In [24]:
```
import hashlib
import os
import csv
```

In [10]:
```
def hash_password(password):
    """hash the password for security before storing it in a file"""
    return hashlib.sha256(password.encode()).hexdigest()
```

In [26]:
```
'''usernames and passwords will be stored in a txt file'''
if not os.path.exists("users.txt"):
    with open("users.txt", "w") as f:
        pass  # create an empty file

''' tasks will be stored in a csv file for easier management'''
if not os.path.exists("tasks.csv"):
        with open("tasks.csv", "w", newline='') as f:
            writer = csv.writer(f)
            writer.writerow(["username", "task_id", "description", "status"])
```

In [21]:
```
def register_user():
    '''Create a function to prompt the user to enter a username and password''
    username = input("Enter a new username: ")
    password = input("Enter a new password: ")
    hashed_pd = hash_password(password)

    '''Ensure that the username is unique'''
    with open("users.txt", "r") as f:
        for line in f:
            existing_username, _ = line.strip().split(",", 1)
            if username == existing_username:
                print("Usernames must be unique. Please try a different one.")
                return

    '''hash the password for security before storing it in a file'''
    with open("users.txt", "a") as f:
        f.write(f"{username},{hashed_pd}\n")

    print(f"{username} registered successfully.")
```

In [22]:
```
register_user()
```

```
user5 registered successfully.
```

In [14]:
```
def login_user():
    username = input("Enter your username: ")
    password = input("Enter your password: ")
    hashed = hash_password(password)

    with open("users.txt", "r") as f:
        for line in f:
            stored_username, stored_hash = line.strip().split(",", 1)
            if username == stored_username and hashed == stored_hash:
                print("Login successful!")
                return username

    '''if not, the the function will reach this line and prints the following
    print("Incorrect username or password.")
    return None
```

In [17]:
```
login_user()
```

```
Invalid username or password.
```

In [27]:
```
def add_task(username):

    '''Create a function that prompts the user for a task description.
       Assign a unique task ID and set the status to Pending
       Store the task in a file, and confirm that the task was added'''

    task_description = input("Enter task description: ")

    # Read all tasks to calculate the next task_id for currently logged in use
    task_id = 1
    with open("tasks.csv", "r") as f:
        reader = csv.reader(f)
        next(reader)  # skip header
        for row in reader:
            if row[0] == username:
                task_id += 1

    with open("tasks.csv", "a", newline='') as f:
        writer = csv.writer(f)
        writer.writerow([username, task_id, task_description, "Pending"])

    print(f"Task #{task_id} added successfully for '{username}'.")
```

In [ ]:
```
logged_in_user = login_user()
```

In [35]:
```
add_task(logged_in_user)
```

```
Task #3 added successfully for 'user1'.
```

In [32]:
```
def view_tasks(username):
    '''Create a function to retrieve and display all tasks for the logged-in us
       Each task should show the task ID, description, and status (Pending or (

    print(f"\n{username} - tasks:")
    with open("tasks.csv", "r") as f:
        reader = csv.reader(f)
        next(reader)  # skip header
        found = False

        #iterate through each row
        for row in reader:
            if row[0] == username:
                print(f"ID: {row[1]} | Description: {row[2]} | Status: {row[3]
                found = True

    if not found:
        print("No tasks found for logged in user.")
```

In [43]:
```
view_tasks(logged_in_user)
```

```
user2 - tasks:
ID: 1 | Description: Finalize assignment | Status: Pending
ID: 2 | Description: Take test | Status: Completed
```

In [41]:

```
def mark_task_completed(username):
    task_id = input("Enter the task ID to mark it as completed: ")

    updated_rows = []
    task_found = False

    with open("tasks.csv", "r") as f:
        reader = csv.reader(f)
        header = next(reader)
        for row in reader:
            if row[0] == username and row[1] == task_id:
                row[3] = "Completed"
                task_found = True
            updated_rows.append(row)

    #Update rows in CSV

    with open("tasks.csv", "w", newline='') as f:
        writer = csv.writer(f)
        writer.writerow(header)
        writer.writerows(updated_rows)

    if task_found:
        print(f"Task #{task_id} marked as completed.")
    else:
        print(f"No task with ID {task_id} found for '{username}'.")
```

In [42]:

```
mark_task_completed(logged_in_user)
```

```
Task #2 marked as completed.
```

In [44]:

```
def delete_task(username):

    '''Create a function that allows the user to select a task by its ID and de

    task_id = input("Enter the task ID to delete: ")

    updated_rows = []
    task_deleted = False

    with open("tasks.csv", "r") as f:
        reader = csv.reader(f)
        header = next(reader)
        for row in reader:
            if row[0] == username and row[1] == task_id:
                task_deleted = True
                continue  # skip this row
            updated_rows.append(row)

    with open("tasks.csv", "w", newline='') as f:
        writer = csv.writer(f)
        writer.writerow(header)
        writer.writerows(updated_rows)

    if task_deleted:
        print(f"Task #{task_id} deleted.")
    else:
        print(f"No task with ID {task_id} found for '{username}'.")
```

In [45]:
```
logged_in_user
```

Out[45]:
```
'user2'
```

In [48]:
```
delete_task(logged_in_user)
```

```
No task with ID 3 found for 'user2'.
```

In [47]:
```
view_tasks(logged_in_user)
```

```
user2 - tasks:
ID: 2 | Description: Take test | Status: Completed
```

In [53]:
```
def start_task_manager():
    print("\nWelcome to the Task Manager")

    username = login_user()
    if logged_in_user:
        while True:
            print(f"\nWelcome, {username}! What would you like to do?")
            print("1. Add Task")
            print("2. View Tasks")
            print("3. Mark Task as Completed")
            print("4. Delete Task")
            print("5. Logout")

            choice = input("Enter your choice (1-5): ")

            if choice == "1":
                add_task(username)
            elif choice == "2":
                view_tasks(username)
            elif choice == "3":
                mark_task_completed(username)
            elif choice == "4":
                delete_task(username)
            elif choice == "5":
                print(f"Logging out {username}...\n")
                break
            else:
                print("Invalid choice. Please enter a number from 1 to 5.")
    else:
        print("You must be logged in to use the Task Manager.")
```

In [54]:
```
start_task_manager()
```

```
Welcome to the Task Manager
```

```
Login successful!

Welcome, user1! What would you like to do?
1. Add Task
2. View Tasks
3. Mark Task as Completed
4. Delete Task
5. Logout

Task #4 added successfully for 'user1'.

Welcome, user1! What would you like to do?
1. Add Task
2. View Tasks
3. Mark Task as Completed
4. Delete Task
5. Logout

user1 - tasks:
ID: 1 | Description: Finalize report | Status: Pending
ID: 2 | Description: Prepare for marathon | Status: Pending
ID: 3 | Description: Do laundry | Status: Pending
ID: 4 | Description: test | Status: Pending

Welcome, user1! What would you like to do?
1. Add Task
2. View Tasks
3. Mark Task as Completed
4. Delete Task
5. Logout

Task #4 marked as completed.

Welcome, user1! What would you like to do?
1. Add Task
2. View Tasks
3. Mark Task as Completed
4. Delete Task
5. Logout

user1 - tasks:
ID: 1 | Description: Finalize report | Status: Pending
ID: 2 | Description: Prepare for marathon | Status: Pending
ID: 3 | Description: Do laundry | Status: Pending
ID: 4 | Description: test | Status: Completed

Welcome, user1! What would you like to do?
1. Add Task
2. View Tasks
3. Mark Task as Completed
4. Delete Task
5. Logout

Task #3 deleted.

Welcome, user1! What would you like to do?
1. Add Task
2. View Tasks
3. Mark Task as Completed
4. Delete Task
5. Logout
```

```
user1 - tasks:
ID: 1 | Description: Finalize report | Status: Pending
ID: 2 | Description: Prepare for marathon | Status: Pending
ID: 4 | Description: test | Status: Completed

Welcome, user1! What would you like to do?
1. Add Task
2. View Tasks
3. Mark Task as Completed
4. Delete Task
5. Logout

Logging out user1...
```