

Auxiliar 1: Cosas previas y funciones

Todos los problemas deben ser resueltos en *Python*, utilizando estrictamente la Receta de Diseño entregada a lo largo del curso. Use nombres apropiados para funciones y variables, y testee cada vez que sea posible.

0. Paso imprescindible para la realización de este curso

Instalar Python :)

1. Funciones

1.1. Pregunta 1

- Defina una función llamada `invertir` que, dado un número de dos dígitos, retorne el mismo número dado vuelta. Ejemplo: `invertir(58)` retorna 85
- Defina una función llamada `velocidadKph(metros, segundos)`, que dada una cantidad de metros recorridos en cierta cantidad de segundos, entregue la velocidad correspondiente en kilómetros por hora. Ejemplo: `velocidadKph(40, 3)` entregará 48.
Observación: esta función trabaja con números reales, no enteros, tenga cuidado con la división.
Hint: el orden de los factores no altera el producto :)

1.2. Pregunta 2

Para esta pregunta, usted supondrá que es un/a cajero/a de un banco (o simplemente una persona muy ordenada que quiere andar con la menor cantidad de billetes posible):

1. Se le pide programar una función llamada `billetes20k`, que dada una cantidad de dinero, le dice cuál es la cantidad máxima de billetes de \$20000 que puede tener. Ejemplo: `billetes20k(95000)` retorna 4, ya que puede tener \$80000 en billetes de \$20000, pero el resto deberá tenerlo en billetes de otro valor.
2. Ahora, le solicitan realizar lo mismo, pero con billetes de \$10000 (es decir, una función que se llame `billetes10k`).
3. Nuevamente le piden lo mismo, pero para billetes de \$5000. Usted, ya cansado/a de escribir la misma función para distintos valores de billetes, decide que es momento de crear una función genérica. Llame a esta función `nBilletes(cantidad, valorBillete)`. Note que esta función tomará 2 parámetros: la cantidad de dinero del que dispone, y la nominación del billete.
4. Finalmente, utilizando la función `nBilletes`, programe una función llamada `organizarDinero`, que dada una cantidad de dinero, imprima en pantalla la mínima cantidad de billetes que necesita para tener ese dinero junto a la nominación correspondiente. Ejemplo: `organizarDinero(98000)` imprimirá algo como:
 - 4 billetes de 20000
 - 1 billetes de 10000
 - 1 billetes de 5000

- 1 billetes de 2000
- 1 billetes de 1000

Varias cosas que debe notar:

- No es necesario cambiar de singular a plural cuando se tiene 1 o más billetes
- Note que cuando ya calculó la cantidad de billetes de \$20000 necesarios, sólo debe trabajar con el resto (lo mismo cuando ya terminó de trabajar con los billetes de \$10000, etc)
- Asuma que sólo existen billetes de 20000, 10000, 5000, 2000 y 1000 pesos (no se preocupe por las monedas -aunque obviamente, estas son las más apañadoras -)
- Finalmente, lo más importante: note que esta función NO RETORNA NADA, pero aún así obtenemos resultados. ¿Cómo puede ser esto? (no, no es magia)