**UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS**

UNIVERSIDAD DEL PERÚ, DECANA DE AMÉRICA



**Grupo 04 - Busqueda Indexada- Login**

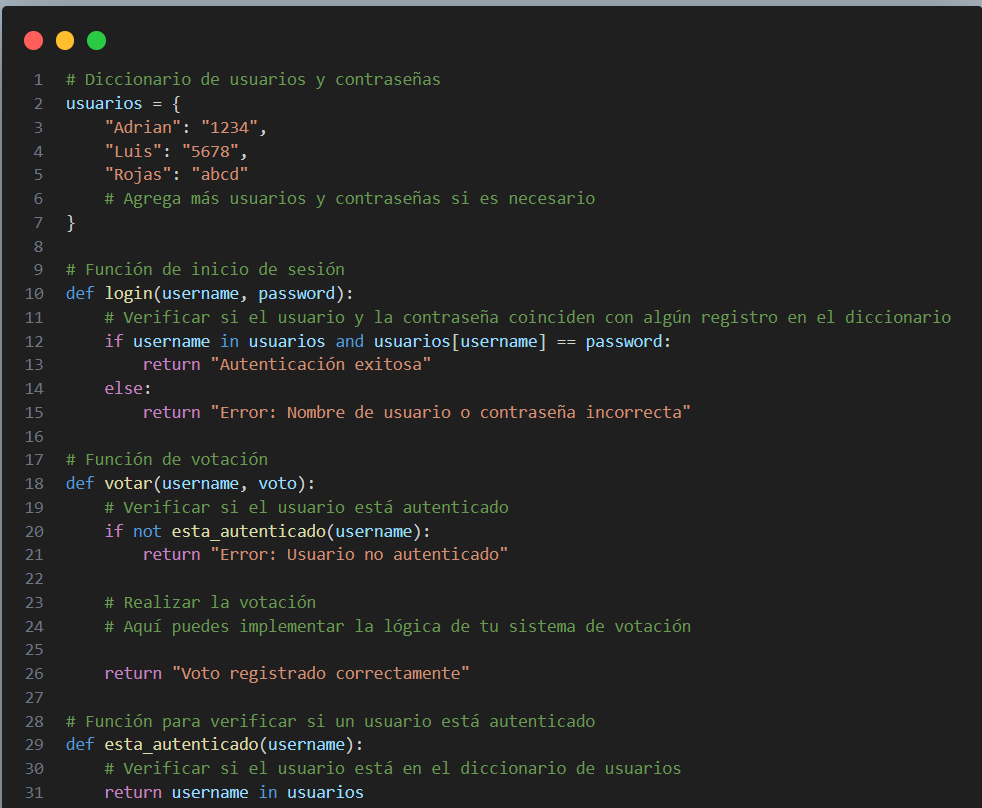
**Curso:** Análisis y Diseño de Algoritmos

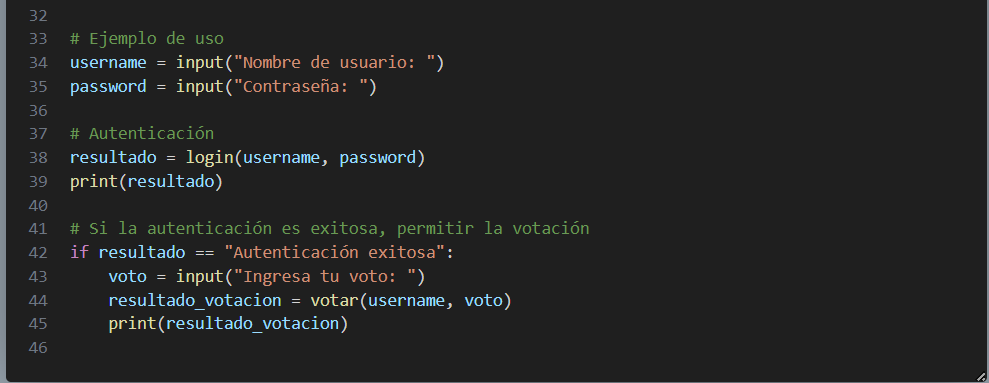
**Integrantes :**

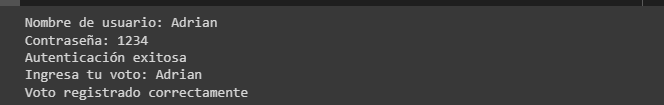
* Canecillas Contreras Juan Mariano
* Romero Cisneros Karlo Brandi
* Soller Barrenechea Carlos Javier
* Manco Méndez Elvis Neiser
* Huarhua Piñas Edson Sebastián

**Docente :** [Prudencio Vidal Javier](https://classroom.google.com/c/NjAyMzU1NTUxMTU3) Antonio

**Capturas de código**







El código anterior implementa un sistema de inicio de sesión y votación en línea utilizando una búsqueda indexada a través de un diccionario en Python. Veamos una explicación paso a paso y luego abordaremos por qué se considera una búsqueda indexada.

1. **Definición del diccionario de usuarios y contraseñas:**

El diccionario de usuarios almacena los nombres de usuario como claves y las contraseñas correspondientes como valores.

1. **Función de inicio de sesión (login):**

La función login recibe un nombre de usuario y una contraseña como parámetros.

Utiliza la búsqueda indexada en el diccionario usuarios para verificar si el nombre de usuario existe y si la contraseña proporcionada coincide con la almacenada en el diccionario.

Si la autenticación es exitosa, devuelve el mensaje "Autenticación exitosa". De lo contrario, devuelve el mensaje "Error: Nombre de usuario o contraseña incorrecta".

1. **Función de votación (votar):**

La función votar recibe un nombre de usuario y un voto como parámetros.

Verifica si el usuario está autenticado llamando a la función esta\_autenticado.

Implementa la lógica de tu sistema de votación. En este caso, solo devuelve un mensaje de "Voto registrado correctamente".

1. **Función para verificar si un usuario está autenticado (esta\_autenticado):**

La función esta\_autenticado recibe un nombre de usuario como parámetro.

Verifica si el nombre de usuario existe en el diccionario usuarios utilizando la búsqueda indexada.

Devuelve True si el usuario está autenticado y False en caso contrario.

1. **Ejemplo de uso:**

El programa solicita al usuario que ingrese su nombre de usuario y contraseña.

Luego, realiza la autenticación llamando a la función login con los datos ingresados.

Si la autenticación es exitosa, se permite al usuario ingresar su voto llamando a la función votar.

**¿Por qué se considera una búsqueda indexada?**

* En este caso, la búsqueda indexada se logra mediante el uso de un diccionario en Python.
* Al almacenar los usuarios y contraseñas como pares clave-valor en el diccionario, se puede acceder de forma rápida y eficiente a la contraseña correspondiente a un nombre de usuario dado.
* En lugar de realizar una búsqueda secuencial en una lista o recorrer una estructura de datos más compleja, el diccionario permite un acceso directo a través de la clave (nombre de usuario).
* Esto mejora significativamente la eficiencia de búsqueda, especialmente cuando hay una gran cantidad de usuarios registrados.
* Es importante tener en cuenta que en este caso, la búsqueda indexada se realiza en memoria, ya que el diccionario se mantiene durante la ejecución del programa y los datos no persisten más allá de eso. Si se requiere una persistencia de datos, se debería considerar el uso de una base de datos o almacenamiento externo.

**CÓDIGO**

# Diccionario de usuarios y contraseñas

usuarios = {

"Adrian": "1234",

"Luis": "5678",

"Rojas": "abcd"

# Agrega más usuarios y contraseñas si es necesario

}

# Función de inicio de sesión

def login(username, password):

# Verificar si el usuario y la contraseña coinciden con algún registro en el diccionario

if username in usuarios and usuarios[username] == password:

return "Autenticación exitosa"

else:

return "Error: Nombre de usuario o contraseña incorrecta"

# Función de votación

def votar(username, voto):

# Verificar si el usuario está autenticado

if not esta\_autenticado(username):

return "Error: Usuario no autenticado"

# Realizar la votación

# Aquí puedes implementar la lógica de tu sistema de votación

return "Voto registrado correctamente"

# Función para verificar si un usuario está autenticado

def esta\_autenticado(username):

# Verificar si el usuario está en el diccionario de usuarios

return username in usuarios

# Ejemplo de uso

username = input("Nombre de usuario: ")

password = input("Contraseña: ")

# Autenticación

resultado = login(username, password)

print(resultado)

# Si la autenticación es exitosa, permitir la votación

if resultado == "Autenticación exitosa":

voto = input("Ingresa tu voto: ")

resultado\_votacion = votar(username, voto)

print(resultado\_votacion)