

UNIT 12 PYTHON. SCRIPTING

Activities (II). Solutions

Computer Systems
CFGS DAW

Autores: Alfredo Oltra / Sergio García

alfredo.oltra@ceedcv.es

sergio.garcia@ceedcv.es

Revisado: Vicent Bosch

2020/2021

Versión:210407.1110

Licencia



Reconocimiento - NoComercial - CompartirIgual (by-nc-sa): No se permite un uso comercial de la obra original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

Nomenclatura

A lo largo de este tema se utilizarán distintos símbolos para distinguir elementos importantes dentro del contenido. Estos símbolos son:

🔔 Actividad opcional. Normalmente hace referencia a un contenido que se ha comentado en la documentación por encima o que no se ha hecho, pero es interesante que le alumno investigue y practique. Son tipos de actividades que no entran para examen

👁 Atención. Hace referencia a un tipo de actividad donde los alumnos suelen cometer equivocaciones.

UD012. SCRIPTING Activities (II). Solutions

Use forums to ask questions or help your classmates :)

Some of these exercises must be done in a Linux system.

1.1 Activity 1

a) Create a script that asks a number and checks if that number is prime or not. You can remember prime definition [here](#).

```
1  #!/usr/bin/python3
2
3  import math
4
5  def checkPrime(n):
6      if (n==1): #1 no es primo por convencion matematica
7          return False
8      elif (n==2):
9          return True
10     elif (n%2==0): #Caso de divisibles por 2 que no son 2
11         return False
12     else:
13         #Obtenemos la parte entera de la raiz del numero N
14         #que es hasta donde necesitamos comprobar
15         #Es mas optimo que limit=n
16
17         limit=int(math.sqrt(n))+1
18         for x in range(3,limit,2):
19             print(x)
20             if(n % x==0):
21                 return False
22     return True
```

This is an infinite loop ;-)

```
23
24  while (True):
25      print("Write a number to check if it is prime")
26      number=int(input())
27      if(checkPrime(number)):
28          print("It is prime")
29      else:
30          print("It is not prime")
```

b) Create a script that asks a number and calculates its [factorial](#)

```
Set as interpreter
1  #!/usr/bin/python3
2
3  import math
4
5  def factorial(n):
6      if (n==1):
7          return 1
8      else:
9          num=1
10
11         for i in range(2,n+1):
12             num=num*i
13
14         return num
15
16
17 while (True):
18     print("Write a number to print its factorial")
19     number=int(input())
20     print(factorial(number))
```

c) Create a script that shows the number of users currently logged in the system.

Tip: w or who command and their parameters could be useful to do this script.

This must be done in a Linux operating system.

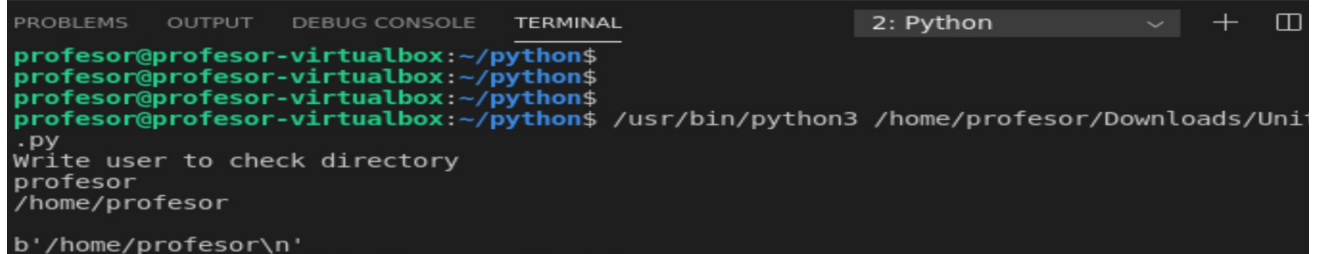
```
1  #!/usr/bin/python3
2  import subprocess
3
4  #OPCION 1
5  p = subprocess.Popen("who | wc -l", stdout=subprocess.PIPE, shell=True)
6  (output, err) = p.communicate()
7  #output da el formato en bytes, para pasar a cadena
8
9  print (str(output.decode("utf-8")))
10
11
12 #OPCION 2|
13 p2=subprocess.check_output("who | wc -l", shell=True)
14 print(p2)
```

d) Create a script that asks a username and shows its home directory. That information should be checked using `/etc/passwd` file.

```

1  #!/usr/bin/python3
2  import subprocess
3
4  print("Write user to check directory")
5  user=input()
6
7  #OPCION 1
8  p = subprocess.Popen("grep "+str(user)+ " /etc/passwd | cut -f 6 -d :",
9  stdout=subprocess.PIPE, shell=True)
10 (output, err) = p.communicate()
11 #output da el formato en bytes, para pasar a cadena
12 print (str(output.decode("utf-8")))
13
14 #OPCION 2
15 p2= subprocess.check_output("grep "+str(user)+ " /etc/passwd | cut -f 6 -d
16 :, shell=True)
17 print(p2)

```



The terminal window shows the script being executed. The prompt is 'Write user to check directory'. The user enters 'profesor'. The script then outputs the home directory: '/home/profesor'.

e) We have a file with two columns (name and country) separated by `:` like

```

Pedro:Spain
Juan:Spain
Mark:Holland
Vladimir:Russia
Terry:USA

```

We have to create a program that shows only country names. They should be shown ordered descendingly by its country name.

If a country appears more than one time, it should be shown only one time.

Tip: `sort` command and their parameters could be useful to do this script.

This must be done in a Linux operating system.

Try do solve it using similar operations seen in previous scripts.