

```

1  -*-*coding:utf-8*-*
2  #  SCRIPT QUE COMPARA LLAVES
3  #  Script Realizado por Diego Alberto Parra Garzón, Esto es software libre
4  #con licencia GPL3, Bogotá, Colombia
5  #
6  #Al cargar la clase:
7  #      CompararUsuario().build()
8  #carga inmediatamente la base de datos de usuarios.
9  #
10 # Para Comparar un usuario con la base de Datos se invoca a la función
11 #      self.CompararArray(NUID_QUE_SE QUIERE_COMPARAR)
12 # Trabaja como un ciclo en un hilo
13 #
14 # Para Revisar el estado de comparación se invoca a la función:
15 #      estadoCom():
16 # La cual devuelve:
17 # "1" si esta comparando,
18 # "2" si se encontro coincidencia en la base de datos,
19 # "3" si no esta en la base de datos.
20 # Es un ciclo en un hilo, trabajar con un while o un clock.schedule
21 #
22 #Para añadir un usuario se invoca primero la función
23 #      CompararArray(NUID_QUE_SE QUIERE_COMPARAR_PARA_AGREGAR)
24 # y luego se invoca
25 #      estadoCom():
26 # Si estadoCom devuelve 3, invocar la función:
27 #      agregarUsuario(nuid, name, codigoIdentificacion)
28 # Esta recibe 3 parametros: la nuid"ID de la llave RFID",
29 # name"Nombre completo del usuario",
30 # codigoIdentificacion"El código de asignación dado por la empresa.".
31 #
32 #Para quitar un usuario se invoca primero la función
33 #      CompararArray(NUID_QUE_SE QUIERE_COMPARAR_PARA_BORRAR)
34 # y luego se invoca
35 #      estadoCom():
36 # Si estadoCom devuelve 2 , invocar la función:
37 #      quitarUsuario()
38 #
39
40 from kivy.clock import Clock
41 from kivy.uix.relativelayout import RelativeLayout
42 from kivy.app import App
43
44 class CompararUsuario(RelativeLayout):
45
46     def CargarArray(self):
47         global array
48         array = []
49
50         #      with open("DataBase.text", "r") as f:
51         with open("Datos/DataBase.text", "r") as f:
52             for line in f:
53                 array.append(line)
54         print array
55
56     def CompararArray(self, valorAcomparar):
57         global nuid
58         self.item = 2
59         nuid = str(valorAcomparar)
60         print "Llamando a compara"
61         self.estadocomparacion = 1
62         self.estadoCom()
63         Clock.schedule_interval(self.compara, 0.01)
64

```

```

65
66     def compara(self, dt):
67         NUID = array[self.item].split(",")
68         print self.item
69         print NUID[0]
70         Nuid = str(NUID[0])
71
72         if ( Nuid == nuid):
73             print "El NUID coincide"
74             Clock.unschedule(self.compara)
75             self.estadocomparacion = 2
76             self.Vitem = self.item - 1
77             self.estadoCom()
78
79         if ( Nuid != nuid):
80             print "El NUID no coincide"
81             self.estadocomparacion = 1
82             self.estadoCom()
83             self.Vitem = self.item - 1
84             self.item = self.item+1
85
86         if (self.item == len(array) ):
87             if ( Nuid == nuid):
88                 print "El NUID coincide"
89                 Clock.unschedule(self.compara)
90                 self.estadocomparacion = 2
91                 self.Vitem = self.item
92                 self.estadoCom()
93
94                 if (Nuid != nuid):
95                     print "El NUID no coincide"
96                     Clock.unschedule(self.compara)
97                     self.estadocomparacion = 3
98                     self.Vitem = "nada"
99                     self.estadoCom()
100
101     def estadoCom(self):
102         #devuelve 1 si esta comparando
103         #devuelve 2 si esta en la base de datos
104         #devuelv 3 si no esta en la base de datos
105         eC = self.estadocomparacion
106         print "Estado de comparacion es: "
107         if eC ==1:
108             print "Comparando actualmente el item: " + str(self.item)
109         if eC ==2:
110             print "Se encontro una coincidencia, script terminado"
111         if eC ==3:
112             print "No se encontro ninguna coincidencia script terminado"
113         return eC
114
115
116     def quitarUsuario(self):
117         arrayCambiado = []
118         for i in range(0, len(array),1):
119             if (i == (int(self.Vitem) +1)):
120                 pass
121             if (i != (int(self.Vitem)+1)) :
122                 arrayCambiado.append(array[i])
123
124         print str(arrayCambiado)
125         var = open("Datos/DataBase.text", "w")
126         for i in range(0,len(arrayCambiado), 1):
127             var.write(str(arrayCambiado[i]))
128         # var.write("\n")

```

```

129         var.close()
130
131     def agregarUsuario(self, nuid, name, codigoIdentificacion):
132         variable = str(nuid)+","+str(name)+","+str(codigoIdentificacion)
133         # var = open("DataBase.text", "a")
134         var = open("Datos/DataBase.text", "a")
135         var.write(variable)
136         var.write("\n")
137         var.close()
138         self.CargarArray()
139
140     def build(self):
141         rl = RelativeLayout()
142         print "Cargando la base de datos."
143         self.CargarArray()
144         print "Base de datos cargada."
145         # self.CompararArray("C0C67A53")
146         # self.agregarUsuario("101324456","Diego parra Garzón","101010")
147         return rl
148
149     #class CompararUSER(App):
150     class CompararUSER(RelativeLayout):
151         def build(self):
152             return CompararUsuario().build()
153
154
155
156     #if __name__ == "__main__" :
157     #    CompararUSER().run()

```