Access Control Verification System in Python

In this project, I developed a Python program that simulates a basic access control system. The program uses predefined lists of approved users and their assigned devices to verify login attempts. It first checks if the provided username exists in the approved users list. If the user is authorized, the program retrieves the corresponding device ID and compares it with the one provided during login. Based on these conditions, the program displays appropriate messages indicating whether the user is approved, whether the device is correctly assigned, or if access should be denied. This approach demonstrates the use of conditional statements, indexing, and logical decision-making in Python to create a simple yet effective security check system.

```
    # If `username` belongs to `approved_users`,

    if username in approved_users:

        # then display "The user ______ is approved to access the system.",

        print("The user", username, "is approved to access the system.")

        # assign `ind` to the index of `username` in `approved_users`,

        ind = approved_users.index(username)

        # and execute the following conditional
        # If `device_id` matches the element at the index `ind` in␣
↪`approved_devices`,

        if device_id == approved_devices[ind]:

          # then display "______ is the assigned device for ______"

          print(device_id, "is the assigned device for", username)

        # Otherwise,

        else:

          # display "______ is not their assigned device"

          print(device_id, "is not their assigned device.")

    # Otherwise (part of the outer conditional and handles the case when␣
↪`username` does not belong to `approved_users`),

    else:

        # Display "The user ______ is not approved to access the system."

        print("The username", username, "is not approved to access the system.")

# Call the function you just defined to experiment with different username and␣
↪device_id combinations

login("bmoreno", "hl0s5o1")
login("elarson", "r2s5r9g")
login("abernard", "4n482ts")
```

```
The user bmoreno is approved to access the system.
hl0s5o1 is the assigned device for bmoreno
The user elarson is approved to access the system.
r2s5r9g is not their assigned device.
The username abernard is not approved to access the system.
```

Access Control Verification System in Python

This project highlights how fundamental programming concepts can be applied to real-world scenarios such as access control and authentication. By combining conditional logic with list operations, the program successfully simulates a two-step verification process, ensuring that both user identity and device authorization are validated. The exercise strengthened my understanding of Python's decision-making structures and demonstrated how they can be used to enhance security and system reliability.