

Modos de uso del intérprete

Lua tiene dos modos de uso diferentes: el modo interactivo y el modo script. A continuación te explicamos con más detalle estos dos modos.

Para simplificar los ejemplos usaremos el símbolo de sistema de Unix (\$), si usas Windows esto representa a por ejemplo C:>.

Modo interactivo

El modo interactivo permite la ejecución del código instrucción a instrucción. Todas las instrucciones son analizadas y ejecutadas inmediatamente y si hay algún resultado, se muestra inmediatamente. Para arrancar el modo interactivo, ejecutamos el comando `lua` sin argumentos. Veamos un ejemplo:

```
$ lua
Lua 5.2.4 Copyright (C) 1994-2015 Lua.org, PUC-Rio
>
```

El carácter `>` sirve para indicar que el intérprete de Lua está listo para aceptar instrucciones. Prueba el modo interactivo con el siguiente código:

```
> print("¡Hola Mundo!")
```

Después de cada instrucción presiona la tecla 'Entrar' para que el intérprete de Lua evalúe la instrucción. Si la instrucción tiene un resultado este será mostrado a continuación. En el caso de nuestro ejemplo la instrucción 'print()' imprime la cadena especificada en la consola:

```
> print("¡Hola Mundo!")
¡Hola Mundo!
>
```

Escribe ahora el siguiente código el intérprete interactivo y recuerda que después de cada instrucción debes pulsar la tecla `Entrar`:

```
> a = 2
> b = 2
> print(a+b)
```

```
4
>
```

En el ejemplo anterior, vemos que las dos primeras instrucciones son evaluadas pero no tienen ninguna salida, porque se trata de asignaciones. La tercera instrucción, como sabes, sí produce una salida, en este caso el valor de la suma de las variables a y b. Este es un ejemplo de instrucciones que son evaluadas y almacenadas en memoria para su uso posterior.

Modo script

En el modo script, a diferencia del modo interactivo en el que escribimos cada una de las instrucciones en la consola, le pasamos al intérprete un archivo en el que se definen todas las instrucciones a evaluar. Esto significa que el intérprete leerá y evaluará las instrucciones, línea a línea e irá mostrando salidas por la consola en función de las instrucciones, ejecutando la integralidad del archivo. El archivo en el que se definen todas las instrucciones recibe el nombre de script. Vamos a repetir el ejemplo anterior, pero esta vez escribiremos las instrucciones en un archivo. Crea un nuevo fichero con el nombre `PruebaScript.lua` y escribe las siguientes instrucciones:

```
a = 2 b = 2
print("; Hola Mundo! ")
print(a+b)
```

Ahora vamos a ejecutar el script usando Lua. Escribe el siguiente comando en la línea de comandos:

```
$ lua PruebaScript.lua
; Hola Mundo!
4
```

Si observas verás que es la misma salida que en el ejemplo interactivo, ya que de hecho, son las mismas instrucciones pero introducidas de modo diferente. Puede que te estés preguntando para qué sirve uno y otro modo. Sin entrar mucho en detalles, el uso más común del modo interactivo sería para el prototipaje, esto es, para hacer pruebas de conceptos. Si estás desarrollando un algoritmo y quieres saber qué resultado produce una instrucción, si no es muy compleja, el modo interactivo te ofrece una forma sencilla y rápida de hacer esa prueba. Por otro lado, el modo script es más adecuado para ejecutar programas más complejos o bien programas que desees persistir. Esto último es importante, porque el código que escribas en el modo interactivo se pierde al salir del modo interactivo.