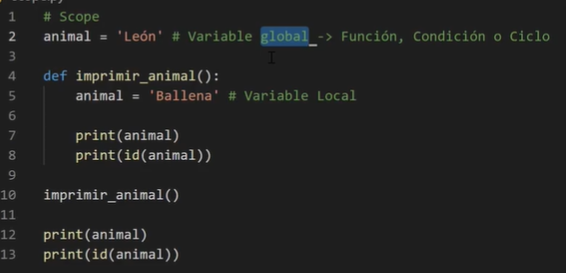
y estos valores colocados por default deben estar siempre a la derecha

**Clase 5, 6 y 7**

\*arg sirve para convertir en una tupla

\*\*kwargs sirve para convertir en un diccionario

Aquí al definir algo dentro de los argumentos no se debe colocar tampoco espacios, todo debe estar junto

**Clase 8**

Cuando una variable es creada antes que una función toma el valor de una variable global. Y si el mismo nombre de la variable es definido dentro de otra función o ciclo será ese valor pero un valor local, pero ambas seguirán siendo variables diferentes. Y las locales solo se usan dentro de ese bloque, fuera de eso toda variable será la global

Todas las variables creadas dentro de un bloque solo se pueden usar en ese bloque y no afuera

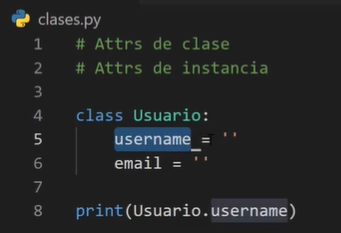
Texto

Descripción generada automáticamentePor lo que las locales solo funcionan dentro de un bloque y una global funciona dentro y fuera de cualquier bloque

Pero si quiero modificar el valor de una variable global tengo que hacerlo con GLOBAL y luego nombro la variable global que quiero cambiar y ya después del bloque le indico el nuevo valor que tendrá esa variable global

**Modulo 8 – Clases**

Clase 2 – estos son atributos de clase

todo lo creado dentro de una clase así como se crea una variable, se le llama atributo. Y se lo imprime primero indicando el nombre de la clase punto y el atributo que queremos llamar

Interfaz de usuario gráfica, Texto

Descripción generada automáticamente

también podemos modificar lo que vale ese atributo pero siempre llamando a la clase punto atributo y luego lo que queremos que sea ahora

<https://github.com/codigofacilito/curso-profesional-python>