# Task Manager with User Authentication

## Objective

The objective of this project is to build a simple console-based Task Manager in Python that allows multiple users to securely register, authenticate, and manage their tasks independently. The application demonstrates basic concepts of file handling, user authentication, and CRUD operations.

## Project Overview

This project is a Python-based Task Manager application. It enables users to create accounts, log in, and perform task management activities such as adding tasks, viewing all tasks, marking tasks as completed, and deleting tasks. User credentials and task data are stored persistently in text files so that data remains intact between sessions.

The program uses:

* **SHA-256 hashing** to store passwords securely.
* Simple **file-based storage** for both users and tasks.
* A **menu-driven console interface** to guide the user through all operations.

## Features

### User Registration and Login

* New users can register by providing a username and password.
* Passwords are hashed and stored in users.txt.
* Existing users can log in using their credentials.

### Add Task

* Users can create new tasks with descriptions.
* Each task is assigned a unique Task ID.

### View Tasks

* All tasks are displayed with their status (Pending or Completed).

### Mark Task as Completed

* Users can mark tasks as completed by specifying the Task ID.

### Delete Task

* Tasks can be removed from the task list.

### Persistent Storage

* User accounts and task files remain available across sessions.

## Conclusion

This project demonstrates how to combine:

* Basic authentication principles,
* File I/O operations,
* Command-line user interaction,  
  to create a functional console-based task manager in Python. It is a solid foundation for understanding how user and task management can be implemented without external databases or frameworks.