Clase 6 – Ciclos For y Recorridos

Marco Teórico

El bucle while sigue haciendo bucles mientras su condición es verdadera (que es la razón de su nombre), pero ¿qué pasa si desea ejecutar un bloque de código solo un cierto número de veces? Puede hacer esto con una sentencia for loop y la función range().

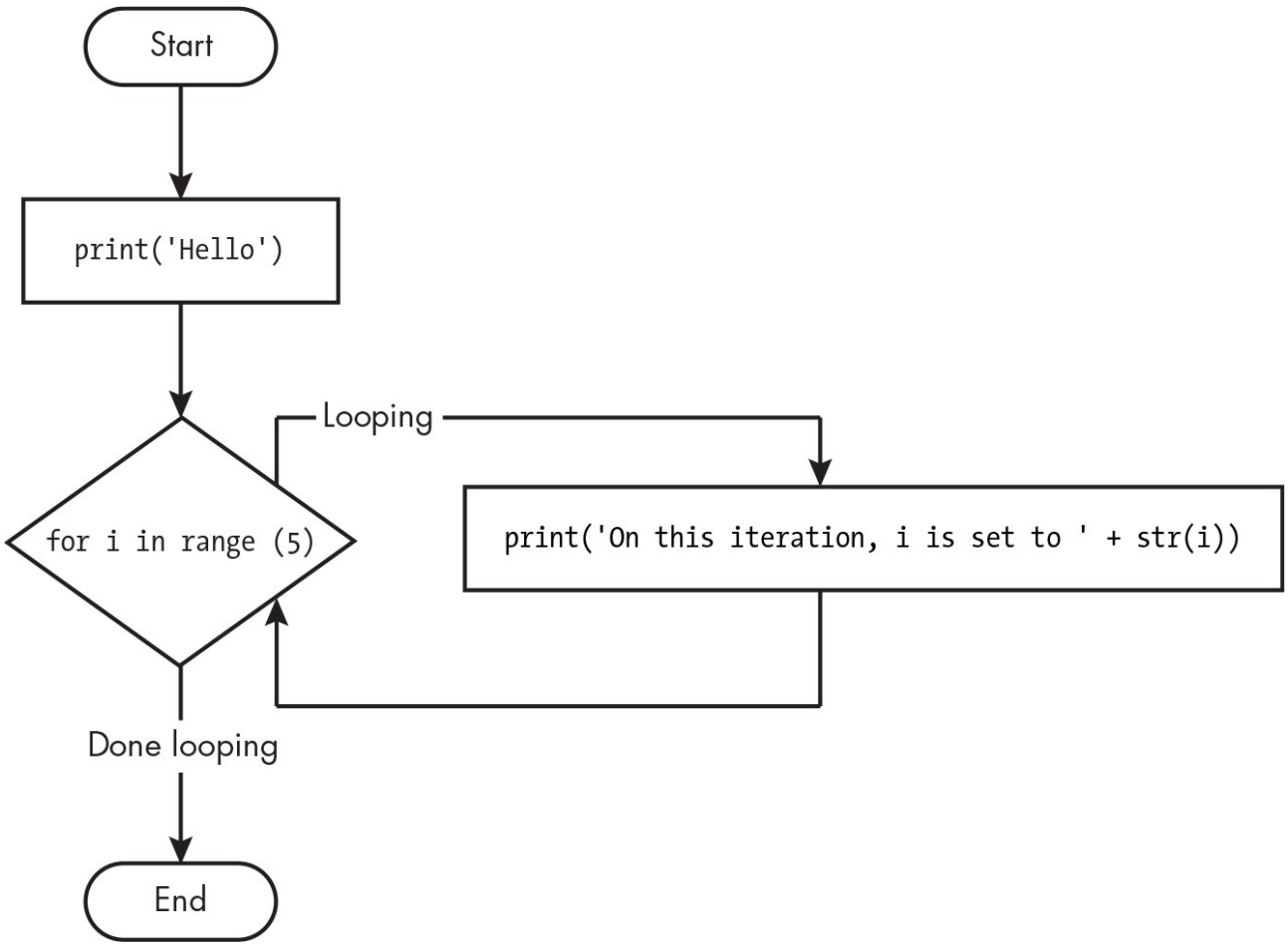
En código, una declaración de forma se ve algo así como i en el rango(5): e incluye lo siguiente:

- La palabra clave para (for)
- Un nombre variable
- La palabra clave en (in)
- Una llamada a la función range() con hasta tres enteros pasados a ella

Vamos a crear un nuevo programa llamado *fiveTimes.py* para ayudarle a ver un bucle de bucle en acción:

```
print('Hello!')
for i in range(5):
    print('On this iteration, i is set to ' + str(i))
print('Goodbye!')
```

El código de la cláusula del bucle para se ejecuta cinco veces. La primera vez que se ejecuta, la variable i se establece en 0. La llamada print() en la cláusula se imprimirá En esta iteración, se establece en 0. Después de que Python termina una iteración a través de todo el código dentro de la cláusula de bucle de for, la ejecución vuelve a la parte superior del bucle, y la declaración de declaración se incrementa i por uno. Esta es la razón por la que el rango(5) da como resultado cinco iteraciones a través de la cláusula, con i establecido en 0, luego 1, luego 2, luego 3 y luego 4. La variable i subirá a, pero no incluirá, el entero pasado a rango (). La Figura 3-6 muestra el diagrama de flujo para el programa *fiveTimes.py*.



Aquí está la salida completa del programa:

```
Hello!
On this iteration, i is set to 0
On this iteration, i is set to 1
On this iteration, i is set to 2
On this iteration, i is set to 3
On this iteration, i is set to 4

Goodbye!
```

El bloque de instrucciones que se repite se suele llamar cuerpo del bucle y cada repetición se suele llamar iteración.

Es especialmente útil para recorrer listas, cadenas y rangos numéricos.

Como otro ejemplo de loop, considera esta historia sobre el matemático Carl Friedrich Gauss. Cuando Gauss era un niño, un profesor quería darle a la clase algo de trabajo ocupado. El profesor les dijo que sumaran todos los números de 0 a 100. A Young Gauss se le ocurrió un truco inteligente para averiguar la respuesta en pocos segundos, pero puede escribir un programa de Python con un bucle para hacer este cálculo por usted:

```
total = 0
for num in range(101):
    total = total + num
print(total)
```

Estructura General

for variable in secuencia:

Función range()

La función range() genera secuencias numéricas.

Función range(inicio, fin, paso)

Ejemplos:

range(5)

range(1, 10)

range(0, 20, 2)

```
for i in range(0, 5):
    print(i)

# Salida:
# 0
# 1
# 2
# 3
# 4
```

En Python se puede iterar prácticamente todo, como por ejemplo una cadena. En el siguiente ejemplo vemos como la i va tomando los valores de cada letra. Mas adelante explicaremos que es esto de los **iterables** e **iteradores**.

```
for i in "Python":
    print(i)

# Salida:
# P
# y
# t
# h
# o
# n
```

Argumentos a rango()

Algunas funciones se pueden llamar con múltiples argumentos separados por una coma, y range() es uno de ellos. Esto le permite cambiar el entero pasado a range() para seguir cualquier secuencia de enteros, incluyendo comenzar en un número distinto de cero:

```
for i in range(12, 16):
    print(i)
```

El primer argumento será donde comienza la variable de bucle, y el segundo argumento dependerá de, pero no incluyendo, el número para detenerse en:

```
12
13
14
15
```

La función range() también se puede llamar con tres argumentos. Los dos primeros argumentos serán los valores de inicio y parada, y el tercero será el *argumento de paso*. La etapa es la cantidad en la que se incrementa la variable después de cada iteración:

```
for i in range(0, 10, 2):
    print(i)
```

Actividades Prácticas

Actividad 1 – Tabla de multiplicar

Generar la tabla de multiplicar de un número ingresado por teclado.

```
numero = int(input("Ingrese un número: "))

for i in range(1, 11):
    print(numero, "x", i, "=", numero * i)
```

Actividad 2 – Contador de vocales

Recorrer una palabra y contar cuántas vocales posee.

Actividad 3 – Sumatoria

Solicitar 5 números y calcular la suma total.

Actividad Integradora

Sistema de notas:

- Pedir 5 calificaciones.
- Mostrar promedio.
- Indicar aprobados y desaprobados.