

```
import json
import hashlib
import os

CREDENTIALS_FILE = "users.json"
DATA_DIR = "tasks"

def load_users():
    if not os.path.exists(CREDENTIALS_FILE):
        return {}
    with open(CREDENTIALS_FILE, "r") as f:
        return json.load(f)

def save_users(users):
    with open(CREDENTIALS_FILE, "w") as f:
        json.dump(users, f)

def hash_password(password):
    return hashlib.sha256(password.encode()).hexdigest()

def register():
    users = load_users()
    print("\n==== User Registration ===")
    username = input("Enter a new username: ").strip()

    if username in users:
        print("Username already exists. Please try logging in.")
        return None

    password = input("Enter a password: ").strip()
    users[username] = hash_password(password)
    save_users(users)
    print("Registration successful.")
    return username
```

```
def login():
    users = load_users()
    print("\n==== User Login ====")
    username = input("Enter username: ").strip()
    password = input("Enter password: ").strip()

    if username not in users:
        print("Username does not exist.")
        return None

    if users[username] == hash_password(password):
        print("Login successful.")
        return username
    else:
        print("Incorrect password.")
        return None


def task_file(username):
    if not os.path.exists(DATA_DIR):
        os.makedirs(DATA_DIR)
    return os.path.join(DATA_DIR, f"{username}_tasks.json")


def load_tasks(username):
    file_path = task_file(username)
    if not os.path.exists(file_path):
        return []
    with open(file_path, "r") as f:
        return json.load(f)


def save_tasks(username, tasks):
    file_path = task_file(username)
    with open(file_path, "w") as f:
        json.dump(tasks, f)


def add_task(username):
    print("\n==== Add Task ====")
```

```

desc = input("Enter task description: ").strip()
tasks = load_tasks(username)
task_id = 1 if not tasks else tasks[-1]["id"] + 1
tasks.append({"id": task_id, "description": desc, "status": "Pending"})
save_tasks(username, tasks)
print("Task added successfully.")

def view_tasks(username):
    print("\n==== Your Tasks ====")
    tasks = load_tasks(username)
    if not tasks:
        print("No tasks found.")
        return
    for t in tasks:
        print(f"ID: {t['id']} | {t['description']} | Status: {t['status']}")

def mark_completed(username):
    tasks = load_tasks(username)
    if not tasks:
        print("No tasks available.")
        return

    task_id = int(input("Enter task ID to mark completed: "))
    for t in tasks:
        if t["id"] == task_id:
            t["status"] = "Completed"
            save_tasks(username, tasks)
            print("Task marked as completed.")
            return
    print("Invalid Task ID.")

def delete_task(username):
    tasks = load_tasks(username)
    if not tasks:
        print("No tasks available.")

```

```
return

task_id = int(input("Enter task ID to delete: "))
new_tasks = [t for t in tasks if t["id"] != task_id]

if len(new_tasks) == len(tasks):
    print("Invalid Task ID.")
    return

save_tasks(username, new_tasks)
print("Task deleted successfully.")

def task_menu(username):
    while True:
        print(f"\nLogged in as: {username}")
        print("1. Add Task")
        print("2. View Tasks")
        print("3. Mark Task as Completed")
        print("4. Delete Task")
        print("5. Logout")

        choice = input("Enter choice: ").strip()

        if choice == "1":
            add_task(username)
        elif choice == "2":
            view_tasks(username)
        elif choice == "3":
            mark_completed(username)
        elif choice == "4":
            delete_task(username)
        elif choice == "5":
            print("Logging out...")
            break
        else:
            print("Invalid choice. Please try again.")

def main():
```

```
while True:
    print("\n==== Task Manager ===")
    print("1. Register")
    print("2. Login")
    print("3. Exit")
    choice = input("Enter choice: ").strip()

    if choice == "1":
        user = register()
        if user:
            task_menu(user)
    elif choice == "2":
        user = login()
        if user:
            task_menu(user)
    elif choice == "3":
        print("Exiting system...")
        break
    else:
        print("Invalid choice. Please try again.")

if __name__ == "__main__":
    main()
```