UNIT 12 PYTHON. SCRIPTING

Activities I Solutions

Computer Systems
CFGS DAW

Autor: Vicent Bosch

vicent.bosch@ceedcv.es

2020/2021

Versión:210403.0956

Licencia

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 I Solutions

Use forums to ask questions or help your classmates :)

1.1 Activity 1

a) Create a script that creates, in current directory, N folders numbered from 00 to N. The numbers with one digit should contain a 0 on the left. The user must enter the number of folders N.

Solution not complete: How would you modify this script so that instead of 70 folders, you should introduce the number of folders using:

a) Keyboard

b) Arguments

```
#Importamos bibliotecas de sistema
import os, sys

#Bucle que recorre 70 elementos de uno en uno
for x in range(1,71,1):
    #Si es menor que diez, metemos 0 a la izquierda
    if x<10:
        #Creamos el directorio
        os.mkdir("0"+str(x))
    else:
        #Creamos el directorio
        os.mkdir(str(x))
```

b) Create a script that asks if a forum is good or bad, asking it using the keyboard. It has to repeat the question until the user writes *GOOD*.

```
#Inicialmente, answer vale cadena
answer=""

while (answer!="GOOD"):

    print("Is our forum GOOD or BAD?")
    #Leemos una linea del teclado y almacenamos en answer
    answer=input()

print("Ok :)")
```

c) Create a calculator that let you select an operation (add, subtract, multiply or divide) and let you enter two operands. At the end it will display the result of the operation.

```
print("Tell us and operation (+,-,* or /)")
#Leemos la operacion en una cadena
operation=input()
print("Tell first operand")
#Leemos la operacion en una cadena
operand1=float(input())
print("Tell second operand")
#Leemos la operacion en una cadena
operand2=float(input())

#Several if and else
if(operation=="+"):
    result=operand1+operand2
    print(result)
elif(operation=="-"):
    result=operand1-operand2
    print(result)
elif(operation=="*"):
    result=operand1*operand2
    print(result)
elif(operation=="/"):
    result=operand1/operand2
    print(result)
else:
    print("Wrong operation")
```

1.2 Activity 2

Create a program that checks whether a directory or a file exists in a specific location. The directory/file and the location must be arguments.

```python
# Importamos bibliotecas de sistema
# Más información: https://docs.python.org/3/library/os.path.html
import os, sys

# Para verificar los argumentos. No es necesario hacerlo.
print("Argument List:", str(sys.argv))

# Se pasan dos argumentos, pero el nombre del archivo siemprees el primer argumento,
# por tanto hay 3 argumentos.
if (len(sys.argv)!=3):
    print("Number or arguments incorrect")
else:
    # Se une directorio (argumento 1) + archivo/directorio (argumento 2)
    # para comprobar si es un directorio
    if (os.path.isdir(str(sys.argv[1])+"/"+str(sys.argv[2]))):
        print("Directory exists")
    else:
        print("This is not a directory, checking if it's a file")
        if (os.path.isfile(str(sys.argv[1])+"/"+str(sys.argv[2]))):
            print("File exists")
        else:
            print("This is not a file")
```

1.3 Activity 3

Create a program to check if hotel rooms are available (they are empty) or not (there is a guest in the room).

Our hotel has 10 rooms labeled from 00 to 10 (*this is not binary code*).

How to do it (proposed solution):

First, create 10 folders labeled for examples: room00NE, room01, room02NE, room03 and so on, where NE means not empty and the others, therefore, are empty.

Base your solution on checking the availability of a room. For example:

> Type a room to check: room02

> room02 is not empty.

The algorithm will try to find the room you've entered, if it finds it that means that the room is empty, if it doesn't find it, it means that the room is not empy.

Hint: you can create a directory or a file to identify each room.

1.4 Activity 4 (expands activity 3)

Create a program to check if hotel rooms are available (they are empty) or not (there is a guest in the room). Also available rooms can be *clean* or *not clean*.

Our hotel has 100 rooms.

In a loop, the program will ask a room number and then will display a menu with those options:

- Check if a room is available or not.
- Check if an available room is cleaned or not.
- Set a room to ***available*** or ***not available***.
- Set a room to ***cleaned*** or ***not cleaned***.
- Exit program.

If an operation can not be performed, our program must show an error.

The program **has to be persistent** (if you run again the program, configuration changes should remain).

Tip: you can create/delete files/directory and check if they exist.