

Python Cheat Sheet	Metodos String	Listas – Acciones Permanentes	Diccionarios – Metodos	zip()
Variables ampliadas por text (CONCATENATION)	 string.upper() MAYUSCULAS string.lower() minusculas string.capitalize() Primera letra de la frase en may. string.title() Primera Letra De Cada Palabra En May. string.swapcase() mINUSCULAS A mAYUSCULAS O vICEVERSA string.strip() quita espacios del principio y final .split() string.split() divide string en lista – por espacios por defecto, o especifica otro divisor en () .replace() string.replace(“frase”, “frase”) reemplaza la primera frase del string por el otro .join() “ ”.join(string) une los elementos de una lista en una string con el separador espificado en “ ” list() list(string) convierte un variable string en una lista string.find(“substring”) encuentra el indice en que empiece el substring/'-1' si no existe el substring string[i] devuelve el elemento en la indice i string[i:j] devuelve un rango de caracteres	Ampliar una lista [lista1, lista2] junta listas pero se mantienen como listas separadas lista1 + lista2 hace una lista mas larga .append() lista.append(x)# añade <u>un</u> solo elemento (lista, string, integer o tuple) a la lista .extend() lista.extend(lista2)# añade los elementos de <u>una</u> lista al final de la lista .insert() .insert(i, x)# mete un elemento (x) en un índice(i) Ordenar una lista .sort() lista.sort()# ordena de menor a mayor, usar con (reverse=True) para ordenar de mayor a menor lista.reverse()# ordena los elementos al revés del orden guardado Quitar elementos de una lista .pop() lista.pop(i)# quita el elemento en indice i y devuelve su valor .remove() lista.remove(x)# quita el primer elemento de la lista con valor x lista.clear()# vacia la lista del lista# borra la lista del lista[i]# borra el elemento en indice i	Obtener informacion de un diccionario dicc.keys() devuelve todas las keys dicc.values() devuelve todos los valores dicc.items() devuelve tuplas de los key:value in/not in comprobar si existe una clave dicc.get(x, y) devuelve el valor asociado al key x, o si no existe devuelve el output y Ampliar un diccionario .update() dicc.update({x:y})# para insertar nuevos elementos dicc[“key”] = valor# para inserter un nuevo key o valor, o cambiar el valor de un key dicc. setdefault(x, y)# devuelve el value del key x, o si no existe la key x, la crea y asigna el valor y por defecto Quitar elementos de un diccionario dicc.pop(x)# elimina la key x (y lo devuelve) dicc.popitem()# elimina el ultimo par de key:value dicc.clear()# vacia el diccionario	zip(iterable1, iterable2) crea una lista de tuplas de parejas de los elementos de las dos listas (mientras se puede) listzip.sort() ordena las tuplas del zip por el primer elemento Sets {} no permiten duplicados, no tienen orden set = {x,y} set(iterable) solo permite un argumento iterable; elimina duplicados in/not in comprobar si hay un elemento len(set) devuelve el no. de elementos Ampliar un set set.add(x)# añadir un elemento set.update(set o lista)# añadir uno o mas elementos con [] o {} o un variable tipo lista o set Quitar elementos de un set set.pop()# elimina un elemento al azar set.remove(x)# elimina el elemento x set.discard(x)# elimina el elemento x (y <u>no</u> devuelve error si no existe) set.clear()# vacia el set
	Listas [] Metodos no permanentes lista = [] crea una lista vacia len(lista) devuelve el no. de elementos min(lista)/max(lista) saca el valor minimo y maximo lista.count() devuelve el no. de elementos que hay en la lista de un valor determinado en los() sorted(lista) ordenar una lista de menor a mayor lista.copy() hacer una copia de la lista	Diccionarios { key : value , } diccionario = {x:y} compuestos por un key(x) unica y un valor(y) (cualquier tipo de datos) dict() variable = dict(x=y, m=n) crear un diccionario dicc.copy() crear una copia len(dicc) devuelve el no. de elementos (x:y) hay en el diccionario sorted(dicc) ordena los keys; usar con .items() para ordenar tuplas de los elementos o .values() para ordenar los valores solos	Tuplas (,) inmutables, indexados tupla = (x,y) tuplas se definen con () y , o solo , tupla1 + tupla2 juntar tuplas tuple(lista) crear tuplas de una lista tuple(dicc) crear tuplas de los keys de un diccionario tuple(dicc.values()) crear tuplas de los valores tuple(dicc.items()) crear tuplas de los key:valores len(tupla) devuelve el no. de elementos in/not in comprobar si hay un elemento tupla.index(x) devuelve el indice de x tupla.count(x) devuelve el no. de elementos con valor x en la tupla *para cambiar el contenido de una tupla hay que convertirla en una lista y luego a tupla*	
Operaciones Algebraicas + sumar / dividir - restar // divider y redondear (modulus) * multiplicar % resto de una division (floor division) ** elevar round(x) redondear numero x	Operaciones Binarias == comprobar si valores coinciden is comprobar si valores son exacamente igual != comprobar si valores son diferentes is not comprobar si valores no son exactamente iguales > (>=) mayor que (mayor o igual que) < (<=) menor que (menor o igual que) and ambas verdaderas or ambas o solo una verdadera in/not in comprobar si hay un valor en una lista etc.			

metodos permanentes (cambia el variable, no devuelve nada)