**Sep**

Cuando se pasan varios argumentos a la función de impresión, se separan automáticamente con un espacio cuando se muestran en la pantalla. Aquí hay un ejemplo, demostrado en modo interactivo:



si no desea que se imprima un espacio entre los elementos, puede pasar el argumento sep = '' a la función de impresión, como se muestra aquí:



También puede utilizar este argumento especial para especificar un carácter que no sea el espacio para separar varios elementos. Aquí hay un ejemplo:



Observe que en este ejemplo, pasamos el argumento sep = '\*' a la función de impresión. Esto especifica que los elementos impresos deben separarse con el carácter \*. Aquí hay otro ejemplo:



**Slash ( \ )**

La mayoría de las declaraciones de programación están escritas en una línea. Si una declaración de programación es demasiado larga, sin embargo, no podrá verlo todo en la ventana del editor sin desplazarse horizontalmente. Además, si imprime el código de su programa en papel y una de las declaraciones es demasiado de largo para caber en una línea, pasará a la siguiente y hará que el código sea difícil de leer.

Python le permite dividir una declaración en varias líneas usando la continuación de línea carácter, que es una barra invertida (\). Simplemente escriba el carácter de barra invertida en el punto desea romper la declaración y luego presione la tecla Intro. Aquí hay una función de impresión llamada que se divide en dos líneas con el carácter de continuación de línea:



El carácter de continuación de línea que aparece al final de la primera línea le dice al intérprete

que la declaración continúa en la siguiente línea. Aquí hay una declaración que realiza un cálculo matemático y se ha dividido para que quepa en dos líneas:



Esta larga declaración se divide en tres líneas. Observe que las dos primeras líneas terminan con una barra invertida.

**Ejercicio estructura de control repetitiva**

**1.-Definición del problema**

Diseñar un programa que permita calcular la comisión de las ventas, se debe utilizar el ciclo while.

**2.- Análisis del problema**

**Entrada:** Solicita 2 datos, uno para el monto de las ventas y otro para la tasa de comisión.

**Proceso:** Calcular la comisión mediante la operación de multiplicar con las ventas y con la tasa de comisión dentro del ciclo while.

**Salida:** Mostrar el resultado de la operación y un mensaje para solicitar al usuario si desea volver a realizar el cálculo deseado.

**3.-Diseño del algoritmo**

**Diagrama de flujo**

Plataforma: Lucidchart

Diagrama

Descripción generada automáticamente

**Pseudocódigo**

# Robles Pulido Efrain

#This program calculates sales commissions.

#Create a variable to control loop

keep\_going’y’

#Calculate a series of comissons

Mientras keep\_going = ‘y’:

#Get a salesperson’s sales and commission rate.

Entrada (“enter the amount of sales: “)

Leer (sales)

Entrada (“enter the commission rate: “)

Leer (comm\_rate)

#Calculate the commission

Commission sales \* comm\_rate

#Display the commission

Imprimir (“The commission is $”, Commission )

#See if the user wants to do another one.

Entrada (“Do you want to calculate another commission (Enter y for yes: “)

Leer (keep\_going)

**4.-Codificación**



