

## Ejercicios/comprobar\_usuario.py.py

```
1  '''ANIDAR ACCESO A UN SITIO WEB:
2  Supongamos que eres un administrador de sistemas y necesitas validar el acceso de los
   usuarios
3  a un sitio web. Crea un script que verifique si el nombre de usuario y la contraseña
   ingresados son
4  correctos y permita el acceso solo si ambos son correctos.
5  Pista 1: Puedes crear dos listas, una con los nombre de usuario como por ejemplo...
6  nombres_usuario = ["juan123", "ana456", „pedro789"]
7  Y otra lista con las contraseñas guardadas para cada usuario...
8  contraseñas = ["clave123", "clave456", „clave789"]
9  Otra opción puede ser que crees una lista de listas con la forma:
10 nombres_contraseñas = [ ["juan123“, "clave123"] , ["ana456“, "clave456“] , ["pedro789“,
11 "clave789“] ]
12 Despues puedes pedir el usuario y contraseña y comprobar si coinciden.
13 Pista 2: Para verificar si el usuario y contraseña son correctos puedes crear un bucle donde
14 recorras los nombres de usuario y compruebes con un if si el nombre de usuario introducido y
   la
15 contraseña coinciden con los datos de tus listas. '''
16 # Lista de listas con nombres de usuario y contraseñas
17 nombres_contraseñas = [
18     ["juan123", "clave123"],
19     ["ana456", "clave456"],
20     ["pedro789", "clave789"]
21 ]
22 # Función para validar el acceso del usuario
23 def validar_acceso(usuario, contraseña):
24     for registro in nombres_contraseñas:
25         if registro[0] == usuario and registro[1] == contraseña:
26             return True
27     return False
28 # Solicitar al usuario que ingrese su nombre de usuario y contraseña
29 nombre_usuario = input("Ingresa su nombre de usuario: ")
30 contraseña_usuario = input("Ingresa su contraseña: ")
31 # Verificar si el acceso es válido
32 if validar_acceso(nombre_usuario, contraseña_usuario):
33     print("Acceso permitido. ¡Bienvenido!")
34 else:
35     print("Acceso denegado. Nombre de usuario o contraseña incorrectos.")
36 """Explicación:
37 Lista de listas: nombres_contraseñas contiene sublistas, cada una con un nombre de usuario y
   su contraseña correspondiente.
38 Función validar_acceso: Recorre la lista nombres_contraseñas y verifica si el nombre de
   usuario y la contraseña ingresados coinciden con algún registro.
39 Solicitar credenciales: El script pide al usuario que ingrese su nombre de usuario y
   contraseña.
40 Validar y mostrar resultado: La función validar_acceso se usa para verificar las credenciales
   ingresadas y se muestra un mensaje de acceso permitido o denegado según el resultado de la
   validación.
41 Este script básico puede ser mejorado agregando características como hashing de contraseñas
   para mayor seguridad, manejo de intentos fallidos, etc."""
```